Министерство науки и высшего образования Российской Федерации

Калужский филиал
федерального государственного бюджетного образовательного учреждения высшего образования
**«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»**
(КФ МГТУ им. Н.Э. Баумана)

Е.В.Красавин, Ю.Е.Гагарин, Амеличева К.А.

Лабораторный практикум по дисциплине «Технологии системного программного обеспечения»: учебное пособие

Калуга – 2024

УДК 004.62
ББК 32.972.1
Б435

Рецензенты:

Доцент кафедры «Системы обработки информации» КФ МГТУ им. Н.Э.Баумана канд. техн.наук, доц. В.О.Трешневская

Заведующий отделом «Численного анализа пассивной безопасности» Центра «ЧАиВВ» ФГУП «НАМИ», канд. техн.наук Д.Ю.Солопов

Утверждено Методической комиссией КФ МГТУ им.Н.Э.Баумана (протокол №_ от __._____._____ г., рег. Номер _____)

Красавин Е.В., Гагарин Ю.Е, Амеличева К.А.

Лабораторный практикум по дисциплине «Проектирование программого обеспечения»: учебное пособие / Е.В.Красавин, Ю.Е.Гагарин, Амеличева К.А. – Калуга: КФ МГТУ им. Н.Э.Баумана, 2024. -159 с.

В учебном пособии приведены теоретические сведения и характеристика исходных данных для выполнения лабораторных работ, рекомендации по их выполнению, требования к оформлению, рекомендуемые источники информации.

Учебное пособие предназначено для студентов КФ МГТУ им. Н.Э.Баумана, обучающихся по направлению подготовки 09.03.04 «Программная инженерия».

Оглавление

ВВЕДЕНИЕ

Настоящий лабораторный практикум составлен в соответствии с программой проведения лабораторных работ по дисциплине «Технологии системного программного обеспечения» на кафедре «Программное обеспечение ЭВМ, информационные технологии» факультета информатики и управления Калужского филиала МГТУ им. Н.Э. Баумана.

Лабораторный практикум предназначен для студентов 4-го курса направления подготовки 09.03.04 «Программная инженерия» и содержит цели и задачи лабораторных работ, основные теоретические сведения, дается описание порядка выполнения и методические указания, приведены контрольные вопросы и формы отчетов по лабораторным работам.

Выполнение лабораторного практикума позволит студентам получить и закрепить знания, умения и навыки, достижения которых является результатом освоения дисциплины «Технологии системного программного обеспечения».

ПРАВИЛА БЕЗОПАСНОСТИ ПРИ ВЫПОЛНЕНИИ ЛАБОРАТОРНЫХ РАБОТ

При выполнении лабораторных работ необходимо руководствоваться требованиями Инструкции по охране труда для пользователей персональных компьютеров (ПК) ИОТ 020-2018.

ЛАБОРАТОРНАЯ РАБОТА №1
УСТАНОВКА И НАСТРОЙКА FREEBSD

ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Целью выполнения лабораторной работы является получение практических навыков по установке и запуску ОС FreeBSD.

Основными задачами выполнения лабораторной работы являются:

1. Научиться устанавливать FreeBSD.
2. Изучить процесс загрузки.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Что такое FreeBSD?

FreeBSD – это операционная система, подобная UNIX, которая свободно доступна в Интернете. Она широко применяется в компанияхпровайдерах услуг Интернета, во встроенных устройствах и в любом другом месте, где важна надежность. Операционная система FreeBSD – это результат непрерывного, в течение более тридцати лет, процесса разработки, исследований и доводки. FreeBSD основана на 4.4BSD-Lite и предназначена для компьютеров Intel (x86 и Itanium®), AMD64, Alpha™ и Sun UltraSPARC®. Ведется работа по портированию и на другие архитектуры.

Что может FreeBSD?

FreeBSD имеет заслуживающие внимания возможности. Некоторые из них:

■ Вытесняющая многозадачность с динамическим регулированием приоритетов, позволяющая плавно и справедливо распределить ресурсы компьютера между приложениями и пользователями, даже при тяжелейших нагрузках.

■ Многопользовательская поддержка, которая позволяет множеству людей использовать FreeBSD совместно для различных задач. Это значит, например, что системная периферия, такая как принтеры и ленточные устройства, правильно разделяется всеми пользователями в системе или сети, и что пользователям или группам пользователей могут быть установлены лимиты каждого ресурса, защищая критические системные ресурсы от перегрузок.

■ Мощный TCP/IP-стек с поддержкой промышленных стандартов, таких как SLIP, PPP, NFS, DHCP и NIS. Это означает, что FreeBSD может легко взаимодействовать с другими системами, а также работать сервером масштаба предприятия, предоставляя жизненно важные функции, такие как NFS (удалённый доступ к файлам) и услуги электронной почты, или представить вашу организацию в Интернете, обеспечивая работу служб WWW, FTP, маршрутизацию и функции межсетевого экрана (брандмауэра).

■ Защита памяти гарантирует, что приложения (или пользователи) не смогут чинить препятствия друг другу. Фатальная ошибка в выполнении одного приложения не скажется на работоспособности всей системы.

■ FreeBSD 32-разрядная операционная система (64-разрядная на Alpha, Itanium®, AMD64, и UltraSPARC®) и изначально создавалась именно такой.

■ Промышленный стандарт X Window System (X11R6) предоставляет графический интерфейс пользователя (GUI) для большинства VGA карт и мониторов, и поставляется с полными исходными текстами.

■ Тысячи готовых к использованию приложений доступны из коллекций портов и пакетов FreeBSD. Зачем искать что-то в сети, когда вы можете найти всё прямо здесь?

■ Тысячи других легко адаптируемых приложений доступны в Интернете. FreeBSD совместима по исходным текстам с большинством популярных коммерческих UNIX® -систем и, таким образом, большинство приложений требуют лишь небольших изменений для сборки (или не требуют вообще).

■ Виртуальная память с поддержкой сброса неиспользуемых

страниц по требованию и «объединение виртуальной памяти и буферного кэша» спроектированы так, чтобы максимально эффективно удовлетворить приложения с огромными аппетитами к памяти и, в то же время, сохранить интерактивность для остальных пользователей.

- Поддержка симметричной многопроцессорности (SMP) для машин с несколькими процессорами.
- Доступность исходных текстов всей системы означает, что вы имеете максимальный контроль над операционной средой.

  Зачем выбирать закрытые решения и уповать на милость производителя, когда вы можете получить по-настоящему открытую систему?
- Обширная online-документация.
- И многое-многое другое!

FreeBSD основана на 4.4BSD-Lite от Computer Systems Research Group (CSRG) Калифорнийского Университета, Беркли, и продолжает славную традицию разработки BSD-систем. В дополнении к прекрасной работе, предоставленной CSRG, Проект FreeBSD тратит многие тысячи часов для тонкой настройки системы для максимальной производительности и надёжности в условиях максимально приближенным к «боевым». Когда большинство коммерческих гигантов только пытаются достичь такого уровня возможностей, производительности и надежности операционных систем для ПК, FreeBSD может предложить все это прямо сейчас!

Применение FreeBSD в действительности ограничено только вашим воображением. От разработки программного обеспечения до автоматизации производства, от складского учета до дистанционной коррекции азимутов спутниковых антенн; если задачи можно решить с помощью коммерческих UNIX® -систем, скорее всего, они решаемы и с помощью FreeBSD! FreeBSD также существенно выигрывает за счет буквально тысяч высококачественных приложений, разработанных исследовательскими центрами и университетами во всём мире, и доступных за минимальную цену или даже бесплатно. Коммерческие приложения также доступны, и их с каждым днем становится всё больше.

Поскольку исходные тексты FreeBSD общедоступны, система может быть оптимизирована в почти невероятной степени для специальных

приложений или проектов, а это, обычно, невозможно при использовании операционных систем от большинства коммерческих производителей. Вот несколько примеров того, как сейчас используется FreeBSD:

■ Интернет-службы: мощнейший TCP/IP стек делает FreeBSD идеальной платформой для большинства Интернет-приложений, таких как:

- FTP-серверы
- Серверы World Wide Web
- Серверы новостей или дискуссионных групп USENET

■ Образование: Вы студент и ваше образование связано с компьютерами или другими инженерными дисциплинами? Нет лучшего пути начать изучение операционных систем, архитектуры компьютера и работы в сети, чем освоить FreeBSD. Количество свободно доступных пакетов САПР, математических и графических пакетов также делают её чрезвычайно полезной для тех, кто использует компьютер как инструмент для выполнения другой работы!

■ Исследования: За счёт доступности исходных текстов для всей системы, FreeBSD — превосходная платформа как для изучения операционных систем и исследований в других областях компьютерных наук. Свободная природа FreeBSD позволяет удалённым группам сотрудничать, обмениваться идеями и совместными разработками, не беспокоясь о наличии специальных лицензий или ограничений на то, что может обсуждаться в открытых форумах.

■ Разработка программного обеспечения: Базовая поставка FreeBSD распространяется с полным набором инструментов для разработки, включая знаменитые компилятор GNU C/C++ и отладчик.

FreeBSD доступна как в исходных текстах, так и в двоичном виде на CDROM, DVD и через анонимный доступ к FTP.

Кто использует FreeBSD?

FreeBSD используется в качестве платформы на некоторых крупнейших сайтах в интернете, включая:

■ Yahoo!

■ Apache

■ Blue Mountain Arts

■ Pair Networks

■ Sony Japan

- Netcraft
- Weathernews
- Supervalu
- TELEHOUSE America
- Sophos Anti-Virus
- JMA Wired
- на многих других.

Цели проекта

Целью Проекта FreeBSD является предоставление программного обеспечения, которое может быть использовано для любых целей и без дополнительных ограничений. Многие из разработчиков внесли значительный вклад в код (и проект) и совершенно не против получать за это иногда финансовую компенсацию, но они определенно ее не требуют. Первая и основная «миссия» проекта — это предоставление кода для всех, кому он необходим, и для любых целей, так чтобы этот код становился всё более распространённым и предоставлял самые широкие возможности. Это является одной из наиболее фундаментальных целей Свободного Программного Обеспечения, и разработчики FreeBSD с энтузиазмом поддерживают её.

Тот код в дереве исходных текстов, который попадает под Стандартную Общественную Лицензию GNU (GPL) или Стандартную Общественную Лицензию Ограниченного Применения GNU (LGPL), предоставляется с дополнительными условиями, хотя они обеспечивают только возможность доступа, а не его ограничение. По причине дополнительных сложностей, которые могут появиться при
коммерческом использовании GPL-продуктов, программное обеспечение, предоставленное под более свободной лицензией BSD, является более предпочтительным.

Модель Разработки FreeBSD

Разработка FreeBSD — это очень открытый и гибкий процесс. FreeBSD в буквальном смысле создана из кода, предоставленного сотнями людей со всего мира, в чем вы можете убедиться, взглянув на список этих людей. Инфраструктура разработки FreeBSD позволяет этим

сотням разработчиков сотрудничать с помощью Интернета. Ведется постоянный поиск новых разработчиков и новых идей, и тех, кто заинтересован в более тесном взаимодействии и хочет принять участие в проекте. Для тех, кто желает уведомить других пользователей FreeBSD об основных направлениях работы, доступен Список рассылки анонсов FreeBSD.

Для независимой работы или тесного сотрудничества, полезно знать о проекте и процессе разработки FreeBSD следующее:

CVS-репозиторий

Главное дерево исходных текстов FreeBSD поддерживается с помощью CVS (Concurrent Versions System), свободно доступной системой контроля исходных текстов, которая поставляется вместе с FreeBSD. Основной CVS репозиторий располагается на компьютере, находящемся в городе Санта Клара, Калифорния (США), откуда и распространяется на множество зеркал по всему миру. Дерево CVS, содержащее ветви - CURRENT и -STABLE, может быть легко скопировано на ваш локальный компьютер.

Коммитеры

Коммиттеры — это люди, которые имеют доступ на запись к главному дереву CVS, и имеют право вносить изменения в главное дерево исходных текстов FreeBSD (термин «коммиттер» появился от названия команды cvs(1) commit , которая используется для внесения изменений в CVS-репозиторий). Лучший способ предоставить ваши

соображения на рассмотрение коммиттеров — использовать команду send-pr(1). Если что-то произошло с системой, вы можете достучаться до них посылкой письма по адресу cvs-committers.

Core-группа FreeBSD

Core-группа FreeBSD могла бы быть эквивалентом Совета Директоров, если бы Проект FreeBSD был компанией. Главная задача Core-группы — гарантировать, что проект в целом в хорошем состоянии и движется в правильном направлении. Приглашение постоянных и

ответственных разработчиков присоединиться к группе коммиттеров — одна из функций Core-группы, так же, как и приглашение новых членов в Core-группу по мере того, как другие уходят. Нынешний состав команды был выбран из рядов коммиттеров путем общего голосования в июле 2006 года. Выборы проходят каждые 2 года.

Некоторые члены Core-группы имеют особые области ответственности, то есть, они являются ответственными за работу отдельной большой части системы. Полный список разработчиков FreeBSD и областей их ответственности можно найти в Списке участников.

Примечание:

Большинство членов Core-группы — волонтёры, и не получают никакой финансовой выгоды от участия в проекте, поэтому вы не должны рассматривать возложенную на них «ответственность» как «гарантированную поддержку».

Внешняя помощь

Последней, но однозначно не менее значимой, и наибольшей группой разработчиков являются сами пользователи, которые предоставляют комментарии и исправления ошибок на почти постоянной основе. Основной путь участвовать в не централизованной разработке — это подписка на Список рассылки FreeBSD, посвящённый техническим дискуссиям, где обсуждаются подобные вещи.

Предоставление кода — не единственный способ помочь проекту; более полный список того, что необходимо сделать, можно найти на Web-сайте проекта FreeBSD.

Вообще говоря, модель разработки организована как «нечеткий набор концентрированных колец». Централизованная модель разработана для удобства пользователей FreeBSD, которые получают простую систему контроля за одной центральной базой кода, и позволяет не оставить за бортом проекта потенциальных помощников! Проект нацелен на то, чтобы предоставить стабильную операционную систему с большим количеством согласованных прикладных программ, которые пользователи смогут легко установить и использовать — эта модель очень хорошо подходит для решения этой задачи.

FreeBSD.org

Вебсайт FreeBSD (http://www.freebsd.org) содержит массу разнообразной информации по вопросам установки и администрирования FreeBSD. Наиболее важными частями являются Справочник (Handbook), сборник FAQ (Frequently Asked Question, часто задаваемые вопросы) и архивы почтовых рассылок, однако здесь же вы найдете огромное число статей на самые разные темы. В дополнение к документации о FreeBSD на вебсайте также имеется большой объем информации о внутреннем руководстве проектом FreeBSD и о состоянии различных частей проекта. Если основной вебсайт работает слишком медленно, то можно воспользоваться зеркалом сайта. На основном сайте имеется раскрывающийся список национальных сайтов зеркал, кроме того можно попробовать ввести адрес в формате http://www.<код_страны>.freebsd.org. Практически во всех странах существуют свои сайты, дублирующие вебсайт FreeBSD.

УСТАНОВКА ОС FREEBSD

Обзор

FreeBSD поставляется с простой в использовании текстовой программой установки. FreeBSD 9.0-RELEASE и более поздние укомплектованы установщиком, называемым bsdinstall, в то время как в релизах, предшествующих FreeBSD 9.0-RELEASE, для установки используется sysinstall.

Аппаратные требования

Минимальная конфигурация

Минимальная аппаратная конфигурация, достаточная для установки FreeBSD, зависит от версии FreeBSD и от аппаратной архитектуры. В зависимости от способа установки FreeBSD вам также может потребоваться поддерживаемый привод CDROM, а в некоторых случаях — сетевой адаптер.

FreeBSD/i386

Для FreeBSD/i386 необходим 486 процессор или выше, а также — как минимум 64 МБ ОЗУ. Для самой минимальной установки потребуется не менее 1.1 ГБ свободного места на жестком диске.

Примечание:

Для устаревших компьютеров более эффективным способом повышения производительности является увеличение объема ОЗУ и объема жесткого диска, нежели установка более быстродействующего процессора.

FreeBSD/amd64

Существует два класса процессоров, на которых может работать FreeBSD/amd64. К первому принадлежат процессоры AMD64, включая AMD Athlon™64, AMD Athlon™64-FX, AMD Opteron™ и более новые.

Ко второму классу процессоров, на которых работает FreeBSD/amd64, принадлежат процессоры архитектуры Intel® EM64T. Перечень процессоров включает следующие семейства: Intel® Core™

2 Duo, Quad, Extreme, семейства Intel® Xeon™ 3000, 5000 и 7000, а также Intel® Core™ i3, i5 и i7.

Если ваш компьютер построен на чипсете nVidia nForce3 Pro-150, то вам необходимо отключить IO APIC в BIOS. Если для этого нет опции в BIOS, отключите ACPI в операционной системе. В чипсете Pro-150 содержатся ошибки, для которых пока не существует исправлений.

FreeBSD/powerpc Apple® Macintosh®

Поддерживаются все американские системы Apple® Macintosh® с встроенным USB. Для многопроцессорных машин есть поддержка SMP.

Ядро (32-бит) может адресовать лишь первые 2 ГБ ОЗУ. На Blue & White PowerMac G3 не поддерживается FireWire®.

FreeBSD/sparc64

Поддерживаемые FreeBSD/sparc64 системы перечислены в проекте FreeBSD/ sparc64.

Для FreeBSD/sparc64 требуется отдельный жесткий диск. На данный

момент нет возможности разделять диск с другой операционной системой.

Поддерживаемое оборудование

Архитектуры и устройства, поддерживаемые каждым релизом FreeBSD, перечислены в файле Hardware Notes. Файл, как правило, называется HARDWARE.TXT, и располагается в корневом каталоге установочного носителя. Также копии списка поддерживаемого оборудования находятся на странице Release Information веб сайта FreeBSD.

Перед установкой

Сделайте резервные копии данных

Сделайте резервные копии всех важных данных с того компьютера, на который планируется установка FreeBSD. Проверьте пригодность резервных копий до начала установки. Перед внесением изменений на

диск инсталлятор FreeBSD запросит подтверждение, но как только изменения будут внесены, то отменить их уже будет невозможно.

Решите куда установить FreeBSD

Если FreeBSD будет единственной установленной операционной системой, и она будет занимать весь жесткий диск, то можете смело пропустить этот раздел. Но если FreeBSD будет разделять диск с другими операционными системами, то во время установки вам понадобится понимание принципов разбиения дисков.

Разделы диска для FreeBSD/i386 и FreeBSD/amd64

Весь объем жестких дисков может быть разделен на множество частей. Эти части называются разделами.

Есть два способа деления диска на разделы. Традиционный способ — Master Boot Record (MBR) — хранит таблицу разделов, вмещающую до четырех первичных разделов. (Так сложилось исторически, что во FreeBSD эти разделы называются слайсами.) Возможны ситуации, в которых четыре раздела недостаточно, поэтому один из первичных разделовможет быть превращен в расширенный раздел. Внутри

расширенного раздела может быть создано несколько логических разделов. Результирующая структура выглядит немного неуклюже, но такова она есть.

Создание Таблицы Разделов GUID (GUID Partition Table, GPT) — это более новый и простой способ деления диска. Также новый способ (GPT) по сравнению с традиционным способом разбиения (MBR) гораздо более гибкий. Распространённые реализации GPT позволяют создавать до 128 разделов на одном диске, тем самым исключая необходимость создания неудобных сущностей наподобие логических дисков.

Предупреждение

Некоторые старые операционные системы, например, Windows® XP, не совместимы со схемой GPT. Если на один диск необходимо установить FreeBSD совместно с такой операционной системой, то следует воспользоваться схемой MBR.

Стандартному загрузчику FreeBSD необходим первичный раздел (MBR) или GPT раздел. Если все первичные или GPT разделы уже задействованы, то для FreeBSD один из них необходимо будет освободить.

Минимальная установка FreeBSD занимает ни много ни мало — 1 ГБ дискового пространства. Однако, это очень минимальная установка, практически не оставляющая свободного места. Более реалистичным минимумом является 3 ГБ без графической подсистемы, а если будет использоваться графическая подсистема, то 5 ГБ или более. Свободное пространство также потребуется приложениям от третьих лиц.

Для создания разделов существует разнообразие свободно распространяемых и коммерческих утилит. GParted Live это свободно распространяемый загрузочный дистрибутив, в который включен редактор разделов GParted. Также GParted включен в многие другие дистрибутивы Live CD от Linux.

Предупреждение

Утилиты для создания разделов могут повредить ваши данные. Поэтому сделайте полную резервную копию и проверьте её целостность перед модификацией разделов диска.

Определенные трудности составляет изменение размеров разделов

Microsoft® Vista. В таких случаях может пригодиться установочный диск от самой Microsoft® Vista.

Пример 1. Использование существующего раздела

Компьютер с ОС Windows® имеет жесткий диск размером 40 ГБ, диск разбит на два раздела по 20 ГБ. Windows® именует их дисками C: и D:. На диске C: данными занято 10 ГБ, а на диске D: — 5 ГБ.

Перемещение данных с диска D: на диск C: освобождает второй раздел для установки FreeBSD.

Пример 2. Уменьшение размера существующего раздела

Компьютер с ОС Windows® имеет жесткий диск размером 40 ГБ, на котором создан один большой раздел, занимающий весь жесткий диск.

Windows® именует этот раздел диском C:. На этом разделе данные занимают 15 ГБ. Конечная цель — отвести для

Windows® раздел размером 20 ГБ, а второй раздел размером 20 ГБ задействовать для установки FreeBSD.

Подобное перераспределение можно выполнить одним из двух способов:

1. Сделайте резервную копию данных вашей Windows®. Далее, переустановите Windows®, создав во время инсталляции раздел размером 20 ГБ.
2. Используйте утилиту редактирования разделов (наподобие GParted) для уменьшения раздела Windows®, а в освободившемся пространстве создайте новый раздел для установки FreeBSD.

Разделы диска, содержащие разные операционные системы, делают возможной загрузку по выбору одной из имеющихся операционных систем.

Соберите информацию о сетевых настройках

Некоторым вариантам установки FreeBSD для загрузки файлов необходимо наличие соединения с сетью. Инсталлятор запросит

информацию о подключении для настройки соединения с сетью через интерфейс Ethernet (через кабельный модем или к модему DSL с интерфейсом Ethernet).

Для автоматического конфигурирования сетевых интерфейсов часто применяется протокол DHCP. Если в подключаемой сети сервис DHCP отсутствует, информацию о подключении к необходимо взять у системного администратора или провайдера Интернет.

1. IP адрес
2. Маска подсети
3. IP адрес шлюза по умолчанию
4. Доменное имя локальной сети
5. IP адрес DNS сервера/серверов

Проверьте сведения об обнаруженных ошибках FreeBSD

Хотя проект FreeBSD борется за то, чтобы каждый релиз FreeBSD был настолько стабильным, насколько это возможно, ошибки порой вкрадываются в процесс разработки.

В очень редких случаях эти ошибки влияют на процесс установки. Как только эти проблемы обнаруживаются и исправляются, их описание попадает в сообщения об ошибках FreeBSD, находящиеся на сайте FreeBSD. Проверьте сообщения об ошибках перед установкой и убедитесь, что отсутствуют проблемы, которые могут затронуть установку. Информация о всех релизах, включая сообщения об ошибках каждого релиза, может быть найдена на странице информации о релизах веб сайта FreeBSD.

Подготовка установочного носителя информации

Установка FreeBSD начинается с загрузки компьютера с установочного носителя, будь то CD, DVD или USB флеш-накопитель. Инсталлятор — это не та программа, которую можно запустить из другой операционной системы.

В дополнение к стандартному установочному носителю, который содержит копии всех установочных файлов FreeBSD, также существует вариант, предназначенный исключительно для загрузки и называемый bootonly. Установочный носитель bootonly не

содержит копий инсталляционных файлов, а загружает их из сети во время установки. Поэтому образ bootonly CD гораздо меньше объемом, а также при его использовании загружаются лишь необходимые файлы, тем самым уменьшается нагрузка на сетевое соединение.

Копии образов установочных носителей находятся на веб сайте FreeBSD. Также, в каталоге с файлами установочных образов находится файл CHECKSUM.SHA256 , который понадобится вам для проверки целостности скачанного файла образа. Проверка целостности файла образа производится сравнением контрольных сумм. Для подсчета последних FreeBSD предоставляет sha256(1), другие операционные системы также располагают подобными программами. Сравните полученную контрольную сумму с одной из CHECKSUM.SHA256. Контрольные суммы должны совпасть
полностью. Несовпадение контрольных сумм значит, что файл поврежден и к использованию не пригоден.

CD- и DVD-образы FreeBSD являются загрузочными. Для установки необходим один из них. Запишите образ на CD или DVD диск при помощи программы для записи CD, которая есть в вашей текущей операционной системе. Во FreeBSD запись дисков осуществляется утилитой cdrecord(1) из комплекта sysutils/cdrtools Коллекции Портов. Для создания загрузочного флеш-накопителя выполните следующие шаги:

1. Получение образа для флеш-накопителя

Образы для флеш-накопителя для FreeBSD 9.0-RELEASE и более поздних могут быть скачаны с каталога ISO-IMAGES/ по адресу ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/arch/arch/ISO-IMAGES/ version/FreeBSD-version-RELEASE-arch-memstick.img . Замените arch и version соответственно на архитектуру и номер версии которую вы планируете установить. Например, образы для флеш- накопителей FreeBSD/i386 9.0-RELEASE находятся на ftp://ftp.FreeBSD.org/pub/ FreeBSD/releases/i386/i386/ISO-IMAGES/9.0/FreeBSD-9.0-RELEASE-i386-memstick.img

Имя образа для флеш-накопителя имеет суффикс .img. Каталог ISO-IMAGES/ содержит определённое количество разных образов, и выбор конкретного образа зависит от устанавливаемой версии FreeBSD, а в некоторых случаях — и от аппаратного обеспечения.

2. Запись образа на флеш-накопитель

Процедура 1. Использование FreeBSD для записи образа

Запись образа при помощи dd(1).

Файл .img не является обычным файлом. Это образ всего содержимого флеш-накопителя. Этот файл не может быть просто скопированным подобно обычному файлу, он должен быть записан непосредственно на целевое устройство при помощи dd(1):

3. dd if=FreeBSD-9.0-RELEASE-i386-memstick.img of=/dev/da0 0 bs=64k

Процедура 2. Использование Windows® для записи образа

Получение Image Writer для Windows®

Image Writer для Windows® — это свободно распространяемое приложение, при помощи которого можно корректно записать образ на флеш-накопитель. Скачайте его с https://launchpad.net/win32-image-writer/и сохраните в любую директорию.

Запись образа при помощи Image Writer

Кликните дважды на иконке Win32DiskImager для запуска приложения. Удостоверьтесь, что буква диска, отображаемая в боксе Device , соответствует устройству флеш-накопителя. Кликните на иконке с папкой и выберите образ, который будет записан на флеш- накопитель. Нажмите кнопку [Save] для подтверждения выбора имени файла. Проверьте, что всё верно, а также что нет открытых директорий с флеш-накопителя в других окнах. Когда всё готово, нажмите кнопку [Write] для записи образа на флеш-накопитель.

Начало установки

По умолчанию, установщик не изменяет данные на ваших дисках до тех пор, пока вы не увидите следующее сообщение:

Your changes will now be written to disk. If you have chosen to overwrite existing data, it will be PERMANENTLY ERASED. Are you sure you want to commit your changes?

Установка может быть прервана в любой момент до появления этого предупреждения, при этом содержимое дисков изменено не будет. Если вы обеспокоены тем, что что-то было настроено неверно, то вы можете просто выключить компьютер до этого сообщения, при этом никаких

повреждений существующих данных не произойдет.

Загрузка на i386™ и amd64

1. Если вы подготовили «загрузочный» USB-накопитель, как описано в Разделе 2.3.5, «Подготовка установочного носителя информации», то вставьте его в USB гнездо перед включением компьютера.

Если вы загружаетесь с CDROM, то вам необходимо будет включить компьютер и при первой возможности вставить CD диск.

2. Настройте вашу машину на загрузку с CDROM или с USB, в зависимости от того, какое устройство используется для установки.

Настройки BIOS позволяют выбрать конкретное загрузочное устройство. Большинство систем также предоставляют возможность выбрать загрузочное устройство во время запуска, часто эта возможность активируется по нажатию клавиши F10, F11, F12 или Escape.

3. Если ваш компьютер загружается как обычно и запускает существующую операционную систему, то:

■ Диск не был вставлен заблаговременно. Оставьте его в приводе и попробуйте перезагрузить ваш компьютер.

■ Ранее внесенные изменения в BIOS не сработали. Попробуйте повторить шаг настройки BIOS пока не получите необходимый порядок загрузки.

■ Ваш нынешний BIOS не поддерживает загрузку с имеющегося загрузочного накопителя. В этом случае можно использовать Plop Boot Manager для загрузки более старых машин с CD или USB.

4. FreeBSD начнет загружаться. Если вы загружаетесь с CDROM, вы увидите поток сообщений, подобный следующему (информация о версиях опущена):

```
Booting from CD-ROM...
645MB medium detected
CD Loader 1.2

Building the boot loader arguments
Looking up /BOOT/LOADER... Found
Relocating the loader and the BTX
Starting the BTX loader

BTX loader 1.00 BTX version is 1.02
Consoles: internal video/keyboard
BIOS CD is cd0
BIOS drive C: is disk0
BIOS drive D: is diskl
BIOS 636kB/261056kB available memory

FreeBSD/i386 bootstrap loader. Revision 1.1
```

```
Loading /boot/defaults/loader.conf /boot/kernel/kernel
text=0x64daa0 data=0xa4e80+0xa9e40 o
syms=[0x4+Ox6cacO+Ox4+Ox88e9d] \
```

Рис.1. Поток сообщений при загрузке с CDROM

5. Отображается меню загрузчика FreeBSD:

Рис.2. Меню загрузчика FreeBSD

Выждите десять секунд или нажмите Enter.

Загрузка Macintosh® PowerPC®

На большинстве машин удерживание клавиши C на клавиатуре во время начальной загрузки активирует загрузку с CD. Иначе, удерживайте Command+Option+O+F, или Windows+Alt+O+F на не- Apple® клавиатурах. На приглашение 0 > введите:

boot cd:,\ppc\loader cd:0

Для Xserves без клавиатур, ознакомьтесь с загрузкой в Open Firmware, которая описана на сайте поддержки Apple®.

Загрузка Sparc64®

Большинство систем Sparc64® настроены на автоматическую загрузку с жесткого диска. Для того, чтобы установить FreeBSD, вам потребуется выполнить загрузку по сети или с CDROM, что подразумевает получение доступа к PROM (OpenFirmware).

Для того, чтобы получить доступ к PROM, перегрузите систему и дождитесь появления загрузочных сообщений. Вид сообщений зависит от модели машины, но должен выглядеть подобно следующему:

```
Sun Blade 100 (UltraSPARC-lie), Keyboard Present
Copyright 1998-2001 Sun Microsystems, Inc. All rights reserved.
OpenBoot 4.2, 128 MB memory installed, Serial #51090132.
Ethernet address 0:3:ba:b:92:d4, Host ID: 830b92d4.
```

Рис.4. Загрузочное сообщение

Если ваша система продолжает загружаться с жесткого диска, то чтобы получить приглашение PROM вам необходимо нажать на клавиатуре L1+A или Stop+A, или же послать сигнал BREAK через последовательную консоль (используя, например, ~# в tip(1) или cu(1)).

Приглашение выглядит подобно следующему:

ok (1.)

ok {0} (2.)

1. Приглашение, отображающееся на системах с одним центральным процессором.

2. Приглашение, отображающееся на многопроцессорных (SMP) системах, цифра указывает на количество активных центральных процессоров.

На этом этапе вставьте CDROM в привод и наберите boot cdrom в приглашении PROM.

Просмотр результата определения устройств (device probe)

Выводимые на экран во время начальной загрузки системы последние пару сотен строк сохраняются, и при необходимости могут быть просмотрены. Чтобы просмотреть содержимое буфера, нажмите Scroll Lock. Это включит режим буфера прокрутки. Далее, для просмотра сохраненных сообщений вы можете использовать клавиши навигации или клавиши PageUp и PageDown. Чтобы выйти из режима просмотра буфера нажмите еще раз Scroll Lock.

Включите прокрутку экранного буфера и просмотрите сообщения, которые были вытеснены с экрана во время определения устройств ядром. Далее будет представлен типичный вывод сообщений определения устройств, однако его содержимое будет отличаться в зависимости от комплекта устройств, установленных в ваш компьютер.

Copyright (c) 1992-2011 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994 The Regents of the University of California. All rights reserved.
FreeBSD is a registered trademark of The FreeBSD Foundation.
FreeBSD 9.0-RELEASE #0 r225473M: Sun Sep 11 16:07:30 BST 2011
root@psi:/usr/obj/usr/src/sys/GENERIC amd64
CPU: Intel(R) Core(TM)2 Duo CPU T9400 @ 2.53GHz (2527.05-MHz K8- class CPU)
Origin = "GenuineIntel" Id = 0x10676 Family = 6 Model = 17 Stepping = 6

Features=0xbfebfbff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,APIC,SEP ,MTRR,PGE,MCA,CMOV,PAT,PSE36,CLFLUSH,DTS,ACPI,MM
X,FXSR,SSE,SSE2,SS,HTT,TM,PBE>
Features2=0x8e3fd<SSE3,DTES64,MON,DS_CPL,VMX,SMX,EST,TM2,SSSE3 ,CX16,xTPR,PDCM,SSE4.1>
AMD Features=0x20100800<SYSCALL,NX,LM>
AMD Features2=0x1<LAHF>
TSC: P-state invariant, performance statistics

real memory = 3221225472 (3072 MB)
avail memory = 2926649344 (2791 MB)
Event timer "LAPIC" quality 400
ACPI APIC Table: <TOSHIB A0064 >

FreeBSD/SMP: Multiprocessor System Detected: 2 CPUs
FreeBSD/SMP: 1 package(s) x 2 core(s)
cpu0 (BSP): APIC ID: 0
cpu1 (AP): APIC ID: 1
ioapic0: Changing APIC ID to 1
ioapic0 <Version 2.0> irqs 0-23 on motherboard
kbd1 at kbdmux0
acpi0: <TOSHIB A0064> on motherboard
acpi0: Power Button (fixed)
acpi0: reservation of 0, a0000 (3) failed
acpi0: reservation of 100000, b6690000 (3) failed
Timecounter "ACPI-safe" frequency 3579545 Hz quality 850
acpi_timer0: <24-bit timer at 3.579545MHz> port 0xd808-0xd80b on acpi0
cpu0: <ACPI CPU> on acpi0
ACPI Warning: Incorrect checksum in table [ASF!] - 0xFE, should be 0x9A (20110527/tbutils-282)
cpu1: <ACPI CPU> on acpi0
pcib0: <ACPI Host-PCI bridge> port 0xcf8-0xcff on acpi0
pci0: <ACPI PCI bus> on pcib0
vgapci0: <VGA-compatible display> port 0xcff8-0xcfff mem 0xff400000-0xff7fffff,0xe0000000-0xefffffff irq 16 at device 2.0 on pci0
agp0: <Intel GM45 SVGA controller> on vgapci0
agp0: aperture size is 256M, detected 131068k stolen memory
vgapci1: <VGA-compatible display> mem 0xffc00000-0xffcfffff at device 2.1 on pci0
pci0: <simple comms> at device 3.0 (no driver attached)
em0: <Intel(R) PRO/1000 Network Connection 7.2.3> port 0xcf80-0xcf9f mem 0xff9c0000-0xff9dffff,0xff9fe000-0xff9fefff irq 20 at device 25.0 on pci0
em0: Using an MSI interrupt
em0: Ethernet address: 00:1c:7e:6a:ca:b0

uhci0: <Intel 82801I (ICH9) USB controller> port 0xcf60-0xcf7f irq 16 at device 26.0 on pci0
usbus0: <Intel 82801I (ICH9) USB controller> on uhci0
uhci1: <Intel 82801I (ICH9) USB controller> port 0xcf40-0xcf5f irq 21 at device

26.1 on pci0

usbus1: <Intel 82801I (ICH9) USB controller> on uhci1

uhci2: <Intel 82801I (ICH9) USB controller> port 0xcf20-0xcf3f irq 19 at device 26.2 on pci0

usbus2: <Intel 82801I (ICH9) USB controller> on uhci2

ehci0: <Intel 82801I (ICH9) USB 2.0 controller> mem 0xff9ff800-0xff9ffbff irq 26.3 at device 26.7 on pci0

usbus3: EHCI version 1.0

usbus3: <Intel 82801I (ICH9) USB 2.0 controller> on ehci0

hdac0: <Intel 82801I High Definition Audio Controller> mem 0xff9f8000-0xff9fbfff irq 22 at device 27.0 on pci0

pcib1: <ACPI PCI-PCI bridge> irq 17 at device 28.0 on pci0

pci1: <ACPI PCI bus> on pcib1

iwn0: <Intel(R) WiFi Link 5100> mem 0xff8fe000-0xff8fffff irq 16 at device 0.0 on pci1

pcib2: <ACPI PCI-PCI bridge> irq 16 at device 28.1 on pci0

pci2: <ACPI PCI bus> on pcib2

pcib3: <ACPI PCI-PCI bridge> irq 18 at device 28.2 on pci0

pci4: <ACPI PCI bus> on pcib3

pcib4: <ACPI PCI-PCI bridge> at device 30.0 on pci0

pci5: <ACPI PCI bus> on pcib4

cbb0: <RF5C476 PCI-CardBus Bridge> at device 11.0 on pci5

cardbus0: <CardBus bus> on cbb0

pccard0: <16-bit PCCard bus> on cbb0

isab0: <PCI-ISA bridge> at device 31.0 on pci0

isa0: <ISA bus> on isab0

ahci0: <Intel ICH9M AHCI SATA controller> port 0x8f58-0x8f5f,0x8f54-0x8f57,0x8f48-0x8f4f,0x8f44-0x8f47,0x8f20-0x8f3f mem 0xff9fd800-0xff9fdfff irq 19 at device 31.2 on pci0

ahci0: AHCI v1.20 with 4 3Gbps ports, Port Multiplier not supported

ahcich0: <AHCI channel> at channel 0 on ahci0

ahcich1: <AHCI channel> at channel 1 on ahci0

ahcich2: <AHCI channel> at channel 4 on ahci0

acpi_lid0: <Control Method Lid Switch> on acpi0

battery0: <ACPI Control Method Battery> on acpi0

acpi_button0: <Power Button> on acpi0

acpi_acad0: <AC Adapter> on acpi0

acpi_toshiba0: <Toshiba HCI Extras> on acpi0

acpi_tz0: <Thermal Zone> on acpi0

attimer0: <AT timer> port 0x40-0x43 irq 0 on acpi0

Timecounter "i8254" frequency 1193182 Hz quality 0
Event timer "i8254" frequency 1193182 Hz quality 100
atkbdc0: <Keyboard controller (i8042)> port 0x60,0x64 irq 1 on acpi0
atkbd0: <AT Keyboard> irq 1 on atkbdc0
kbd0 at atkbd0
atkbd0: [GIANT-LOCKED]
psm0: <PS/2 Mouse> irq 12 on atkbdc0
psm0: [GIANT-LOCKED]
psm0: model GlidePoint, device ID 0
atrtc0: <AT realtime clock> port 0x70-0x71 irq 8 on acpi0
Event timer "RTC" frequency 32768 Hz quality 0
hpet0: <High Precision Event Timer> iomem 0xfed00000-0xfed003ff on acpi0 Timecounter "HPET" frequency 14318180 Hz quality 950 Event timer "HPET" frequency 14318180 Hz quality 450
Event timer "HPET1" frequency 14318180 Hz quality 440
Event timer "HPET2" frequency 14318180 Hz quality 440
Event timer "HPET3" frequency 14318180 Hz quality 440
uart0: <16550 or compatible> port 0x3f8-0x3ff irq 4 flags 0x10 on acpi0

sc0: <System console> at flags 0x100 on isa0
sc0: VGA <16 virtual consoles, flags=0x300>
vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0
ppc0: cannot reserve I/O port range
est0: <Enhanced SpeedStep Frequency Control> on cpu0
p4tcc0: <CPU Frequency Thermal Control> on cpu0
est1: <Enhanced SpeedStep Frequency Control> on cpu1
p4tcc1: <CPU Frequency Thermal Control> on cpu1
Timecounters tick every 1.000 msec
hdac0: HDA Codec #0: Realtek ALC268
hdac0: HDA Codec #1: Lucent/Agere Systems (Unknown)
pcm0: <HDA Realtek ALC268 PCM #0 Analog> at cad 0 nid 1 on hdac0
pcm1: <HDA Realtek ALC268 PCM #1 Analog> at cad 0 nid 1 on hdac0
usbus0: 12Mbps Full Speed USB v1.0
usbus1: 12Mbps Full Speed USB v1.0
usbus2: 12Mbps Full Speed USB v1.0
usbus3: 480Mbps High Speed USB v2.0
ugen0.1: <Intel> at usbus0
uhub0: <Intel UHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus0
ugen1.1: <Intel> at usbus1
uhub1: <Intel UHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus1

ugen2.1: <Intel> at usbus2
uhub2: <Intel UHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus2
ugen3.1: <Intel> at usbus3
uhub3: <Intel EHCI root HUB, class 9/0, rev 2.00/1.00, addr 1> on usbus3
uhub0: 2 ports with 2 removable, self powered
uhub1: 2 ports with 2 removable, self powered
uhub2: 2 ports with 2 removable, self powered
uhub3: 6 ports with 6 removable, self powered
ugen2.2: <vendor 0x0b97> at usbus2
uhub8: <vendor 0x0b97 product 0x7761, class 9/0, rev 1.10/1.10, addr 2> on usbus2
ugen1.2: <Microsoft> at usbus1
ada0 at ahcich0 bus 0 scbus1 target 0 lun 0
ada0: <Hitachi HTS543225L9SA00 FBEOC43C> ATA-8 SATA 1.x device
ada0: 150.000MB/s transfers (SATA 1.x, UDMA6, PIO 8192bytes)
ada0: Command Queueing enabled
ada0: 238475MB (488397168 512 byte sectors: 16H 63S/T 16383C)
ada0: Previously was known as ad4
ums0: <Microsoft Microsoft 3-Button Mouse with IntelliEyeTM, class 0/0, rev 1.10/3.00, addr 2> on usbus1
SMP: AP CPU #1 Launched!
cd0 at ahcich1 bus 0 scbus2 target 0 lun 0
cd0: <TEAC DV-W28S-RT 7.0C> Removable CD-ROM SCSI-0 device
cd0: 150.000MB/s transfers (SATA 1.x, ums0: 3 buttons and [XYZ] coordinates ID=0
UDMA2, ATAPI 12bytes, PIO 8192bytes)
cd0: cd present [1 x 2048 byte records]
ugen0.2: <Microsoft> at usbus0
ukbd0: <Microsoft Natural Ergonomic Keyboard 4000, class 0/0, rev 2.00/1.73, addr 2> on usbus0 kbd2 at ukbd0
uhid0: <Microsoft Natural Ergonomic Keyboard 4000, class 0/0, rev 2.00/1.73, addr 2> on usbus0 Trying to mount root from cd9660:/dev/iso9660/FREEBSD_INSTALL [ro]...

Внимательно просмотрите вывод определения устройств и убедитесь, что FreeBSD обнаружила все ожидаемые вами устройства. Если устройство не было найдено, то оно не будет упомянуто в выводе. Модули ядра позволяют вам добавить поддержку устройств, драйвера которых отсутствуют в ядре GENERIC.

После процедуры определения устройств вы увидите Рисунке 3, «Выбор вариантов работы установочного носителя». Установочный носитель может использоваться одним из трёх способов: для установки FreeBSD, как Live CD, или просто для доступа к оболочке FreeBSD. Используйте клавиши навигации для выбора опции, а Enter — для подтверждения выбора.

Рис.5. Выбор вариантов работы установочного носителя

Выбор опции [Install] вызовет программу-установщик.

Введение в bsdinstall

bsdinstall это текстовая программа для установки FreeBSD, созданная Nathan Whitehorn <nwhitehorn@FreeBSD.org> и представленная в 2011 году для FreeBSD 9.0.

Примечание

В комплекте с PC-BSD есть программа pc-sysinstall от Kris Moore <kmoore@FreeBSD.org>, которая также может использоваться для установки FreeBSD. Несмотря на то, что эту программу путают с bsdinstall, обе они между собой никак не связаны.

Система меню bsdinstall контролируется клавишами навигации, а также Enter, Tab, Space и другими.

Выбор раскладки клавиатуры (Keymap)

В зависимости от используемой системной консоли, bsdinstall может предложить выбрать отличную от настроенной по умолчанию раскладку клавиатуры.

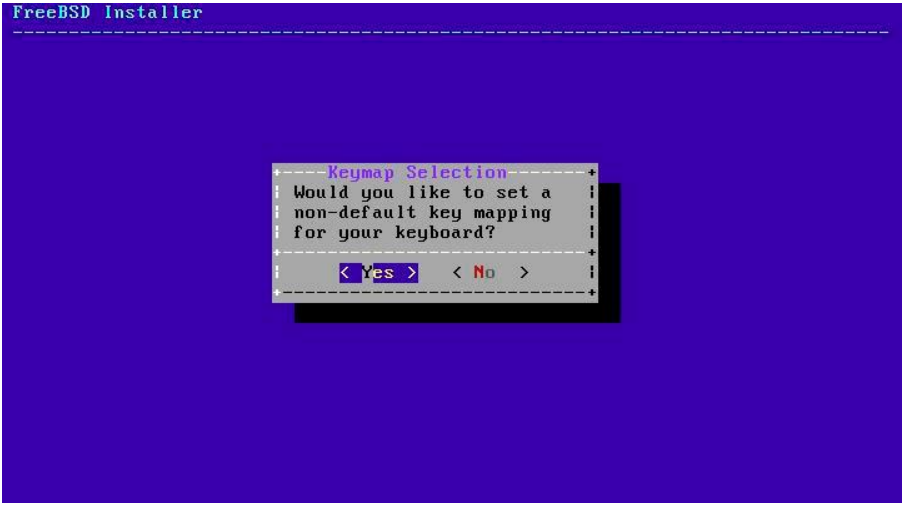

Рис.6. Выбор раскладки клавиатуры

Если нажата кнопка [YES], отобразится следующее меню выбора раскладки клавиатуры. Иначе, это меню выбора отображено не будет, а будет использоваться раскладка клавиатуры по умолчанию.

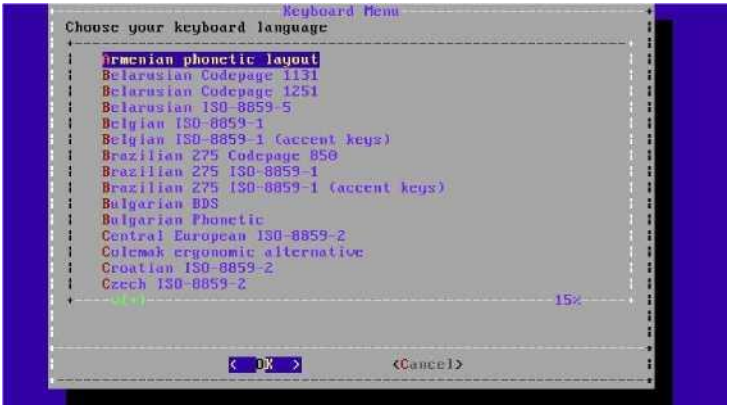

Рис.7. Меню выбора раскладки клавиатуры

Используя клавиши навигации и клавишу Enter, выберите раскладку, которая наиболее близко соответствует клавиатуре, подключенной к системе.

Примечание

Нажатие Esc приведет к выбору раскладки по умолчанию. Выбор опции United States of America ISO-8859-1 тоже является безопасным в том случае, если возникают трудности с определением раскладки.

Установка имени хоста

Далее, bsdinstall предложит указать имя хоста для устанавливаемой системы.

Рис.8. Установка имени хоста

Вводимое имя хоста должно быть полным (fully-qualified), например: machine3.example.com.

Выбор устанавливаемых компонентов

Далее, bsdinstall предложит выбрать дополнительные компоненты для установки.

Рис.9. Выбор устанавливаемых компонентов

Определение перечня компонентов для установки в наибольшей мере зависит от планируемого использования системы и от количества доступного дискового пространства. Ядро и набор утилит FreeBSD (вместе называемые «базовой системой») устанавливаются всегда.

В зависимости от типа установки, некоторые из следующих компонентов могут не появляться.

- doc - Дополнительная документация, преимущественно исторического характера. Документация, предоставляемая Проектом Документирования FreeBSD может быть установлена позже.
- games - Несколько традиционных игр BSD, в том числе fortune, rot13, и другие.
- lib32 - Библиотеки совместимости для запуска 32-битных приложений на 64-битных версиях FreeBSD.
- ports - Коллекция Портов FreeBSD.

  Коллекция Портов — это простой и удобный способ установки программ. Она не содержит исходных кодов, необходимых для компиляции приложений. Коллекция Портов — это множество файлов, при помощи которого автоматизируется загрузка, компиляция и установка программных пакетов сторонних разработчиков.
- src - Исходный код системы.

FreeBSD распространяется с полным исходным кодом как для ядра,

так и для программ базовой системы. Для большинства приложений исходный код системы не нужен, однако он может потребоваться при построении некоторых программ, распространяемых в виде исходных кодов (например, драйверов или модулей ядра), или для разработки FreeBSD.

Полное дерево исходных кодов требует 1 ГБ дискового пространства, пересборка всей системы FreeBSD требует дополнительно 5 ГБ пространства.

Установка по сети

Установочный носитель bootonly не содержит копий установочных файлов. В случае использования такого носителя необходимые файлы должны быть получены загрузкой из сети.

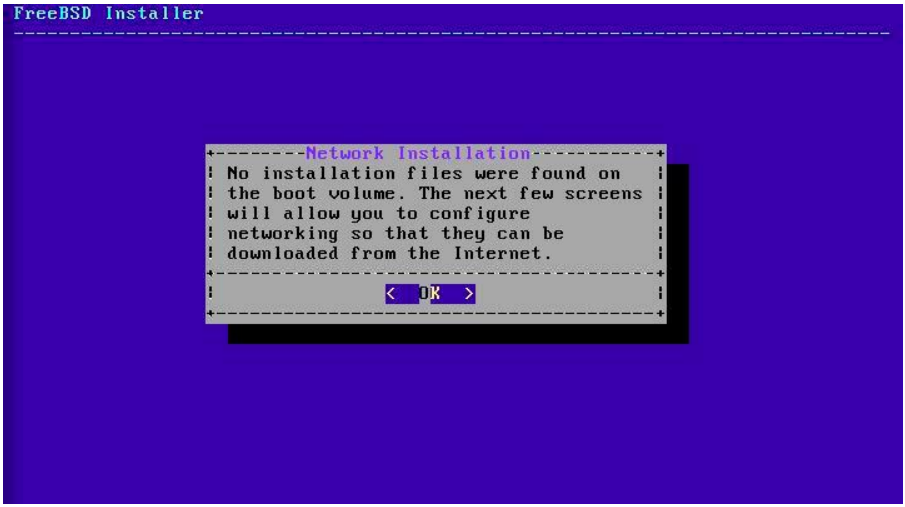

Рис.10. Установка по сети

После настройки сетевого соединения, которая детально описана в лабораторной работе №3, выбирается зеркало сайта. Зеркала сайта содержат копии файлов FreeBSD. Выберите зеркало, размещенное в 35

том регионе мира, что и компьютер, на который устанавливается FreeBSD. Если зеркало расположено ближе к целевому компьютеру, то файлы могут быть получены быстрее, тем самым уменьшится время установки.

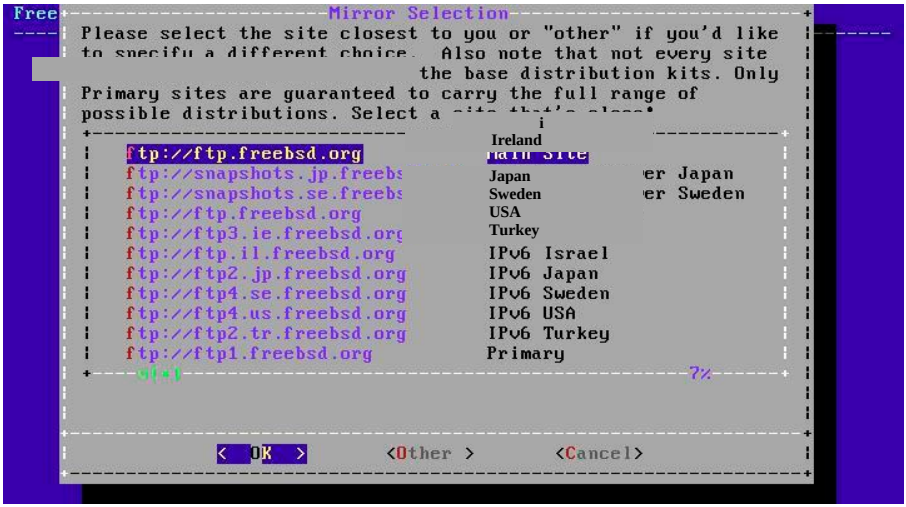

Рис.11. Выбор зеркала сайта

Дальнейший сценарий одинаков для всех способов установки.

Выделение дискового пространства

Есть три способа осуществить разбиение дискового пространства для FreeBSD. Шаблонное (guided) разбиение автоматически настраивает разделы диска, ручное (manual) разбиение позволяет опытным пользователям создавать разделы согласно своим требованиям. И наконец, есть возможность вызвать командный интерпретатор, в котором можно будет непосредственно запускать утилиты наподобие gpart, fdisk и bsdlabel.

Рис.12. Выбор способа разбиения: шаблонное (guided), ручное (manual), вызов командного интерпретатора(shell)

Шаблонное (guided) разбиение

Если в системе есть несколько дисков, то выберите один, на который будет устанавливаться FreeBSD.

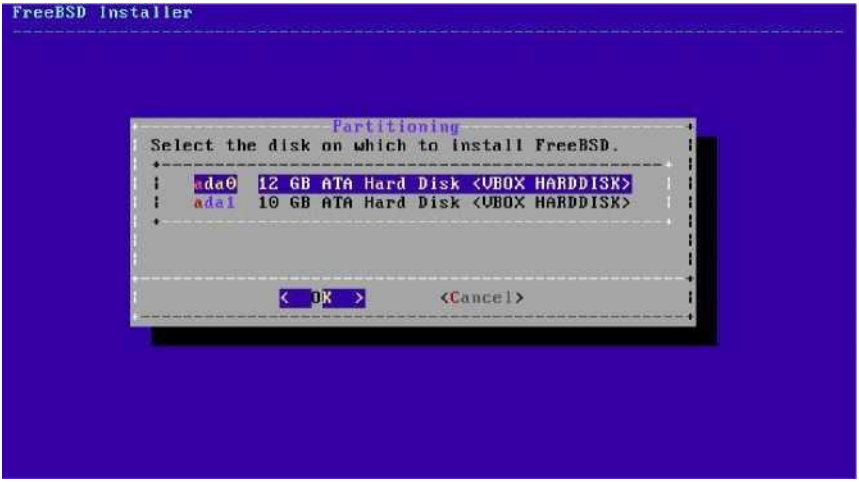

Рис.13. Выбор из множества дисков

Для FreeBSD может быть выделен весь диск или только его часть. Если выбирается [Entire Disk], то создается стандартное разбиение, занимающее весь диск. Выбрав [Partition], вы получите создание разделов в неиспользуемой области диска.

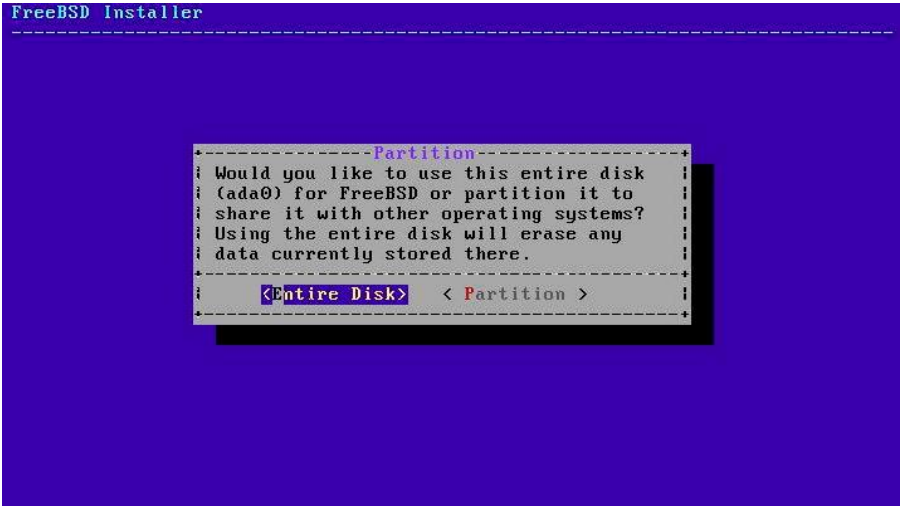

Рис.14. Выбор всего диска или раздела

По завершении разбиения дискового пространства внимательно просмотрите результат. Если была допущена ошибка, то вам предоставляется возможность либо вернуть конфигурацию к исходному состоянию нажав [Revert], либо выполнить автоматическое переразбиение выбрав [Auto]. Также разделы могут быть созданы, изменены или удалены вручную. Если результат разбиения корректен, выберите [Finish] для продолжения установки.

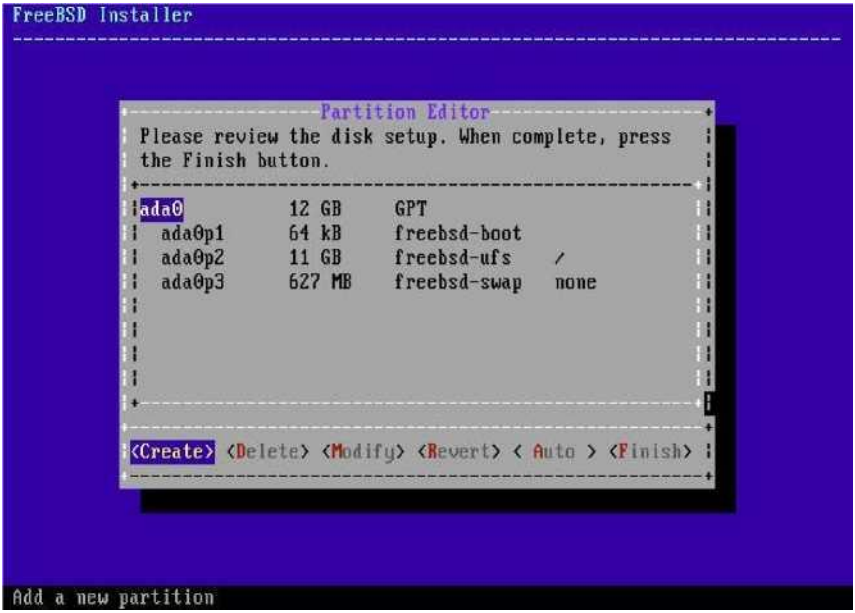

Рис.15. Просмотр созданных разделов

Ручное (manual) разбиение

Ручное разбиение начинается с редактора разделов.

Рис.16. Ручное создание разделов

Перемещение подсвечивания на имя устройства (в этом примере — ada0) и выбор [Create] приведет вас к меню с перечнем схем разбиения.

Рис.17. Выбор схемы разбиения

Как правило, схема GPT является наиболее подходящей для PC- совместимых компьютеров. Для более старых операционных систем, которые несовместимы с GPT, может потребоваться разбиение MBR. Остальные схемы разбиения, в общем, используются для нераспространенных или старых компьютерных систем.

Таблица 1. Схемы разбиения

Аббревиатура	Описание
APM	Apple Partition Map, используемая на PowerPC Macintosh.
BSD	Метки BSD без MBR, иногда называемые «dangerously dedicated mode». За подробностями обратитесь к bsdlabel(8)
GPT	Таблица разделов GUID

Продолжение таблицы 1

Аббревиатура	Описание
MBR	Master Boot Record
PC98	Разновидность MBR, используемая компьютерами NET PC-98
VTOC8	Volume Table Of Contents, используемая компьютерами Sun SPARC64 и UltraSPARC.

После того, как схема разбиения определена, повторный выбор [Create] приводит к созданию новых разделов диска.

Рис.18. Создание нового раздела

Стандартная установка FreeBSD со схемой GPT создаст как минимум три раздела:

- freebsd-boot - загрузочный код FreeBSD.
- freebsd-ufs - файловая система UFS FreeBSD.
- freebsd-swap - FreeBSD область подкачки.

Разумеется, возможно создание большего количества разделов с файловыми системами, и некоторые пользователи предпочитают выделять отдельные разделы для таких файловых систем, как /, /var, /tmp, и /usr. Иллюстрация подобного разбиения приведена в Пример 3, «Создание традиционного разбиения под файловые системы.».

При указании размеров допускается использование общепринятых аббревиатур, таких как K для килобайт, M для мегабайт, или G для гигабайт.

Подсказка

Должное выравнивание секторов обеспечивает наилучшую производительность, а создание разделов с размерами, кратными 4 Кбайт, помогает обеспечить правильное выравнивание как на дисках с размером сектора 512 байт, так и на устройствах с размером сектора 4 Кбайт. В общем, задание размеров, кратных 1 Мбайт или 1 Гбайт— это наиболее простой способ выполнить выравнивание начал разделов на позицию, кратную 4 Кбайт. Исключение: на данный момент размер раздела freebsd-boot не должен превышать 512 Кбайт из-за ограничений загрузочного кода.

В случае, если раздел будет содержать файловую систему, ей потребуется точка монтирования. Если планируется создать единственный раздел UFS, то точка монтирования должна быть /.

Также будет запрошена метка. Метка — это имя, присвоенное разделу. Имя устройства или его номер может измениться если устройство будет подключено к другому контроллеру или порту, а метка раздела останется неизменной. Ссылки на метки вместо имён устройств и номеров разделов в файлах типа /etc/fstab делают систему более толерантной к замене оборудования. Метки GPT появляются после подключения диска вкаталоге /dev/gpt/ . У других схем разбиения есть свои особенности поддержки меток, и их метки располагаются в других подкаталогах каталога /dev/.

Подсказка

Во избежание конфликтов имен меток используйте уникальные имена для каждой файловой системы. Несколько букв, взятых от имени компьютера, его назначения или размещения может быть добавлено к метке. Например, корневому разделу UFS для компьютера в лаборатории можно присвоить метку labroot или rootfs-lab.

Пример 3. Создание традиционного разбиения под файловые системы.

Для традиционного разбиения, в котором каталоги /, /var, /tmp и /usr представляют собой отдельные файловые системы на их собственных разделах, создайте схему разбиения GPT, потом создайте разделы, как это указано ниже. Показанные размеры разделов являются типичными для жесткого диска размером 20Гб. Если диск большего размера, то будет

уместным отвести больше места для раздела подкачки или для раздела с файловой системой /var. Задействованные в этом примере метки имеют префикс ex, от слова «example», вам же рекомендуется использовать другие уникальные имена меток.

По умолчанию, загрузчик gptboot FreeBSD ожидает, что первый найденный раздел UFS будет корневым разделом (/).

Таблица 2. Пример традиционного разбиения под файловые системы

Тип раздела	**Размер**	**Точка монтирования**	**Метка**
freebsd-bool	512K		
freebsd-ufs	2G	/	exrootfs
freebsd- swap	4G		exswap
freebsd-ufs	2G	/var	exvarfs
freebsd-ufs	1G	/tmp	extmpfs
freebsd-ufs	Соглашайтесь со значением по умолчанию (оставшаяся часть объема диска)	/usr	exusrfs

Для продолжения установки по завершении создания необходимых разделов выберите [Finish].

Разбиение с использованием FDisk

При запуске FDisk будет показан список всех жестких дисков, обнаруженных ядром во время тестирования устройств. Рисунок 18. показывает пример системы с двумя IDE дисками. Они были названы ad0 и ad2.

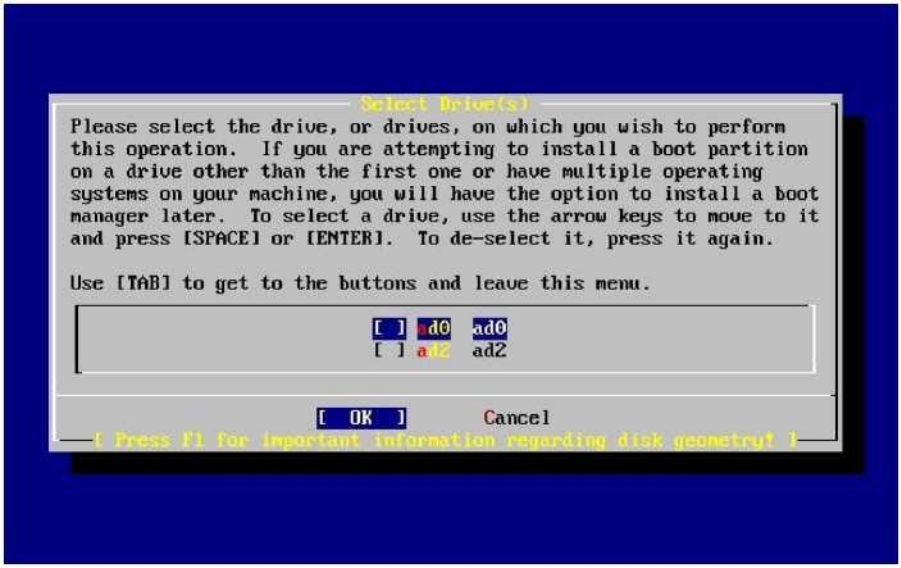

Рис.19. Выберите диск для FDisk

Вы можете быть удивлены, почему устройства ad1 здесь нет. Почему оно было пропущено?

Предположим, что у вас есть два жестких диска IDE, один master на первом контроллере IDE, а второй master на втором контроллере IDE. Если FreeBSD пронумерует их в том порядке, в котором нашла, ad0 и ad1, все будет работать.

Но если вы добавите третий диск, как slave устройство на первый контроллер IDE, он станет ad1, а предыдущий ad1 станет ad2. Поскольку имена устройств (таких как ad1s1a) используются для обращения к файловым системам, вы можете вдруг обнаружить, что некоторые из ваших файловых систем больше не отображаются правильно и вам потребуется изменить конфигурацию FreeBSD.

Для обхода этой проблемы, ядро может быть настроено так, чтобы именовать IDE диски на основе их местоположения, а не порядка, в котором они были найдены. С этой схемой master диск на втором контроллере IDE будет всегда устройством ad2, если даже нет устройств ad0 или ad1.

Это конфигурация ядра FreeBSD по умолчанию, поэтому на экране

показаны ad0 и ad2. У компьютера, с которого был взят этот снимок экрана, есть по одному IDE диску на обоих master каналах IDE контроллеров и ни одного диска на каналах slave.

Вы должны выбрать диск, на который хотите установить FreeBSD, и нажать [OK].

Запустившийся fdisk будет выглядеть примерно, как Рисунок 19.

Экран FDisk разбит на три раздела.

Первый раздел, занимающая первые две линии экрана, показывает подробную информацию о выбранном в данный момент диске, включая его имя во FreeBSD, геометрию и общий размер диска.

Второй раздел показывает имеющиеся в данный момент на диске слайсы, где они начинаются и заканчиваются, их размер, имя, которое им дала FreeBSD, описание и подтип. На этом примере показаны два маленьких неиспользованных слайса, которые являются артефактами схемы разметки диска на PC. Также показан один большой FAT слайс, который почти всегда является диском C: в MS-DOS / Windows, и дополнительный слайс, который может содержать диски с другими буквами для MS-DOS / Windows. Третий раздел показывает команды, доступные в FDisk.

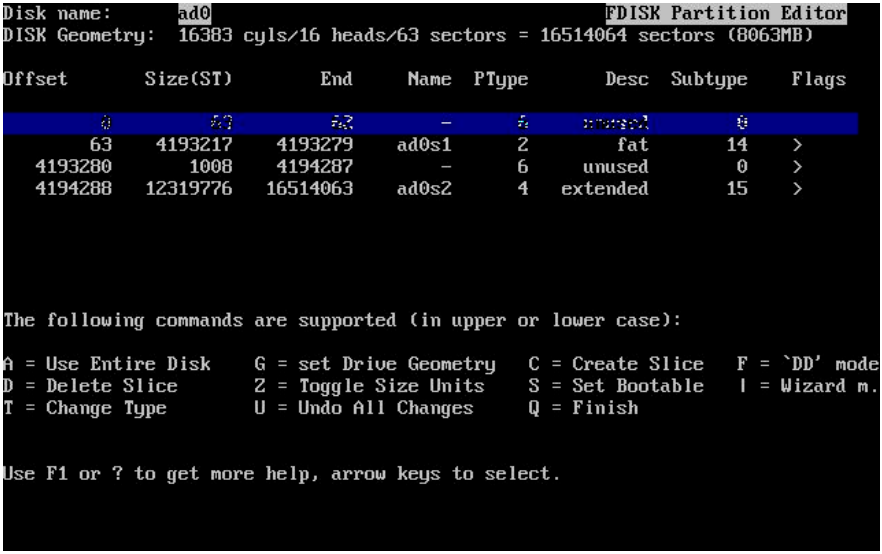

Рис.20. Типичные разделы fdisk перед редактированием

Ваши действия теперь будут зависеть от того, как вы хотите разбить диск на слайсы.

Если вы хотите использовать для FreeBSD весь диск, нажмите A, что соответствует опции Использовать весь диск (Use Entire Disk). Существующие слайсы будут удалены, и заменены на небольшую область, помеченную как неиспользуемая (unused) (это опять же артефакт разметки диска PC), и один большой слайс для FreeBSD. Когда вы сделаете это, нужно выбрать вновь созданный слайс FreeBSD используя клавиши навигации, а затем нажать S, чтобы сделать слайс загрузочным. Экран будет похож на Рис. 14. Обратите внимание, что A в колонке Flags означает, что слайс активен и с него будет происходить загрузка.

Если вы будете удалять существующий слайс для освобождения места под FreeBSD, выберите слайс, используя клавиши навигации, и нажмите D. Затем можете нажать C, и получить приглашение на ввод размера слайса, который вы хотите создать. Введите соответствующее значение и нажмите Enter. Значение по умолчанию в этом поле означает наибольший размер слайса, который может быть выбран; это может быть наибольший непрерывный блок неразмеченного пространства или размер всего жесткого диска.

Если вы уже освободили место для FreeBSD (возможно, используя утилиту вроде PartitionMagic®), можете нажать C для создания нового слайса. Будет также предложено ввести размер слайса, который вы хотите создать.

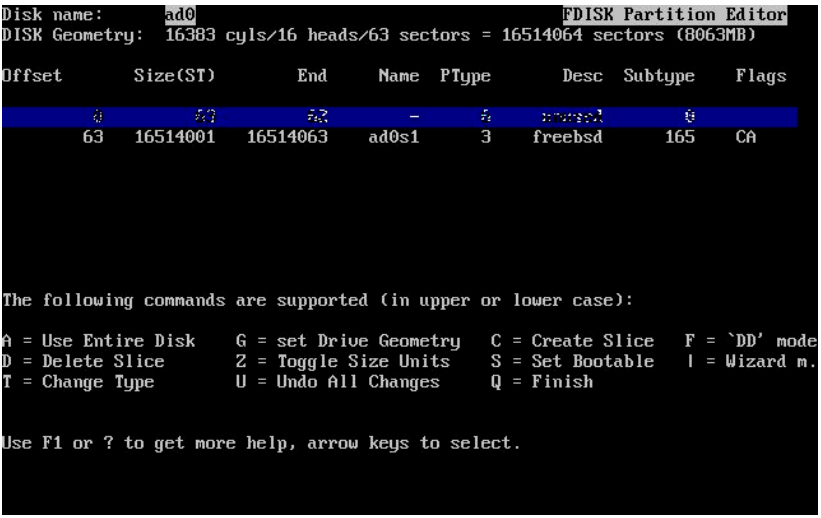

Рис.21. Разбиение в Fdisk с использованием всего диска

Когда закончите, нажмите Q.

Завершение установки

Следующий шаг — ваш последний шанс прервать установку и предотвратить изменение данных на жестком диске.

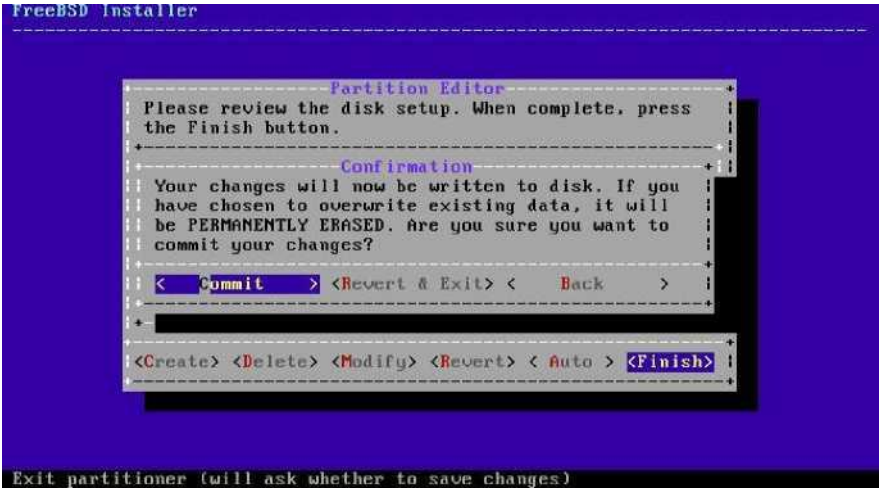

Рис.22. Заключительное подтверждение

Для продолжения выберите [Commit]. Если необходимо внести изменения, для возвращения к редактору разделов нажмите [Back]. Выбор [Revert & Exit] дает возможность выйти из установщика без внесения изменений на жесткий диск.

Продолжительность установки варьируется в зависимости от выбранного дистрибутива, способа установки и быстродействия компьютера. Далее последует очередь сообщений, информирующих о ходе установки.

Первым делом установщик запишет информацию о разделах на диск и отформатирует разделы посредством newfs. Если выполняется установка по сети, то bsdinstall продолжит загрузку необходимых файлов дистрибутива.

Рис.23. Загрузка файлов дистрибутива

Далее последует проверка целостности файлов дистрибутива, чтобы удостовериться, что они не были повреждены во время загрузки или чтения с установочного носителя.

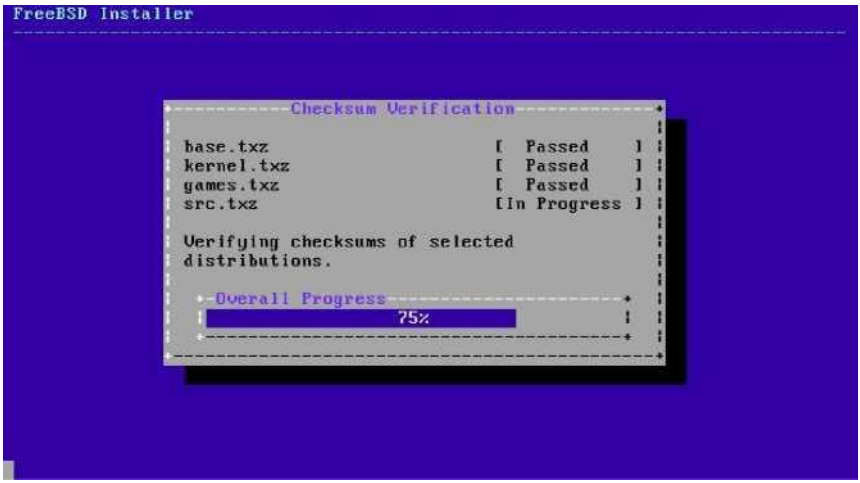

Рис.24. Проверка файлов дистрибутива

И в заключение, проверенные файлы распаковываются на диск.

Рис.25. Извлечение файлов дистрибутива

Как только запрошенные файлы дистрибутива распакуются, bsdinstall приступит к выполнению послеустановочных конфигурационных задач

После установки

После успешной установки FreeBSD последуют меню настройки различных опций. Настройки опций могут быть изменены путем повторного входа в соответствующие разделы финального конфигурационного меню перед загрузкой в свежеустановленную систему FreeBSD.

Установка пароля пользователя root

Установка пароля пользователя root — обязательна. Заметьте, что во время ввода пароля набираемые символы не отображаются на экране. После ввода будет запрошен повторный ввод пароля. Это помогает предотвратить опечатки при наборе.

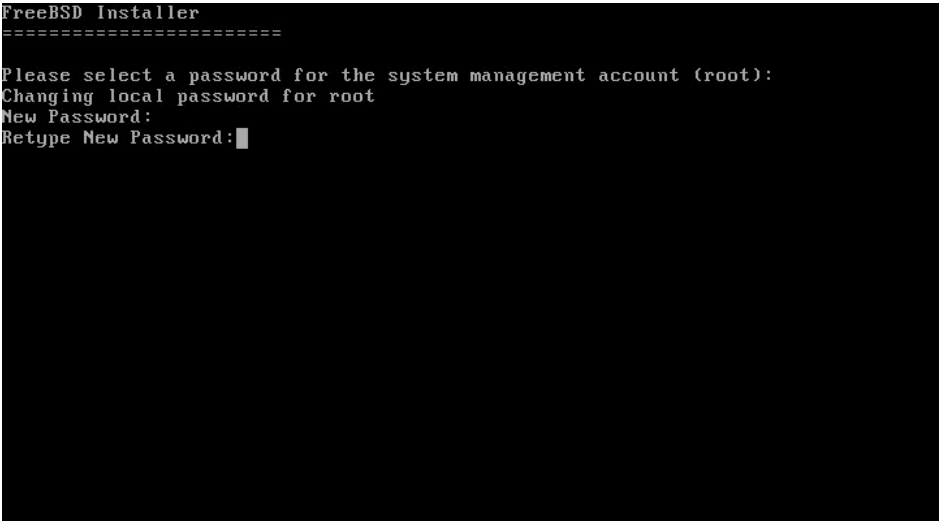

Рис.65. Установка пароля пользователя root

Настройки опций продолжатся после успешной установки пароля.

Загрузка и завершение работы FreeBSD

Во время загрузки FreeBSD отображается множество информационных сообщений. Большинство из них вытеснится за пределы экрана; это нормально. По завершении загрузки системы будет отображено приглашение ко входу (login prompt). Сообщения, которые переместились за пределы экрана, могут быть просмотрены: при нажатии Scroll-Lock включается режим буфера прокрутки. Клавиши PgUp, PgDn, а также клавиши навигации могут быть задействованы для прокручивания буфера. Повторное нажатие ScrollLock разблокирует дисплей и вернет его в нормальный режим.

На приглашение login: введите добавленное во время установки имя пользователя, в этом примере — asample . За исключением случаев крайней необходимости избегайте входа под учетной записью root.

Упомянутый выше буфер прокрутки ограничен в размере, поэтому в него могут умещаться не все сообщения. После входа в систему большинство из них можно просмотреть подав команду dmesg | less из командной строки. Для возврата к командной строке после просмотра

сообщений нажмите q. Типичные сообщения загрузки (информация о версиях опущена):

Copyright (c) 1992-2011 The FreeBSD Project.
Copyright (c) 1979, 1980, 1983, 1986, 1988, 1989, 1991, 1992, 1993, 1994
The Regents of the University of California. All rights reserved.
FreeBSD is a registered trademark of The FreeBSD Foundation.
root@farrell.cse.buffalo.edu:/usr/obj/usr/src/sys/GENERIC amd64
CPU: Intel(R) Core(TM)2 Duo CPU E8400 @ 3.00GHz (3007.77-MHz K8-class CPU) Origin = "GenuineIntel" Id = 0x10676 Family = 6 Model = 17 Stepping = 6
Features=0x783fbff<FPU,VME,DE,PSE,TSC,MSR,PAE,MCE,CX8,APIC,SEP, MTRR,PGE,MCA,CMOV,PAT,PSE36,MMX,FXSR,SSE,SSE2>
Features2=0x209<SSE3,MON,SSSE3>
AMD Features=0x20100800<SYSCALL,NX,LM>
AMD Features2=0x1<LAHF>
real memory = 536805376 (511 MB)
avail memory = 491819008 (469 MB)
Event timer "LAPIC" quality 400
ACPI APIC Table: <VBOX VBOXAPIC>
ioapic0: Changing APIC ID to 1
ioapic0 <Version 1.1> irqs 0-23 on motherboard
kbd1 at kbdmux0
acpi0: <VBOX VBOXXSDT> on motherboard
acpi0: Power Button (fixed)
acpi0: Sleep Button (fixed)
Timecounter "ACPI-fast" frequency 3579545 Hz quality 900
acpi_timer0: <32-bit timer at 3.579545MHz> port 0x4008-0x400b on acpi0
cpu0: <ACPI CPU> on acpi0

pcib0: <ACPI Host-PCI bridge> port 0xcf8-0xcff on acpi0
pci0: <ACPI PCI bus> on pcib0
isab0: <PCI-ISA bridge> at device 1.0 on pci0
isa0: <ISA bus> on isab0
atapci0: <Intel PIIX4 UDMA33 controller> port 0x1f0-0x1f7,0x3f6,0x170-0x177,0x376,0xd000-0xd00f at device 1.1 on pci0
ata0: <ATA channel 0> on atapci0
ata1: <ATA channel 1> on atapci0
vgapci0: <VGA-compatible display> mem 0xe0000000-0xe0ffffff irq 18 at device 2.0 on pci0
em0: <Intel(R) PRO/1000 Legacy Network Connection 1.0.3> port 0xd010-0xd017 mem 0xf0000000-0xf001ffff irq 19 at device 3.0 on pci0
em0: Ethernet address: 08:00:27:9f:e0:92

pci0: <base peripheral> at device 4.0 (no driver attached)

pcm0: <Intel ICH (82801AA)> port 0xd100-0xd1ff,0xd200-0xd23f irq 21 at device 5.0 on pci0

pcm0: <SigmaTel STAC9700/83/84 AC97 Codec>

ohci0: <OHCI (generic) USB controller> mem 0xf0804000-0xf0804fff irq 22 at device 6.0 on pci0

usbus0: <OHCI (generic) USB controller> on ohci0

pci0: <bridge> at device 7.0 (no driver attached)

acpi_acad0: <AC Adapter> on acpi0

atkbdc0: <Keyboard controller (i8042)> port 0x60,0x64 irq 1 on acpi0

atkbd0: <AT Keyboard> irq 1 on atkbdc0

kbd0 at atkbd0

atkbd0: [GIANT-LOCKED]

psm0: <PS/2 Mouse> irq 12 on atkbdc0

psm0: [GIANT-LOCKED]

psm0: model IntelliMouse Explorer, device ID 4

attimer0: <AT timer> port 0x40-0x43,0x50-0x53 on acpi0

Timecounter "i8254" frequency 1193182 Hz quality 0

Event timer "i8254" frequency 1193182 Hz quality 100

sc0: <System console> at flags 0x100 on isa0

sc0: VGA <16 virtual consoles, flags=0x300>

vga0: <Generic ISA VGA> at port 0x3c0-0x3df iomem 0xa0000-0xbffff on isa0

atrtc0: <AT realtime clock> at port 0x70 irq 8 on isa0

Event timer "RTC" frequency 32768 Hz quality 0

ppc0: cannot reserve I/O port range

Timecounters tick every 10.000 msec

pcm0: measured ac97 link rate at 485193 Hz

em0: link state changed to UP

usbus0: 12Mbps Full Speed USB v1.0

ugen0.1: <Apple> at usbus0

uhub0: <Apple OHCI root HUB, class 9/0, rev 1.00/1.00, addr 1> on usbus0 cd0 at ata1 bus 0 scbus1 target 0 lun 0

cd0: <VBOX CD-ROM 1.0> Removable CD-ROM SCSI-0 device

cd0: 33.300MB/s transfers (UDMA2, ATAPI 12bytes, PIO 65534bytes)

cd0: Attempt to query device size failed: NOT READY, Medium not present

ada0 at ata0 bus 0 scbus0 target 0 lun 0

ada0: <VBOX HARDDISK 1.0> ATA-6 device

ada0: 33.300MB/s transfers (UDMA2, PIO 65536bytes)

ada0: 12546MB (25694208 512 byte sectors: 16H 63S/T 16383C)

ada0: Previously was known as ad0

Timecounter "TSC" frequency 3007772192 Hz quality 800

Root mount waiting for: usbus0

uhub0: 8 ports with 8 removable, self powered

Trying to mount root from ufs:/dev/ada0p2 [rw]...

Setting hostuuid: 1848d7bf-e6a4-4ed4-b782-bd3f1685d551.

Setting hostid: 0xa03479b2.

Entropy harvesting: interrupts ethernet point_to_point kickstart.

Starting file system checks:

/dev/ada0p2: FILE SYSTEM CLEAN; SKIPPING CHECKS

/dev/ada0p2: clean, 2620402 free (714 frags, 327461 blocks, 0.0% fragmentation)

Mounting local file systems:.

vboxguest0 port 0xd020-0xd03f mem 0xf0400000-0xf07fffff,0xf0800000-0xf0803fff irq 20 at device 4.0 on pci0 vboxguest: loaded successfully

Setting hostname: machine3.example.com.

Starting Network: lo0 em0.

lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> metric 0 mtu 16384 options=3<RXCSUM,TXCSUM>

inet6 ::1 prefixlen 128

inet6 fe80::1%lo0 prefixlen 64 scopeid 0x3

inet 127.0.0.1 netmask 0xff000000

nd6 options=21<PERFORMNUD,AUTO_LINKLOCAL>

em0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> metric 0 mtu 1500 options=9b<RXCSUM,TXCSUM,VLAN_MTU,VLAN_HWTAGGING,VLAN_HWCSUM> ether 08:00:27:9f:e0:92

nd6 options=29<PERFORMNUD,IFDISABLED,AUTO_LINKLOCAL> media: Ethernet autoselect (1000baseT <full-duplex>) status: active

Starting devd.

Starting Network: usbus0.

DHCPREQUEST on em0 to 255.255.255.255 port 67

DHCPACK from 10.0.2.2

bound to 192.168.1.142 -- renewal in 43200 seconds.

add net ::ffff:0.0.0.0: gateway ::1

add net ::0.0.0.0: gateway ::1

add net fe80::: gateway ::1

add net ff02::: gateway ::1

ELF ldconfig path: /lib /usr/lib /usr/lib/compat /usr/local/lib 32-bit compatibility

ldconfig path: /usr/lib32

Creating and/or trimming log files.

Starting syslogd.

No core dumps found.

Clearing /tmp (X related).

Updating motd:.

Configuring syscons: blanktime.

Generating public/private rsa1 key pair.

Your identification has been saved in /etc/ssh/ssh_host_key.

Your public key has been saved in /etc/ssh/ssh_host_key.pub.

The key fingerprint is:

10:a0:f5:af:93:ae:a3:1a:b2:bb:3c:35:d9:5a:b3:f3

root@machine3.example.com

The key's randomart image is:

```
+-- [RSA1 1024] ---- +
|      o..          |
|      o . .        |
| .    o            |
|      o            |
|      o S          |
|      + + o        |
|o . + *            |
|o+ ..+ .           |
|==o..o+E           |
+-----------------  +
```

Generating public/private dsa key pair.

Your identification has been saved in /etc/ssh/ssh_host_dsa_key.

Your public key has been saved in /etc/ssh/ssh_host_dsa_key.pub.

The key fingerprint is:

7e:1c:ce:dc:8a:3a:18:13:5b:34:b5:cf:d9:d1:47:b2

root@machine3.example.com

The key's randomart image is

```
: +-- [ DSA 1024]---+
|      ..          ..|
|      o .         . +   |
|      ...         .E .|
|      . . o       o . .  |
|      + S =     . |
```

```
|     + . = o   |
|     + . * .   |
|     . . o .   |
|     .o. .     |
+--------------- +
```

Starting sshd.
Starting cron.
Starting background file system checks in 60 seconds. Thu Oct 6 19:15:31 MDT 2011
FreeBSD/amd64 (machine3.example.com) (ttyv0) login:

На медленных машинах генерирование ключей RSA и DSA может занять ощутимое время. Это происходит лишь при первой загрузке новой системы, и лишь в случае, когда sshd настроен на автоматический запуск. Последующие загрузки будут проходить быстрее.

По умолчанию во FreeBSD не устанавливается никаких графических оболочек, однако в наличии они имеются.

Завершение работы FreeBSD

Корректное завершение работы компьютера с FreeBSD помогает защитить от повреждений не только данные, но даже и аппаратное обеспечение. Не стоит просто выключать питание. Если вы входите в группу wheel, то станьте суперпользователем набрав в командной строке команду su и введя пароль пользователя root. Или же, войдите в систему как root и наберите команду *shutdown -p now*. Система корректно завершит работу и выключится. Комбинация клавиш Ctrl+Alt+Del может быть задействована для перезагрузки системы, однако во время нормальной работы пользоваться ею не рекомендуется.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Произвести установку операционной системы FreeBSD. Для установки необходимо:

1. Взять CD диск с образом операционной системы.
2. Зайти BIOS и изменить последовательность загрузки.
3. Выбрать параметры запуска ядра.
4. Запустить установку.
5. Выбрать необходимую раскладку клавиатуры.
6. Указать имя хоста.
7. Выбрать устанавливаемые компоненты.
8. Необходимо разбить жесткий диск на разделы, для этого используется утилита cfdisk или fdisk, а также шаблонное разбиение. (При необходимости удалить существующие разделы. Создать новый раздел, указать, что он является основным. Создать раздел подкачки. Записать изменения на диск.)
9. Завершить установку.
10. Задать пароль.
11. Перезагрузить компьютер.
12. Выполнить вход в систему под пользователем root.
13. Завершить работу с FreeBSD.
14. Ответить на контрольные вопросы и подготовить отчет.

ФОРМА ОТЧЕТА О ВЫПОЛНЕННОЙ РАБОТЕ

Отчет на защиту предоставляется в печатном или электронном виде.

Структура отчета (на отдельном листе(-ах)): титульный лист, формулировка задания (вариант), этапы выполнения работы (со скриншотами), результаты выполнения работы (скриншоты и содержимое файлов).

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Опишите назначение менеджера загрузки (Boot Manager).
2. Опишите назначение раздела подкачки.
3. Назовите точку монтирования корневой файловой системы.
4. Назовите утилита, которая запускает процесс установки.
5. Дайте определение FreeBSD.
6. Перечислите возможности FreeBSD.
7. Раскройте область применения FreeBSD.
8. Изложите концепцию проекта FreeBSD.
9. Приведите алгоритм разработки проекта FreeBSD.
10. Дайте определение CVS-репозиторию.
11. Опишите вклад коммитеров.
12. Дайте определение Core-группе.
13. Предложите пути установки FreeBSD.
14. Назовите минимальные требования для установки FreeBSD.
15. Перечислите этапы установки FreeBSD.
16. Опишите особенности bsdinstall.
17. Охарактеризуйте Bootonly.
18. Предложите методы разделения дискового пространства.
19. Объясните, как задать пароль пользователю root.

ЛАБОРАТОРНАЯ РАБОТА №2
НАСТРОЙКА МЕЖСЕТЕВОГО ЭКРАНА

ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Целью выполнения лабораторной работы является получение практических навыков по настройке межсетевого экрана.

Основной задачей выполнения лабораторной работы является: научиться использовать и настраивать межсетевой экран в ОС FreeBSD на примере IPFW.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Межсетевые экраны

Межсетевые экраны (firewall, брандмауэр) делают возможной фильтрацию входящего и исходящего трафика, идущего через систему. Межсетевой экран использует один или более наборов "правил" для проверки сетевых пакетов при их входе или выходе через сетевое соединение, он или позволяет прохождение трафика или блокирует его. Правила межсетевого экрана могут проверять одну или более характеристик пакетов, включая но не ограничиваясь типом протокола, адресом хоста источника или назначения и портом источника или назначения.

Межсетевые экраны могут серьезно повысить уровень безопасности хоста или сети. Они могут быть использованы для выполнения одной или более нижеперечисленных задач: Для защиты и изоляции приложений, сервисов и машин во внутренней сети от нежелательного трафика,

приходящего из внешней сети интернет.

Для ограничения или запрещения доступа хостов внутренней сети к сервисам внешней сети интернет.

Для поддержки преобразования сетевых адресов (network address translation, NAT), что позволяет использование во внутренней сети приватных IP адресов (либо через один выделенный IP адрес, либо через адрес из пула автоматически присваиваемых публичных адресов).

Что такое FireWall

FireWall — это модуль ядра, который обрабатывает всю входящую информацию до того, как она будет передана соответствующим программам; и всю исходящую информацию, какой бы программой она ни была отправлена. FireWall анализирует эти данные и либо пропускает их дальше, либо блокирует, основываясь на некоторых правилах. Правильная настройка FireWall позволяет защитить систему от нежелательных внешних вторжений и ограничить возможности программ, работающих внутри системы.

FireWall— это не программа, а подсистема ядра, что он может блокировать трафик и что его можно гибко настраивать.

Принцип работы

Существует два основных способа создания наборов правил межсетевого экрана: "включающий" и "исключающий". Исключающий межсетевой экран позволяет прохождение всего трафика, за исключением трафика, соответствующего набору правил. Включающий межсетевой экран действует прямо противоположным образом. Он пропускает только трафик, соответствующий правилам и блокирует все остальное.

Включающий межсетевой экран обеспечивает гораздо большую степень контроля исходящего трафика. Поэтому включающий межсетевой экран является лучшим выбором для систем, предоставляющих сервисы в сети Интернет. Он также контролирует тип трафика, порождаемого вне и направляющегося в вашу приватную сеть. Трафик, не попавший в правила, блокируется, а в файл протокола вносятся соответствующие записи. Включающие межсетевые экраны обычно более

безопасны, чем исключающие, поскольку они существенно уменьшают риск пропуска межсетевым экраном нежелательного трафика.

Примечание

Если не указано иначе, то все приведенные в этом разделе примеры наборов правил и конфигураций относятся к типу включающего межсетевого экрана.

Безопасность может быть дополнительно повышена с использованием "межсетевого экрана с сохранением состояния". Такой межсетевой экран сохраняет информацию об открытых соединениях и разрешает только трафик через открытые соединения или открытие новых соединений. Недостаток межсетевого экрана с сохранением состояния в том, что он может быть уязвим для атак DoS (DenialofService, отказ в обслуживании), если множество новых соединений открывается очень быстро. Большинство межсетевых экранов позволяют комбинировать поведение с сохранением
состояния и без сохранения состояния, что позволяет создавать оптимальную конфигурацию для каждой конкретной системы.

Пакеты межсетевых экранов

В FreeBSD встроено три программных межсетевых экрана. Это IPFILTER (известный также как IPF), IPFIREWALL (известный также как IPFW) и OpenBSDPacketFilter (также известный как PF). Помимо этого, FreeBSD содержит два пакета ограничения трафика (шейпера): altq и dummynet. Dummynet традиционно сильно связан с IPFW, а ALTQ с PF. В настоящее время IPFILTER не поддерживает ограничение пропускной способности сетевого соединения. Для реализации этой функции предлагается использовать IPFILTER совместно с одним из двух существующих пакетов ограничения трафика. Конфигурация следующая: IPFILTER задействуется для фильтрации и трансляции трафика, а IPFW с *dummynetили* PF с ALTQ — для контроля пропускной способности сетевого соединения. IPFW и PF для контроля исходящих и входящих пакетов используют наборы правил, хотя и разными способами с разным синтаксисом правил.

Причина, по которой в FreeBSD включено более одного пакета межсетевых экранов, заключается в том, что разные сети выдвигают к ним

различные требования и используют разные предпочтения. Нет одного пакета, который был бы очевидно лучше других.

Поскольку все межсетевые экраны основаны на анализе значений выбранных полей заголовка пакета, для создания правил межсетевого экрана необходимо понимание принципов TCP/IP, того, что означают различные поля заголовка пакета, и как эти поля используются в обычной сессии. Хорошим примером является: *http://www.ipprimer.com/overview.cfm*.

НАСТРОЙКА МЕЖСЕТЕВОГО ЭКРАНА

IPFW

IPFIREWALL (IPFW) — представляет собой межсетевой экран, написанный и поддерживаемый добровольными участниками проекта FreeBSD. Он использует *stateless* правила, т.е. правила без учета состояния, и наследование техники кодирования правил для получения того, что называется простой логикой с сохранением состояния (*stateful*).

Пример простейшего набора правил IPFW (находится в */etc/rc.firewallu /etc/rc.firewall6*) в стандартной установке FreeBSD достаточно прост и не рассчитана непосредственное использование без изменений. В нём не используется фильтрация с сохранением состояния, которая даёт преимущества во многих конфигурациях, поэтому он не может быть взят за основу для этого раздела.

Синтаксис правил IPFW без сохранения состояния обеспечивает расширенные возможности фильтрации, которые намного превосходят уровень знаний обычного пользователя межсетевого экрана. IPFW рассчитан на профессиональных пользователей или технически продвинутых любителей, которые предъявляют повышенные требования к фильтрации пакетов. Чтобы использовать возможности IPFW в полную силу, необходимы углубленные знания того, как в различных протоколах формируются и используются заголовки пакетов.

IPFW состоит из семи компонентов, главный из которых — процессор правил фильтрации уровня ядра и интегрированный в него механизм учета пакетов, а также средства протоколирования пакетов, правило divert, посредством которых вызывается функция NAT и другие возможности специального назначения, средства для ограничения

скорости (шейпинга) трафика (dummynet), средства перенаправления fwd, средства организации сетевого моста bridge и механизм ipstealth. IPFW поддерживает протоколы IPv4 и IPv6.

Включение IPFW

IPFW включён в базовую установку FreeBSD в виде отдельного подгружаемого модуля. Система динамически загружает модуль ядра, когда в *rc.conf* присутствует строка *firewall_enable="YES*" . Если использовать функциональность NAT не планируется, то в этом случае дополнительно компилировать IPFW в состав ядра FreeBSD не требуется.

После перезагрузки системы с *firewall_enable="YES"* в *rc.conf* на экране в процессе загрузки отобразится выделенное белым сообщение:

ipfw2 initialized, divert disabled, rule-based forwarding disabled, 0
ipfw2 initialized, divert disabled, rule-based forwarding disabled, default to deny, logging disabled

Загружаемый модуль скомпилирован с возможностью протоколирования информации о трафике. Для включения протоколирования и установки уровня его детализации имеется переключатель, значение которого можно установить в конфигурационном файле */etc/sysctl.conf*. При добавлении следующих двух строк протоколирование будет включено при следующей загрузке системы:

net.inet.ip.fw.verbose=1
net.inet.ip.fw.verbose_limit=5

Параметры ядра

Включение следующих параметров в ядро FreeBSD не является обязательным, если дополнительно не требуется функциональность NAT. Эти параметры представлены здесь в качестве справочной информации для дальнейших примеров.

options IPFIREWALL

Этот параметр включает IPFW в состав ядра.

options IPFIREWALL_VERBOSE

Этот параметр включает протоколирование пакетов, которые проходят через IPFW по правилам с ключевым словом log.

options IPFIREWALL_VERBOSE_LIMIT=5

Ограничение числа пакетов, прошедших через *syslogd*, отдельно для каждого правила. Этот параметр имеет смысл использовать в недружественной среде, когда необходимо отслеживать активность межсетевого экрана. Это закрывает возможность атак типа «отказ в обслуживании» через флуд сообщениями *syslog*.

options IPFIREWALL_DEFAULT_TO_ACCEPT

Этот параметр включает для IPFW разрешающую политику по умолчанию. Это удобно на первых этапах настройки IPFW.

options IPDIVERT

Включение функциональности NAT.

Примечание

Межсетевой экран будет блокировать все входящие и исходящие пакеты, если отсутствует параметр ядра IPFIREWALL_DEFAULT_TO_ACCEPT или правило, явно разрешающее эти соединения.

Параметры /etc/rc.conf

Включение межсетевого экрана:

firewall_enable="YES"

Для выбора одного из стандартных режимов работы межсетевого экрана, предоставляемых FreeBSD, выберите наиболее подходящий в файле */etc/rc. firewall* и разместите так, как указано ниже:

firewall_type="open"

Возможны следующие значения для этого параметра:

- *open*— пропускать весь трафик.
- *client*— защищать только эту машину.
- *simple*— защищать всю сеть.
- *closed*— полностью запретить IP трафик, за исключением *loopback* интерфейса.
- UNKNOWN — отключить загрузку правил межсетевого экрана.
- *filename*— абсолютный путь к файлу, содержащему правила межсетевого экрана.

Есть два варианта загрузки собственных правил в межсетевой экран ipfw. Первый способ — задать переменную firewall_type в виде абсолютного пути файла, содержащего правила межсетевого экрана без каких-либо параметров командной строки для самого ipfw. Ниже приведён простой пример набора правил, который блокирует весь входящий и исходящий трафик:

add deny in add deny out

Второй способ — установить значение переменной firewall_script в виде абсолютного пути исполняемого скрипта, содержащего команды ipfw, которые будут выполнены во время загрузки операционной системы. Правильный формат правил исполняемого скрипта должен соответствовать формату файла, приведённому ниже:

```
#!/bin/sh ipfw -q flush
ipfw add deny in ipfwadddenyout
```

Примечание

Если переменной firewall_type присвоено значение client или simple, то правила, расположенные по умолчанию в /etc/rc.firewall, должны быть приведены в соответствие с конфигурацией данной

машины. Также заметим, что для используемых в этой главе примеров в качестве значения переменной firewall_script используется /etc/ipfw.rules.

Включение протоколирования:

firewall_logging="YES"

Предупреждение

Единственное, что делает параметр firewall_logging, — присвоение логической единицы переменной sysctlnet.inet.ip.fw. В rc.conf нет переменной для ограничения протоколирования, но это можно сделать через переменную sysctl вручную либо используя файл /etc/sysctl.conf:

net.inet.ip. fw.verbose_limit=5

Команда IPFW

Команда *ipfw* — это стандартный механизм для ручного добавления/удаления отдельных правил в активной цепочке правил межсетевого экрана. Основная проблема при использовании этого метода состоит в том, что при перезагрузке операционной системы все изменения, сделанные с помощью данной команды, будут утеряны. Вместо этого рекомендуется записать все правила в файл, из которого они будут считываться во время загрузки операционной системы, а также для полной замены текущего набора правил на содержимое из файла.

Тем не менее, команду *ipfw* удобно использовать для отображения текущей конфигурации правил на экране консоли. Учетный модуль IPFW динамически создаёт счётчики для каждого правила, которые подсчитывают количество пакетов, соответствующих условиям срабатывания правила. В процессе тестирования отображение правила со своим счётчиком является одним из способов проверки, срабатывает ли

правило при прохождении через него пакета или нет.

Вывод полного списка правил:

 # ipfwlist

Вывод полного списка правил с маркером времени последнего срабатывания правила:

 # ipfw -t list

Следующий пример выводит учетную информацию, количество совпавших пакетов и сами правила. Первым столбцом идет номер правила, за ним следует число совпавших исходящих пакетов, третий столбец — число соответствующих входящих пакетов, и затем само правило.

 # ipfw -a list

Вывод динамических правил вместе со статическими:

 # ipfw -d list

Отобразить статические и динамические правила, в т.ч. с истекшим временем действия:

 # ipfw -d -e list Обнуление счетчиков: # ipfwzero

Обнулить счетчики для правила под номером *NUM*:

 # ipfwzero NUM

Набор правил IPFW

Набор правил (*ruleset*) представляет собой группу правил IPFW, которые разрешают или запрещают прохождение пакета через межсетевой экран на основании значений, содержащихся в пакете.

Двунаправленный обмен пакетов между машинами является сессией. Набор правил межсетевого экрана анализирует как пакеты,

приходящие из глобальной сети, так и ответные пакеты, исходящие из системы. Каждый TCP/IP сервис (такой как *telnet, www, mail,* и т.д.) принадлежит определенному протоколу и привилегированному (прослушиваемому) порту. Пакеты, предназначенные для конкретного сервиса, передаются с непривилегированного (с высоким значением) порта по адресу назначения на указанный порт сервиса. Все эти параметры (т.е. порты и адреса) могут быть использованы в качестве критериев фильтрации при создании правил, которые пропускают или блокируют сервисы.

Когда пакет попадает в межсетевой экран, он сравнивается с каждым правилом, начиная с первого, двигаясь по множеству правил верху вниз в порядке увеличения номера правил. Когда пакет совпадает с критерием выбора правила, выполняется действие, указанное в правиле, и на этом поиск правил прекращается. Такой метод поиска известен как «выигрыш первого совпадения», т.е. после срабатывания правила оставшиеся не просматриваются. Если содержимое пакета не соответствует ни одному из правил, он принудительно попадает на встроенное правило по умолчанию, заданное под номером 65535, которое запрещает и отбрасывает все пакеты без какого-либо отклика в сторону отправителя.

Примечание

Поиск продолжается после правил count, skipto и tee.

Упомянутые здесь инструкции основаны на использовании правил, содержащих параметры с сохранением состояния keepstate, limit, in, out и via. Это основной механизм для кодирования набора правил межсетевого экрана закрытого типа.

Предупреждение

Будьте осторожны, когда работаете с правилами межсетевого экрана, так как вы можете легко заблокировать самого себя.

Синтаксис правил

Представленный здесь синтаксис правил был упрощен для создания

стандартного набора правил межсетевого экрана закрытого

типа. Для полного описания синтаксиса правил смотрите страницу Справочника *ipfw*.

Правила содержат ключевые слова: эти ключевые слова записываются в строке в определенном порядке слева направо. Ключевые слова выделены полужирным шрифтом. Некоторые ключевые слова имеют дополнительные параметры, которые могут являться ключевыми словами для них самих и также содержать вложенные дополнительные параметры.

Символ # используется для обозначения начала комментария и может быть расположен в конце строки с правилом или в начале строки над правилом. Пустые строки игнорируются.

CMD RULE_NUMBER ACTION LOGGING SELECTION STATEFUL

CMD

Каждое новое правило должно начинаться с префикса add для добавления во внутреннюю таблицу.

RULE_NUMBER

Каждое правило обозначено номером в диапазоне 1..65535.

ACTION

При соответствии пакета описанным в правиле критериям фильтрации будет выполнено одно из следующих действий.

allow | *accept* | *pass* | *permit*

Все эти действия означают одно и то же — пакеты, совпадающие с правилом, могут покинуть обработку правил межсетевого экрана. На этом поиск прекращается.

check-state

Проверяет пакет на соотвествие динамической таблице правил. Если совпадение найдено, выполняется действие, содержащееся в правиле, породившем данное динамическое правило, иначе выполняется переход к следующему правилу. Правило check-state не имеет критериев фильтрации. При отсутствии правила check-state в наборе правил, проверка по динамической таблице происходит на первом правиле keep-state или limit.

deny | drop

Оба слова означают отбрасывание пакетов, совпавших с правилом. Поиск прекращается.

Протоколирование

Когда пакет совпадает с правилом, содержащим ключевое слово log, информация об этом событии записывается в syslogdc пометкой SECURITY. Запись в журнал происходит только в том случае, если число срабатываний для данного правила не превышает значения параметра log-amount. Если значение log amount не объявлено, то ограничение берется из значения переменной sysctlnet.inet.ip.fw.verbose_limit. В обоих случаях обнуление значения отменяет ограничение. По достижению установленного лимита запись в журнал может быть повторно включена путем сброса счетчика срабатываний или счетчика пакетов для этого правила; смотрите описание команды ipfw resetlog.

Примечание

Протоколирование осуществляется после проверки на соответствие всем условиям в правиле и перед выполнением окончательного действия (accept, deny) над пакетом. Вы должны выбрать сами, какие действия правил вы хотите включить в журнал.

Условия отбора

Ключевые слова, представленные в этом разделе, используются для описания атрибутов пакета, по которым проверяется условие

срабатывания того или иного правила. Для совпадения используется следующая последовательность атрибутов общего назначения:

udp | tcp | icmp

Также могут быть использованы имена протоколов, описанные в */etc/protocols*. Указанное значение обозначает протокол для совпадения. Это является обязательным требованием.

fromsrctodst

Ключевые слова *from* и *to* служат для фильтрации по IP адресам. Обязательно должны быть указаны и источник, и получатель. *any* — это специальное ключевое слово, которое соответствует любому IP адресу. me — это специальное ключевое слово, которое соответствует любому из IP адресов, сконфигурированных на интерфейсе вашей системы FreeBSD, и служит для указания компьютера, на котором работает межсетевой экран (т.е. этот компьютер), как показано на примерах *from me to any*, *from any to me*,from 0.0.0.0/0 *to any, from any to* 0.0.0.0/0 , *from* 0.0.0.0 *to any, from any to* 0.0.0.0 и *from me to* 0.0.0.0 . IP адрес указывается в виде четырёх чисел, разделённых точками, или дополнительно с префиксом сети (нотация CIDR). Это является обязательным требованием. Для упрощения вычислений, связанных с IP адресами, используйте порт *net-mgmt/ipcalc*. Более подробную информацию можно посмотреть на странице программы: *http://jodies.de/ipcalc* .

portnumber

Для протоколов, работающих с портами (такие как TCP и UDP), обязательным требованием является указание номера порта соответствующего сервиса. Вместо номера порта можно использовать имя сервиса (из */etc/services*).

in | out

Отбор соответственно по входящим и исходящим пакетам. Присутствие одного из этих ключевым слов в правиле обязательно для формирования критерия фильтрации.

via IF

Совпадает с пакетами, проходящими через указанный интерфейс. Ключевое слово *via* включает обязательную проверку на указанном интерфейсе в общий процесс поиска совпадений.

setup

Это обязательное ключевое слово определяет начало запроса сессии для TCP пакетов.

keep-state

Это обязательное ключевое слово. При совпадении межсетевой экран создает динамическое правило, которое по умолчанию будет совпадать с двунаправленным трафиком между отправителем и получателем для данной пары IP/порт по указанному протоколу.

limit {src-addr | src-port | dst-addr | dst-port}

Межсетевой экран разрешит только *N* соединений с одинаковым набором параметров, указанных в правиле. Можно задавать один или несколько адресов и портов отправителя и получателя. В одном и том же правиле использование *limit* и *keepstate* не допускается. Параметр *limit* предоставляет такую же функцию с сохранением состояний, что и *keep- state*, плюс свои собственные.

Параметры для правил с сохранением состояния

С точки зрения фильтрации по правилам с сохранением состояния весь трафик выглядит как двусторонний обмен пакетами, включая

данные о сессиях. При такой фильтрации у нас есть средства сопоставления и определения корректности процедуры двустороннего обмена пакетами между стороной, породившей пакет, и стороной- получателем. Любые пакеты, которые не подходят под шаблон сессии, автоматически отбрасываются как злонамеренные.

Параметр *check-state* служит для указания места в наборе правил IPFW, в котором пакет будет передан на поиск соответствий динамическим правилам. В случае совпадения пакет пропускается, при этом создается новое динамическое правило для следующего пакета, принадлежащего данной двусторонней сессии. В противном случае пакет движется по обычным правилам, начиная со следующей позиции.

Динамические правила уязвимы к атаке SYN-пакетами, которые могут породить гигантское количество динамических правил. Для предотвращения такого рода атак во FreeBSD предусмотрен еще один параметр — *limit*. Этот параметр служит для ограничения количества одновременно установленных сессий путём проверки полей отправителя и получателя, в зависимости от параметра *limit,* с использованием IP адреса пакета для поиска открытых динамических правил, которые представляют собой счетчик количества совпадений для данного IP адреса и этого правила. Если это количество превышает значение, указанное в параметре *limit,* то такой пакет отбрасывается.

Протоколирование сообщений межсетевого экрана

Преимущества протоколирования очевидны: это предоставляет возможность отслеживать постфактум, прохождение каких пакетов было отклонено, откуда эти пакеты пришли и куда они назначались для тех правил, в которых включена функция записи в журнал. Это замечательный инструмент для отслеживания атак на вашу систему.

Даже при включенной функции ведения журнала само по себе оно производиться не будет. Администратор межсетевого экрана определяет, для каких правил будет включена функция ведения журнала, и добавляет к этим правилам *log*. Обычно в журнал пишутся только запрещающие правила, такие как правила *deny* для входящего
ICMP *ping*. Довольно часто конец списка добавляют дублирующее правило вида

«*ipfwdefaultdenyeverything*» приставкой log. Это позволяет отслеживать все

пакеты, не совпадающие ни с одним из правил в вашем наборе. Будьте крайне осмотрительны при использовании функции ведения журнала, так как это чревато несоразмерным разрастанием файла журнала, вплоть до полного заполнения им места на жестком диске. DoS атаки, направленные на переполнение свободного пространства жесткого диска, являются одними из самых старейших. Помимо заполнения жесткого диска это неприятно еще и тем, что сообщения журнала пишутся не только в *syslogd*, но также отображаются на экране системной консоли, и это вскоре начинает сильно раздражать.

Параметр ядра IPFIREWALL_VERBOSE_LIMIT=5 ограничивает число идущих подряд сообщений в системный регистратор *syslogd*, касающихся пакетов, совпавших с правилом. Когда этот параметр включен в ядро, число последовательно идущих сообщений для определенного правила обрезается указанным числом. От записи 200 идентичных сообщений особого прока нет. В данном случае для сработавшего правила в журнале *syslogd* будут зафиксированы 5 сообщений подряд, остальные идентичные сообщения будут подсчитаны и отправлены в *syslogd* как одно сообщение такого вида:

last message repeated 45 times

Путь к файлу, в который пишутся сообщения, задается в файле */etc/syslog.conf*. По умолчанию это файл */var/log/security*.

Написание скрипта правил

Наиболее опытные пользователи IPFW создают скрипт, содержащий в себе правила, оформленные таким образом, что они могут быть исполнены как обыкновенный *sh*- скрипт. Основное преимущество такого подхода в том, что правила можно полностью заменить на новые без необходимости в перезагрузке системы для ихактивации. Это крайне удобно на этапе разработки и тестирования

набора правил, т.к. перезагружать весь список правил можно сколь

угодно часто. Помимо того, поскольку это скрипт, то здесь можно объявить некие часто используемые значения в виде переменной, и использовать её во множестве правил, как показано в примере ниже.

Синтаксис примера, приведенного ниже, совместим с тремя командными оболочками: *sh, csh, tcsh.* Для использования значения ранее объявленной переменной имя переменной предваряется символом $. Во время присвоения имя переменной не имеет префикса $, присваиваемое значение должно быть заключено в "двойные кавычки".

Так выглядит файл с правилами, с которого вы можете начать:

```
   ####### начало примера скрипта с правилами ipfw
#########
   ipfw -q -f flush # Сброс всех правил. # Установки по
умолчанию
   oif="tun0" # наш интерфейс
   odns="192.0.2.11"   # IP DNS сервера провайдера
cmd="ipfw -q add "     # префикс для создания правил
ks="keep-state" # просто лень вводить каждый раз
   $cmd 00500 check-state
   $cmd 00502 deny all from any to any frag
   $cmd 00501 deny tcp from any to any established
   $cmd 00600 allow tcp from any to any 80 out via $oif
setup $ks
   $cmd 00610 allow tcp from any to $odns 53 out via
$oif setup $ks
   $cmd 00611 allow udp from any to $odns 53 out via
$oif $ks ########## конец примера скрипта с правилами
ipfw #########
```

Вот и все, что нужно сделать. Сами правила в этом примере не столь важны, они написаны ради того, чтобы продемонстрировать использование подстановки значения переменной по ее имени.

Если бы этот скрипт находился в файле */etc/ipfw.rules*, то правила можно было бы перезагрузить следующей командой.

```
   # ipfw -q -f flush
   # ipfw -q add check-state

#      ipfw -q add deny all from any to any frag
```

 # ipfw -q add deny tcp from any to any established
 # ipfw -q add allow tcp from any to any 80 out via tun0 setup keepstate
 # ipfw -q add allow tcp from any to 192.0.2.11 53 out via tun0 setup keep-state
 # ipfw -q add 00611 allow udp from any to 192.0.2.11 53 out via tun0 keep- state

Набор правил с сохранением состояния

Следующий набор правил, не включающий в себя правила трансляции адресов NAT, является примером того, как создавать правила для межсетевого экрана закрытого типа высокого уровня защиты. Закрытый межсетевой экран разрешает трафик, описанный в разрешающих правилах, и по умолчанию блокирует всё остальное. Межсетевой экран, предназначенный для защиты сегментов сети, имеет как минимум два интерфейса, для которых должны быть написаны правила для работы межсетевого экрана.

Все разновидности операционных систем UNIX, включая FreeBSD, используют интерфейс lo0 и IP адрес 127.0.0.1 для передачи данных внутри операционной системы. Правила межсетевого экрана должны содержать в своем составе правила, разрешающие беспрепятственное прохождение трафика по этому интерфейсу.

Интерфейс, подключенный к Интернет, является местом для размещения правил авторизации и контроля доступа исходящих и входящих соединений. Это может быть туннельный интерфейс PPP tun0 или сетевой адаптер, подключенный к DSL или кабельному модему. В случае, когда за межсетевым экраном один и более интерфейсов подсоединён к локальной сети, должны присутствовать правила для беспрепятственного прохождения исходящих пакетов с этих интерфейсов LAN. Правила изначально разделяются на три основных раздела: интерфейсы, не ограниченные правилами, правила для исходящего трафика на внешнем интерфейсе и правила для входящего трафика на внешнем интерфейсе.

В каждом из разделов, относящихся к внешнему интерфейсу, правила должны быть упорядочены по следующему принципу:
наиболее используемые расположены в начале, наименее используемые — в конце. Последним должно идти правило блокирования и занесения

в журнал информации о пакетах на этом интерфейсе, не попавших под предыдущие правила.

Раздел, описывающий правила для исходящего трафика на внешнем интерфейсе, содержит только разрешающие правила *allow*, состоящие из значений фильтрации, которые однозначно определяют сервис, которому разрешен доступ в Интернет. Все правила включают в себя поля *proto, port, in/out, via* и *keepstate*. Правила, содержащие *proto tcp*, имеют также параметр setup, который служит для определения начала сессии, которое в дальнейшем передается как условие срабатывания в динамическую таблицу.

В разделе, описывающем правила для входящего трафика на внешнем интерфейсе, в самом начале должны стоять правила, блокирующие нежелательные пакеты. Так должно быть по двум причинам. Первая состоит в том, что пакеты, сформированные злоумышленником, могут частично или полностью соответствовать разрешающим правилам *allow*. Вторая причина состоит в том, что заведомо не интересующие нас пакеты могут быть просто отклонены, вместо того, чтобы быть перехваченными и записанными в файл журнала по последнему правилу. Последнее правило в каждом разделе блокирует и регистрирует в журнале все пакеты и может быть использовано для юридических обоснований в ходе разбирательств против злоумышленников, атаковавших вашу систему.

Также следует убедиться в том, что ваш сервер не отвечает ни на какие другие формы непредусмотренного трафика. Некорректные пакеты должны быть просто отброшены. В результате атакующие не получат информацию о том, достиг ли его пакет вашего сервера. Чем меньше атакующие будут знать о вашей системе, тем более она защищена. Назначение нераспознанного номера порта можно посмотреть в файле */etc/services/* или по адресу *http://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers*. Рекомендуем ознакомиться с содержимым ссылки относительно номеровпортов, используемых троянами: *http://www.sans.org/security-resources/idfaq/oddports.php/*

Пример правил для межсетевого экрана закрытого типа

Следующие правила, не включающие поддержку NAT, являются

логически полным набором правил для межсетевого экрана закрытого типа. При использовании этого набора правил вы вполне можете быть уверены в безопасности вашей системы. Просто закомментируйте некоторые из правил *pass* для тех служб, которые вам не требуются. Чтобы избежать занесения в журнал нежелательных сообщений, добавьте правило denyв раздел, описывающий входящий трафик на интерфейсе. Замените название интерфейса dc0, упоминающегося в правилах ниже, на название интерфейса (NIC), который соединяет вашу систему с глобальной сетью. Для PPP соединений это будет tun0.

Примечание по использованию этих правил:

Все запросы начала сессии с внешней сетью используют параметр *keep-state*.

Все разрешенные сервисы внешней сети имеют параметр *limit* для защиты от флуда.

Все правила используют параметры *in* или *out* для указания направления трафика

Все правила используют параметр *via имя-интерфейса* для уточнения интерфейса, через который проходит пакет.

Следующие правила записываются в */etc/ipfw.rules*.

```
#     ########### Начало   файла   с   правилами   IPFW
###############
#     Сброс всех правил перед началом работы скрипта.
   ipfw -q -f flush
#     Префикс для создания правил
   cmd="ipfw -q add"
   pif="dc0"    # название внешнего интерфейса,          #
принадлежащего глобальной сети
   ######################################################
#######
#    Нет ограничений на внутреннем интерфейсе локальной
сети
#    Нет необходимости в этом, если у вас нет локальной
сети.
#    Замените xl0 на название интерфейса вашей локальной
```

```
сети
   ########################################################
#######
#    $cmd 00005 allow all from any to any via xl0
########################################################
### ######
#     Нет ограничений на интерфейсе Loopback
   ########################################################
#######
   $cmd 00010 allow all from any to any via lo0
########################################################
### ######
#    Разрешить пакет, если он был ранее добавлен в "дина-
мическую"
#     таблицу при помощи выражения allowkeep-state
   ########################################################
######
   $cmd                  00015                 check-state
########################################################
### ######
#    Раздел правил для исходящего трафика на внешнем ин-
терфейсе
#     Анализ запросов начала сессии, идущих из-за межсе-
тевого экрана
#    в локальную сеть или от этого шлюза в интернет.
########################################################
### ######
#     Разрешить исходящий трафик к DNS серверу провайдера
#    x.x.x.x должен быть IP адресом DNS сервера вашего
провайдера
#    Продублируйте эти строки, если у вас больше одного
DNS сервера
#     Эти IP адреса можно взять из файла /etc/resolv.conf
   $cmd  00110 allow tcp from any  to x.x.x.x 53 out via
$pif setup keepstate
   $cmd  00111 allow udp from any  to x.x.x.x 53 out via
$pif keep-state
#     Разрешить исходящий трафик к DHCP серверу провай-
дера для cable/DSL конфигураций.
#    Это правило не нужно для .userppp. соединений с гло-
бальной сетью # в этом случае вы можете удалить эти пра-
вила.
#    Используйте это правило для записи необходимого нам
IP адреса в лог-файл. # Затем укажите IP адрес в заком-
ментированном правиле и удалите первое правило.
```

```
  $cmd 00120 allow log udp from any to any 67 out via
$pif keep-state #$cmd 00120 allow udp from any to x.x.x.x
67 out via $pif keep-state
#    Разрешить исходящий трафик для незащищенного www
соединения
  $cmd 00200 allow tcp from any to any 80 out via $pif
setup keep-state
#    Разрешить исходящий трафик для защищенного www со-
единения
#    https с поддержкой TLS и SSL
  $cmd 00220 allow tcp from any to any 443 out via $pif
setup keepstate
#    Разрешить исходящий POP/SMTP
  $cmd 00230 allow tcp from any to any 25 out via $pif
setup keep-state
  $cmd 00231 allow tcp from any to any 110 out via $pif
setup keepstate
#      Разрешить исходящий трафик для FreeBSD
(makeinstall& CVSUP)
#      По сути назначаем пользователю root полные при-
вилегии.
  $cmd 00240 allow tcp from me to any out via $pif setup
keep-state uid root # Разрешаем исходящий icmpping
  $cmd 00250 allow icmp from any to any out via $pif
keep-state
#    Разрешаем исходящий трафик Time
  $cmd 00260 allow tcp from any to any 37 out via $pif
setup keep-state
#     Разрешаем исходящий трафик nntp news
  $cmd 00270 allow tcp from any to any 119 out via $pif
setup keepstate
#    Разрешаем исходящий защищённый трафик FTP, Telnet и
SCP # Эта функция использует SSH (secureshell)
  $cmd 00280 allow tcp from any to any 22 out via $pif
setup keep-state
#     Разрешаем исходящий трафик whois
  $cmd 00290 allow tcp from any to any 43 out via $pif
setup keep-state
#   Запрещаем и заносим в журнал остальной исходящий
трафик.
#    Обеспечивает политику межсетевого экрана закрытого
типа
  $cmd 00299 deny log all  from any to any out via $pif
########################################################
### #####
```

```
#    Раздел правил для входящего трафика на внешнем ин-
терфейсе
#     Анализ пакетов, приходящих из глобальной сети,
#     предназначенных для этого шлюза или локальной сети
   ######################################################
######
#    Запрещаем весь входящий трафик с немаршрутизируемых
сетей
   $cmd 00300 deny all from 192.168.0.0/16 to any in via
$pif #RFC 1918 private IP
   $cmd 00301 deny all from 172.16.0.0/12 to any in via
$pif #RFC 1918 private IP
   $cmd 00302 deny all from 10.0.0.0/8 to any in via
$pif #RFC 1918 private IP
   $cmd 00303 deny all from 127.0.0.0/8 to any in via $pif
#loopback
   $cmd 00304 deny all from 0.0.0.0/8 to any in via
        $pif
#loopback
   $cmd 00305 deny all from 169.254.0.0/16 to any in via
$pif #DHCP auto-config
   $cmd 00306 deny all from 192.0.2.0/24 to any in via
$pif#reserved for docs
   $cmd 00307 deny all from 204.152.64.0/23 to any in via
$pif #Sun cluster interconnect
   $cmd 00308 deny all from 224.0.0.0/3 to any in via
$pif#Class D & E multicast # Запрещаем пинг извне
   $cmd 00310 deny icmp from any to any in via $pif #
Запрещаемident
   $cmd 00315 deny tcp from any to any 113 in via $pif
#      Запрещаем все     Netbios службы.  137=name,
138=datagram, 139=session # Netbiosэто MS/Windows сервис
обмена.
#    Блокируем MS/Windows hosts2 запросы сервера имен на
порту 81
   $cmd 00320 deny tcp from  any to any 137 in via  $pif
   $cmd 00321 deny tcp from  any to any 138 in via  $pif
   $cmd 00322 deny tcp from  any to any 139 in via  $pif
   $cmd 00323 deny tcp from any to any 81 in via $pif
#     Запрещаем любые опоздавшие пакеты
   $cmd 00330 deny all from any to any frag in via $pif
#    Запрещаем ACK пакеты, которые не соответствуют ди-
намической таблице правил.
   $cmd 00332 deny tcp from any to any established in via
$pif
```

```
#     Разрешаем входящий трафик с DHCP сервера провайдера.
Это правило
#     должно содержать IP адрес DHCP сервера вашего про-
вайдера, поскольку
#     только ему разрешено отправлять пакеты данного типа.
Необходимо только
#     для проводных и DSL соединений. Для 'userppp' со-
единений   с глобальной
#     сетью использовать это правило нет необходимости.
Это тот же IP
   адрес,
#     выбранный и используемый вами в разделе правил для
исходящего трафика.
#     $cmd 00360                    allow udp from any to
x.x.x.x 67 in  via
$pif keep-state
#     Разрешить входящий трафик для www, так как я ис-
пользую сервер
   apache
   $cmd 00400 allow tcp from any to me 80 in via $pif
setup limit srcaddr 2
#     Разрешить входящий трафик безопасных FTP, Telnet и
SCP из глобальной сети
   $cmd 00410 allow tcp from any to me 22 in via $pif
setup limit srcaddr 2
#     Разрешить входящий нешифрованный трафик Telnet из
глобальной сети
#     считается небезопасным, потому что ID и PW переда-
ются через глобальную # сеть в открытом виде.
#     Удалите этот шаблон, если вы не используете telnet.
   $cmd 00420 allow tcp from any to me 23 in via $pif
setup limit srcaddr 2
#     Отбрасываем и заносим в журнал все входящие соеди-
нения снаружи
   $cmd 00499 deny log all from any to any in via $pif #
Всё остальное запрещено по умолчанию
#     Запрещаем и заносим в журнал все пакеты для даль-
нейшего анализа
   $cmd 00999 deny log all from any to any

   ###############     Конец     файла     правил     IPFW
####################
```

Пример правил с сохранением состояний и поддержкой NAT

Здесь перечислены некоторые дополнительные конфигурационные параметры, которые нужно включить, чтобы активировать функцию NAT в IPFW. В файл конфигурации ядра к остальным параметрам IP-FIREWALL нужно добавить строку *option* IPDIVERT .

В дополнение к обычным параметрам IPFW в */etc/rc.conf* добавим следующее:

```
   natd_enable="YES" # Включить функцию NATD
   natd_interface="rl0" # Название внешнего сетевого ин-
терфейса
   natd flags="-dynamic -m" # -m = по возможности сохра-
нить номера портов
```

Использование динамических правил с правилом *divertnatd* (NetworkAddressTranslation) значительно затрудняет логику составления правил. Расположение *check-state* и *divertnatd* в таблице правил влияет на поведение межсетевого экрана. Это уже не просто последовательный логический поток. При применении вышеозначенных параметров становится доступным новый тип действия *skipto*. При использовании *skipto* нумерация правил становится обязательной. В качестве аргумента *skipto* используется номер правила, к которому нужно перейти.

Ниже последует пример метода кодирования, не снабженный комментариями, приведенный здесь для внесения ясности относительно последовательности прохождения пакетов через набор правил.

Обработка правил начинается с первого по счету и идет последовательно, по правилу за раз, до достижения конца файла, либо если проверяемый пакет соответствует критериям фильтрации; в последнем случае пакет покидает межсетевой экран. Для правил под номерами 100, 101, 450, 500 и 510 важен порядок их расположения. Эти правила управляют трансляцией исходящих и входящих пакетов,

таким образом в таблицу *keep- state* заносятся только приватные IP адреса локальной сети. Обратите внимание, что все правила allow и deny указывают направление, по которому передается пакет (исходящее или входящее) и сетевой интерфейс. Также стоит отметить, что все запросы

начала исходящей сессии передаются с использование *мskipto rule* 500 для трансляции адресов.

Предположим, что пользователь локальной сети запрашивает страницу через браузер. Веб-страницы передаются по порту 80. Пакет входит в межсетевой экран. Этот пакет не попадает под правило 100, потому что в критериях фильтрации этого правила указан параметр *in*. Этот пакет не попадает под правило 101, потому что это первый пакет сессии и он еще не был занесен в динамическую таблицу *keep-state*. Достигнув, наконец, правила 125, пакет удовлетворяет всем критериям фильтрации. Этот пакет является выходящим из интерфейса, взаимодействующим с глобальной сетью. На данном этапе у пакета в качестве исходящего адреса всё еще указан приватный IP адрес локальной сети. По условию этого правила к пакету применяются два действия. Параметр *keep-state* создаст новую запись в динамической таблице *keep-state,* и выполнится действие, указанное в правиле. Указанное действие является частью информации, заносимой в динамическую таблицу. В данном случае это *skipto rule* 500 . Правило 500 транслирует (NAT) адреса пакета и отпускает его наружу. Данное замечание очень важно. Этот пакет идет к цели, где генерируется ответный пакет и отправляется обратно. Этот новый пакет входит в начало списка правил. На этот раз пакет соответствует правилу 100 и его IP адрес назначения транслируется обратно на соответствующий IP адрес локальной сети. Затем он обрабатывается правилом *check-state,* и поскольку для него уже присутствует в динамической таблице правило, соответствующее данной сессии, пакет пропускается в локальную сеть. Дальше пакет приходит к отправившему его компьютер у локальной сети, и генерируется новый пакет, запрашивающий новую порцию данных с удаленного сервера. На этот раз пакет сразу проверяется правилом *check state,* и в случае присутствия исходящей записи данного пакета
выполняется действие skipto 500 . Пакет переходит к правилу 500, транслируется и пропускается во внешнюю сеть.

Для входящего трафика все пакеты, являющиеся частью уже установленной сессии, автоматически разбираются правилом *checkstate* и правильно расположенными правилами *divertnatd.* Всё, что нам остается сделать, это запретить все плохие пакеты и разрешить

прохождение внутрь сети пакетов только для разрешенных сервисов. Допустим, на сервере с межсетевым экраном запущен *apache*, и мы хотим разрешить людям из глобальной сети доступ на локальный вебсайт. Новый входящий пакет, запрашивающий начало сессии, соответствует правилу 100, и его IP адрес транслируется как локальный IP системы с межсетевым экраном. Далее пакет проверяется на соответствие вредоносному трафику и в случае отсутствия соответствия попадает на правило 425. В случае соответствия данному правилу происходят две вещи. Пакет правил помещается в динамическую таблицу *keep-state*, но в этот раз любая новая сессия запросов, порожденных с этого IP, ограничена 2 одновременными соединениями. Это защищает от перегрузки сервис, работающей по указанному номеру порта. В качестве действия в правиле указан *allow*, следовательно пакет пропускается в локальную сеть. Пакет, сформированный в качестве ответа, попадает под *checkstate* и распознается им как принадлежащий существующей сессии. Далее он передаётся на правило 500, где происходит обратная трансляция, после чего пакет пропускается на внешний интерфейс.

Пример файла правил #1:

```
#!/bin/sh cmd="ipfw -q add" skip="skipto 500" pif=rl0
ks="keep-state"
good_tcpo="22,25,37,43,53,80,443,110,119" ipfw -q -f
flush
$cmd 002 allow all from any to any via xl0
#Разрешаем трафик на локальном интерфейсе
$cmd 003 allow all from any to any via lo0
#разрешаем трафик на интерфейсе loopback
$cmd 100 divert natdip from any to any in via $pif
$cmd 101 check-state
# Разрешенные исходящие пакеты
$cmd 120 $skip udp from any to xx.168.240.2 53 out via
$pif $ks
$cmd 121 $skip udp from any to xx.168.240.5 53 out via
$pif $ks
$cmd 125 $skip tcp from any to any $good_tcpo out via
$pif setup $ks
$cmd 130 $skip icmp from any to any out via $pif $ks
$cmd 135 $skip udp from any to any 123 out via $pif
```

```
# Запрещаем весь входящий трафик с не маршрутизируемых
адресных пространств
$cmd 300 deny all from 192.168.0.0/16 to any in via
$pif #RFC 1918
Для локальных IP
$cmd 301 deny all from 172.16.0.0/12 to any in via $pif
#RFC 1918
Для локальных  IP
$cmd 302 deny  all from10.0.0.0/8 to any invia $pif
#RFC 1918 для  локальных IP
$cmd 303 deny  all from127.0.0.0/8 to any invia $pif
#loopback
$cmd 304  deny all from  0.0.0.0/8 to any in via $pif
#loopback
$cmd 305  deny all from  169.254.0.0/16 to any in via
$pif #DHCP авто- конфигурации
$cmd 306  deny all from  192.0.2.0/24 to any in via
$pif #Зарезервировано для документации
$cmd 307 deny all from 204.152.64.0/23 to any in via
$pif #Sun cluster
$cmd 308 deny all from 224.0.0.0/3 to any in via $pif
#  Class D& E multicast # Разрешаем входящие пакеты
$cmd 400  allow udp from xx.70.207.54 to any 68 in $ks
$cmd 420  allow tcp from any to me 80 in via $pif setup
limit srcaddr 1
$cmd 450  deny log ip from any to any
#  Раздел skipto для правил с сохранением состояния для
исходящих пакетов
$cmd 500  divert natdip from any to any out via $pif
$cmd 510  allow ip from any to any
#  ############ Окончание файла правил #############
```

Следующий пример во многом повторяет то, что приведено выше, но использует самодокументирующий стиль записи с исчерпывающими комментариями для то- го, чтобы помочь начинающему составителю правил IPFW лучше понимать, для че- го предназначено то или иное правило.

Пример файла правил #2:

```
#!/bin/sh
############## Начало файла правил IPFW
#####################
```

```
# Сброс всех правил перед началом работы скрипта.
ipfw -q -f flush
# Задание стандартных переменных cmd="ipfw -q add"
skip="skipto 800" pif="rl0" # название внешнего интер-
фейса, # принадлежащего глобальной сети
########################################################
#####
# Нет ограничений на внутреннем интерфейсе локальной
сети
# Замените xl0 на название интерфейса вашей локальной
сети
########################################################
#####
$cmd 005 allow all from any to any via xl0
########################################################
###
#####
# Нет ограничений на интерфейсе Loopback
########################################################
#####
$cmd 010 allow all from any to any via lo0
########################################################
###
#####
# Трансляция адреса, если пакет является входящим
########################################################
#####
$cmd 014 divert natdip from any to any in via $pif
########################################################
###
#####
# Разрешить пакет, если он был ранее добавлен в динами-
ческую # таблицу при помощи выражения allowkeep- state
########################################################
####
$cmd 015 check-state
########################################################
###
#####
# Раздел правил для исходящего трафика на внешнем ин-
терфейсе
# Анализ запросов начала сессии, идущих из-за
межсетевого экрана
# в локальную сеть или от этого шлюза в интернет.
########################################################
```

```
###
#####
#  Разрешить исходящий трафик к DNS серверу провайдера
#  x.x.x.x должен быть IP адресом DNS сервера вашего
провайдера
# Продублируйте эти строки, если у вас больше одного
DNS сервер
# Эти IP адреса можно взять из файла /etc/resolv.conf
$cmd 020 $skip tcp from any to  x.x.x.x 53 out via $pif
setup keepstate
# Разрешить исходящий трафик к DHCP серверу провайдера
для cable
DSL конфигураций.
$cmd 030 $skip udp from any to x.x.x.x 67 out via $pif
keep-state
# Разрешить исходящий трафик для незащищенного www со-
единения
$cmd 040 $skip tcp from any to any 80 out via $pif
setup keep-state
# Разрешить исходящий трафик для защищенного www соеди-
нения
# https с поддержкой TLS и SSL
$cmd 050 $skip tcp from any to any 443 out via $pif
setup keep-state
# Разрешить исходящий POP/SMTP
$cmd  060 $skip tcp from any to any 25 out via $pif
setup keep-state
$cmd  061 $skip tcp from any to any 110 out via $pif
setup keep-state
# Разрешить исходящий трафик для FreeBSD (makeinstall&
CVSUP)
# По сути назначаем пользователю root полные привиле-
гии.
$cmd 070 $skip tcp from me  to any out via $pif setup
keep-state uid root
# Разрешаем исходящий icmp ping
$cmd 080 $skip icmp from any  to any out via $pif keep
state
# Разрешаем исходящий трафик Time
$cmd 090 $skip tcp from any to any 37 out via $pif
setup keep-state
# Разрешаем исходящий трафик nntpnews (т.е. newsgroups)
$cmd 100 $skip tcp from any to any 119 out via $pif
setup keep-state
# Разрешаем исходящий защищённый трафик FTP, Telnet и
```

```
SCP
# Эта функция использует SSH (secureshell)
$cmd 110 $skip tcp from any to any 22  out via $pif
setup keep-state
# Разрешаем исходящий трафик whois
$cmd 120 $skip tcp from any to any 43  out via $pif
setup keep-state
# Разрешаем исходящий трафик ntp
$cmd 130 $skip udp from any to any 123 out via $pif
keep-state
########################################################
###
#####
# Раздел правил для входящего трафика на внешнем интер-
фейсе
# Анализ пакетов, приходящих из глобальной сети,
# предназначенных для этого шлюза или локальной сети
########################################################
####
# Запрещаем весь входящий трафик с немаршрутизируемых
сетей
$cmd 300 deny all from 192.168.0.0/16 to any in via
                       $pif
                       #RFC 1918 private IP
$cmd 301 deny all  from 172.16.0.0/12 to any in via $pif
#                        RFC1918 private IP
$cmd 302 deny all  from 10.0.0.0/8 to any in via $pif
#                        RFC1918 private IP
$cmd 303 deny all  from 127.0.0.0/8 to any in via $pif
                       #loopback
$cmd  304  deny all from 0.0.0.0/8 to any in via $pif
                       #loopback
$cmd  305  deny all from 169.254.0.0/16 to any in via
                       $pif
#                        DHCP auto-config
$cmd 306 deny all from 192.0.2.0/24 to any in via $pif
                       #reserved for docs
$cmd 307 deny all from 204.152.64.0/23 to any in via
                       $pif
                        #Sun cluster
$cmd 308 deny all from 224.0.0.0/3 to any in via $pif
                        #Class D & E multicast
# Запрещаем ident
$cmd 315 deny tcp from any to any 113 in via $pif
# Запрещаем все Netbios службы. 137=name, 138=datagram,
```

```
139=session
#   Netbiosэто MS/Windows сервис обмена.
# Блокируем MS/Windows hosts2 запросы сервера имен на
порту 81
$cmd 320 deny tcp  from anyto  any137 in  via$pif
$cmd 321 deny tcp  from anyto  any138 in  via$pif
$cmd 322 deny tcp  from anyto  any139 in  via$pif
$cmd 323 deny tcp  from any to  any 81 in via $pif
# Запрещаем любые опоздавшие пакеты
$cmd 330 deny all  from any toany frag in via $pif
# Запрещаем ACK пакеты, которые не соответствуют дина-
мической таблице правил.
$cmd 332 deny tcp from any to  any established in via
$pif
# Разрешаем входящий трафик с DHCP сервера провайдера.
Это правило
# должно содержать IP адрес DHCP сервера вашего
провайдера, поскольку
# только ему разрешено отправлять пакеты данного типа.
Необходимо только
# для проводных и DSL соединений. Для 'userppp'
соединений с глобальной
#   сетью использовать это правило нет  необходимости.
Это тот же IP адрес,
# выбранный и используемый вами в разделе правил для
исходящего трафика.
$cmd 360 allow udp from x.x.x.x  to any 68 in via $pif
keep-state
# Разрешить входящий трафик для www, т.к. я использую
Apache сервер.
$cmd 370 allow tcp from any  to me 80 in via $pif setup
limit srcaddr 2
# Разрешить входящий трафик безопасных FTP, Telnet и
SCP из глобальной сети
$cmd 380 allow tcp from any  to me 22 in via $pif setup
limit srcaddr 2
# Разрешить входящий нешифрованный трафик Telnet из
глобальной сети
# считается небезопасным, потому что ID и PW передаются
через глобальную
#   сеть в открытом виде.
# Удалите этот шаблон, если вы не используете telnet.
$cmd 390 allow tcp from any to  me 23 in via $pif setup
limit srcaddr  2
# Отбрасываем и заносим в журнал все неразрешенные
```

```
входящие
Соединения из  глобальной сети
$cmd 400 deny log all from any  to any in via $pif
#  Отбрасываем и заносим в журнал все неразрешенные
исходящие
Соединения в глобальную сеть
$cmd 450 deny log all from any to any out via $pif
#  Место для skipto в правилах с сохранением состояния
для исходящих соединений
$cmd  800 divert natdip from any to any out via $pif
$cmd  801 allow ip from any to any
#  Всё остальное запрещено по умолчанию
#  Запрещаем и заносим в журнал все пакеты для
дальнейшего анализа
$cmd 999 deny  log all from any to any
#  ############# Окончание файла правил IPFW
   #   ###################
```

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Порядок выполнения:

1. Изучить краткий теоретический материал.
2. Включить IPWF
3. Указать тип межсетевого экрана.
4. Вывести полный список существующих правил
5. Включить протоколирование сообщений межсетевого экрана
6. Задать правило с сохранением состояния
7. Задать правило без сохранения состояния.
8. Написать скрипт правил по предоставленному примеру.
9. Написать правила для межсетевого экрана закрытого типа.
10. Написать правила с сохранением состояний и поддержкой NAT.
11. После установки каждого правила необходимо проверить, что правила работают корректно (попытаться обратиться по сети к другому компьютеру)
12. Завершить работу с FreeBSD.
13. Ответить на контрольные вопросы и оформить отчет.

ФОРМА ОТЧЕТА О ВЫПОЛНЕННОЙ РАБОТЕ

Отчет на защиту предоставляется в печатном или электронном виде.

Структура отчета (на отдельном листе(-ах)): титульный лист, формулировка задания (вариант), этапы выполнения работы (со скриншотами), результаты выполнения работы (скриншоты и содержимое файлов).

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Опишите назначение межсетевого экрана.
2. Назовите задачи, которые выполняет межсетевой экран.
3. Опишите принцип работы межсетевого экрана.
4. Назовите существующие пакеты межсетевого экрана.
5. Опишите синтаксис правил межсетевого экрана.
6. Дайте определение NAT.
7. Охарактеризуйте понятие «Правило с сохранением состояния»
8. Охарактеризуйте понятие «Правило без сохранения состояния»
9. Изложите концепцию межсетевого экрана открытого типа.
10. Изложите концепцию межсетевого экрана закрытого типа.
11. Объясните, как включить IPWF.
12. Опишите процесс настройки межсетевого экрана.

ЛАБОРАТОРНАЯ РАБОТА №3
НАСТРОЙКА ПОЧТОВОГО СЕРВЕРА

ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Целью выполнения лабораторной работы является получение практических навыков по настройке почтового сервера в среде ОС FreeBSD.

Основными задачами выполнения лабораторной работы являются:

1. Научиться настраивать почтовый сервер Sendmail под ОС FreeBSD

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

«Электронная почта» называемая также email, является на сегодняшний день одним из самых популярных средств связи. В данной лабораторной работе рассматриваются основы работы с почтовым сервером в FreeBSD, а также введение в процесс отправки и получения почты в FreeBSD.

В работе почтовой системы задействованы пять основных частей: пользовательский почтовый клиент (Mail User Agent, MUA), почтовый сервис (демон) (Mail Transfer Agent, MTA), сервер DNS, удаленный или локальный почтовый ящик, и конечно сам почтовый сервер.

Пользовательский почтовый клиент

Обычно, это программа типа mutt, pine, elm, mail, а также программы с графическим интерфейсом, такие, как balsa или xfmail, или интегрированные приложения (например, какой-либо WWW браузер типа Netscape). Все эти программы общаются с

локальным почтовым сервером, вызывая какой-либо демон, или напрямую по протоколу TCP.

Почтовый демон

FreeBSD по умолчанию поставляется с sendmail, но помимо того поддерживает множество других демонов почтового сервера, вот лишь некоторые из них:

- exim;
- postfix;
- qmail.

Почтовый демон выполняет только две функции: он отвечает за прием входящей почты и отправку исходящей. Он не отвечает за выдачу почты по протоколам POP или IMAP, и не обеспечивает подключения к локальным почтовым ящикам mbox или Maildir. Для этих целей вам может потребоваться дополнительный демон.

Предупреждение

Старые версии sendmail содержат некоторые серьезные ошибки безопасности, которые могут привести к получению атакующим локального и/или удаленного доступа к вашему компьютеру. Убедитесь, что вы работаете с современной версией, свободной от таких ошибок. Или установите альтернативный MTA из Коллекции Портов FreeBSD.

Email и DNS

Служба имен доменов (Domain Name System, DNS) и соответствующий ей даемон named играют важную роль в доставке почты. Для доставки почты с вашего сайта другому, даемон почтового сервера обратится к DNS для определения удаленного хоста, отвечающего за доставку почты по назначению. Тот же процесс происходит при доставке почты с удаленного хоста на ваш почтовый сервер.

DNS отвечает за сопоставления имен хостов IP адресам, как и за хранение информации, предназначенной для доставки почты, известной как MX записи. Запись MX (Mail eXchanger) определяет хост или хосты, которые будут получать почту для определенного домена. Если для вашего имени хоста или домена нет записи MX, почта будет доставлена непосредственно на ваш хост, IP адрес которого определен в записи A.

Вы можете просмотреть MX записи для любого домена с помощью команды host, как показано в примере ниже:

```
% host -t mx FreeBSD.org
FreeBSD.org mail is handled (pri=10) by mx1.FreeBSD.org
```

Получение почты

Получение почты для вашего домена выполняет почтовый сервер. Он сохраняет отправленную в ваш домен почту в формате либо mbox (это метод по умолчанию), либо Maildir, в зависимости от настроек. После сохранения почты ее можно либо прочитать локально, используя такие приложения как mail, mutt, или удаленно, по таким протоколам как POP или IMAP. Это означает, что для локального чтения почты вам не потребуется устанавливать сервер POP или IMAP.

4.1. Доступ к удаленным почтовым ящикам по протоколам POP и IMAP

Для удаленного доступа к почтовым ящикам вам потребуется доступ к POP или IMAP серверу. Хотя удаленный доступ обеспечивают оба протокола POP и IMAP, последний предоставляет множество дополнительных возможностей, вот некоторые из них:

- IMAP может как хранить сообщения на удаленном сервере, так и забирать их.
- IMAP поддерживает одновременные обновления.

• IMAP может быть очень полезен для низкоскоростных соединений, поскольку позволяет пользователям получить структуру сообщений без их загрузки; он также может использоваться для выполнения таких задач как поиск на сервере, для минимизации объема передаваемых между клиентом и сервером данных.

Для установки POP или IMAP сервера необходимо выполнить следующие действия:

1. Выберите IMAP или POP сервер, который подходит вам наилучшим образом. Следующие POP и IMAP серверы хорошо известны и могут быть приведены в качестве примера:
   - qpopper;
   - teapop;
   - imap-uw;
   - courier-imap;

2. Установите POP или IMAP даемон, выбранный из Коллекции Портов.
3. Если потребуется, настройте /etc/inetd.conf для запуска POP или IMAP сервера.

Предупреждение

Необходимо отметить, что и POP и IMAP серверы передают информацию, включая имя пользователя и пароль, в незашифрованном виде. Это означает, что если вы хотите защитить передачу информации по этим протоколам, потребуется использовать туннелирование сессий через ssh или при помощи SSL.

Доступ к локальным почтовым ящикам

Доступ к почтовым ящикам может быть осуществлен непосредственно путем использования MUA на сервере, где эти

ящики расположены. Это можно сделать используя приложения вроде mutt или mail.

Почтовый хост

Почтовый хост это сервер, который отвечает за отправку и получение почты для вашего компьютера, и возможно, для всей вашей сети.

Настройка sendmail

В FreeBSD по умолчанию программой передачи почты (Mail Transfer Agent, MTA) является sendmail. Работа sendmail заключается в приеме почты от почтовых программ пользователей (Mail User Agents, MUA) и отправке ее на соответствующий адрес, в соответствии с имеющимися настройками. sendmail может также принимать входящие соединения по сети и доставлять почту в локальные почтовые ящики или перенаправлять их другой программе. sendmail использует следующие файлы настройки:

Имя файла	Назначение
/etc/mail/access	Файл базы данных доступа sendmail
/etc/mail/aliases	Синонимы почтовых ящиков
/etc/mail/local-host-names	Список хостов, для которых sendmail принимает почту
/etc/mail/mailer.conf	Настройки почтовой программы
/etc/mail/mailertable	Таблица доставки почтовой программы
/etc/mail/sendmail.cf	Основной файл настройки sendmail
/etc/mail/virtusertable	Таблицы виртуальных пользователей и доменов

/etc/mail/access

База данных доступа определяет список хостов или IP адресов, имеющих доступ к локальному почтовому серверу, а также тип предоставляемого доступа. Хосты могут быть перечислены как OK, REJECT, RELAY или просто переданы процедуре обработки

ошибок sendmail с заданным сообщением об ошибке. Хостам, перечисленным с параметром по умолчанию OK, разрешено отправление почты на этот хост, если адрес назначения почты принадлежит локальной машине. Все почтовые соединения от хостов, перечисленных с параметром REJECT, отбрасываются. Для хостов, перечисленных с параметром RELAY, разрешена передача через этот сервер почты с любым адресом назначения.

Пример 2.1. - Настройка базы данных доступа sendmail

cyberspammer.com 550 We do not accept mail from spammers
FREE.STEALTH.MAILER@ 550 We do not accept mail from spammers
another.source.of.spam REJECT
okay.cyberspammer.com OK
128.32 RELAY

В этом примере приведены пять записей. К отправителям, чей адрес соответствует записи в левой части таблицы, применяется правило записанное в правой части таблицы. В первых двух примерах код ошибки будет передан процедуре обработке ошибок sendmail. В этом случае на удаленном хосте будет получено соответствующее сообщение. В следующем примере почта отбрасывается от определенного хоста, another.source.of.spam. В четвертом примере разрешается прием почты от хоста okay.cyberspammer.com, имя которого более точно совпадает с этой записью, чем с cyberspammer.com в примере выше. При более точном совпадении правила перезаписываются. В последнем примере разрешается пересылка почты от хостов с IP адресами, начинающимися с 128.32. Эти хосты смогут отправлять почту через этот почтовый сервер для других почтовых серверов.

После изменения этого файла для обновления базы данных вам потребуется запустить make в каталоге /etc/mail/.

/etc/mail/aliases

База данных синонимов содержит список виртуальных почтовых ящиков, принадлежащих другим пользователям, файлам, программам, или другим синонимам. Вот несколько примеров, которые могут быть использованы для /etc/mail/aliases:

Пример 2.2. - Mail Aliases

root: localuser
ftp-bugs: joe,eric,paul
bit.bucket: /dev/null
procmail: "|/usr/local/bin/procmail"

Формат файла прост; имя почтового ящика слева от двоеточия сопоставляется назначению(ям) справа. В первом примере производится простое сопоставление почтового ящика root почтовому ящику localuser, для которого затем опять будет произведен поиск в базе данных синонимов. Если совпадений не обнаружится, сообщение будет доставлено локальному пользователю localuser. В следующем примере приведен список рассылки. Почта на адрес ftp-bugs рассылается на три локальных почтовых ящика: joe, eric и paul. Обратите внимание, что удалённый почтовый ящик может быть задан в виде <user@example.com>. В следующем примере показана запись почты в файл, в данном случае /dev/null. И в последнем примере показано отправление почты программе, в данном случае почтовое сообщение переправляется через канал UNIX® на стандартный вход /usr/local/bin/procmail.

После обновления этого файла вам потребуется запустить make в каталоге /etc/mail/ для обновления базы данных.

6.3. /etc/mail/local-host-names

В этом файле находится список имен хостов, принимаемых программой sendmail в качестве локальных. Поместите в этот файл

любые домены или хосты, для которых sendmail должен принимать почту. Например, если этот почтовый сервер должен принимать почту для домена example.com и хоста mail.example.com, его файл local-hostnames может выглядеть примерно так:

example.com
mail.example.com

После обновления этого файла необходимо перезапустить sendmail, чтобы он смог перечитать изменения.

/etc/mail/sendmail.cf

Основной файл настройки sendmail, sendmail.cf управляет общим поведением sendmail, включая все, от перезаписи почтовых адресов до отправки удаленным серверам сообщений об отказе от пересылки почты. Конечно, файл настройки с таким многообразием возможностей очень сложен и подробное его описание выходит за рамки данного раздела. К счастью, для стандартных почтовых серверов изменять этот файл придется не часто.

Основной файл настройки sendmail может быть собран из макроса m4, определяющего возможности и поведение sendmail. Подробнее этот процесс описан в файле

/usr/src/contrib/sendmail/cf/README.

Для применения изменений после правки файла необходимо перезапустить sendmail.

/etc/mail/virtusertable

Файл virtusertable сопоставляет виртуальные почтовые домены и почтовые ящики реальным почтовым ящикам. Эти почтовые ящики могут быть локальными, удаленными, синонимами, определенными в /etc/mail/aliases, или файлами.

Пример - Пример таблицы виртуального домена

root@example.com	*root*
postmaster@example.com	*postmaster@noc.example.net*
@example.com	*joe*

В примере выше мы видим сопоставление адресов для домена example.com. Почта обрабатывается по первому совпадению с записью в этом файле. Первая запись сопоставляет адрес <root@example.com> локальному почтовому ящику root. Вторая запись сопоставляет <postmaster@example.com> локальному почтовому ящику postmaster на хосте noc.example.net. Наконец, до этого момента адрес в домене example.com не совпал ни с одним из предыдущих, будет применено последнее сопоставление, которому соответствует всякое другое почтовое сообщение, отправленное на любой адрес в example.com. Это сообщение будет доставлено в локальный почтовый ящик joe.

Установка другой почтовой программы

Как уже упоминалось, FreeBSD поставляется с MTA (Mail Transfer Agent) sendmail. Следовательно, по умолчанию именно эта программа отвечает за вашу исходящую и входящую почту.

Однако, по различным причинам некоторые системные администраторы заменяют системный MTA. Эти причины варьируются от простого желания попробовать другой MTA до потребности в определенных возможностях пакета, основанного на другой почтовой программе. К счастью, вне зависимости от причины, в FreeBSD такая замена выполняется просто.

Установка нового MTA

Вам предоставлен широкий выбор MTA. Начните с поиска в Коллекции Портов FreeBSD, где их немало. Конечно, вы можете использовать любой MTA по желанию, взятый откуда угодно, если только сможете запустить его под FreeBSD.

Начните с установки нового MTA. После установки у вас будет возможность решить, действительно ли он подходит вашем нуждам, а также настроить новое программное обеспечение перед тем, как заменить им sendmail. При установке новой программы убедитесь, что она не пытается перезаписать системные файлы, такие как /usr/bin/sendmail. Иначе ваша новая почтовая программа фактически начнет работать до того, как вы ее настроите.

Обратитесь к документации на выбранный MTA за информацией по его настройке.

Отключение sendmail

Предупреждение

Если вы отключите сервис исходящей почты sendmail, необходимо заменить его альтернативной системой доставки почты. Если вы не сделаете этого, системные программы, такие как periodic, не смогут отправлять сообщения по электронной почте как обычно. Многие программы в вашей системе могут требовать наличия функционирующей sendmail-совместимой системы. Если приложения будут продолжать использовать программу sendmail для отправки почты после того, как вы её отключили, почта может попасть в неактивную очередь sendmail и никогда не будет доставлена.

Для полного отключения sendmail, включая сервис исходящей почты, используйте

```
sendmail_enable="NO"
sendmail_submit_enable="NO"
 sendmail_outbound_enable="NO"
 sendmail_msp_queue_enable="NO"
```

в /etc/rc.conf.

Если вы хотите отключить только сервис входящей почты sendmail, установите

sendmail_enable="NO"

в /etc/rc.conf. Дополнительная информация о параметрах запуска sendmail доступна на странице справочника rc.sendmail.

Запуск нового MTA при загрузке

Новый MTA можно запускать автоматически при загрузке системы добавив соответствующую строку в /etc/rc.conf . Ниже приведен пример для postfix:

 # echo 'postfix _enable=«YES»' >> /etc/rc.conf

С этого момента MTA будет запускаться автоматически во время загрузки системы.

Замещение sendmail как почтовой программы по умолчанию

Программа sendmail настолько распространена в качестве стандартной программы для систем UNIX®, что многие программы считают, что она уже установлена и настроена. По этой причине многие альтернативные MTA предоставляют собственные совместимые реализации интерфейса командной строки sendmail; это облегчает их использование в качестве "прозрачной" замены sendmail.

Поэтому если вы используете альтернативную почтовую программу, потребуется убедиться, что когда программное обеспечение пытается выполнить стандартные исполняемые файлы sendmail, такие как /usr/bin/sendmail, на самом деле выполняются программы вновь установленной почтовой системы. К счастью, FreeBSD предоставляет систему, называемую mailwrapper, которая выполняет эту работу за вас.

Когда установлен sendmail, файл /etc/mail/mailer.conf выглядит примерно так:

sendmail /usr/libexec/sendmail/sendmail
send-mail /usr/libexec/sendmail/sendmail
mailq /usr/libexec/sendmail/sendmail
newaliases /usr/libexec/sendmail/sendmail
hoststat /usr/libexec/sendmail/sendmail
purgestat /usr/libexec/sendmail/sendmail

Это означает, что когда выполняется какая-то из этих стандартных программ (например сам sendmail), система на самом деле вызывает копию mailwrapper, называемую sendmail, которая обращается к mailer.conf и выполняет вместо этого /usr/libexec/sendmail/sendmail. Такая схема делает простой замену программ, которые на самом деле выполняются, когда вызываются стандартные функции sendmail.

Поэтому если вы хотите выполнять /usr/local/supermailer/bin/sendmail-compat вместо sendmail, отредактируйте /etc/mail/mailer.conf так:

Sendmail /usr/local/supermailer/bin/sendmail-compat
send-mail /usr/local/supermailer/bin/sendmail-compat
mailq /usr/local/supermailer/bin/mailq-compat
newaliases /usr/local/supermailer/bin/newaliases-compat
hoststat /usr/local/supermailer/bin/hoststat-compat
purgesta t /usr/local/supermailer/bin/purgestat-compat

Запуск новой почтовой программы

Как только вы все настроили, потребуется или уничтожить процесс sendmail, который уже не нужен и запустить новую почтовую программу, или просто перегрузить систему. Перезагрузка также даст вам возможность проверить, правильно

ли настроена система для автоматического запуска MTA при загрузке.

Поиск и устранение неисправностей

. Использование FQDN для хостов вне моей подсети.

Вы, видимо, обнаружили, что хост, к которому вы обратились, оказался на самом деле в другом домене; например, если вы находитесь в домене foo.bar.edu и хотите обратиться к хосту mumble в домене bar.edu, то должны указать его полное доменное имя, mumble.bar.edu, а не просто mumble.

Традиционно, программа разрешения имен BSD BIND позволяла это делать. Однако, текущая версия BIND, поставляемая с FreeBSD, больше не добавляет имена доменов, отличающихся от того, в котором вы находитесь, для не полностью указанных имен хостов. То есть, имя mumble будет опознан как mumble.foo.bar.edu или будет искаться в корневом домене.

Это отличается от предыдущего поведения, при котором поиск продолжался в доменах mumble.bar.edu и mumble.edu. Если вам интересны причины объявления такого поведения плохой практикой и даже ошибкой в безопасности, обратитесь к RFC 1535.

Хорошим решением будет поместить строку

search foo.bar.edu bar.edu

вместо ранее используемой:

domain foo.bar.edu

в файл /etc/resolv.conf. Однако удостоверьтесь, что порядок поиска не нарушает "границ полномочий между локальным и внешним администрированием", в терминологии RFC 1535.

sendmail выдает ошибку “mail loops back to myself”

В FAQ по sendmail дан следующий ответ:

- Я получаю такие сообщения об ошибке:

553MX list for domain.net points back to relay.domain.net
554 <user@domain.net>... Local configuration error
Как можно решить эту проблему?

- Согласно записям MX, почта для домена domain.net перенаправляется на хост relay.domain.net, однако последний не распознается как domain.net. Добавьте domain.net в файл /etc/mail/local-host-names [известный как /etc/sendmail.cw до версии 8.10] (если вы используете FETURE(use_cw_file)) или добавьте "Cw domain.net" в файл /etc/mail/sendmail.cf.

FAQ по sendmail можно найти на http://www.sendmail.org/faq/ и рекомендуется прочесть его при желании произвести некоторые "усовершенствования" настроек почтовой системы.

Организация работы почтового сервера при коммутируемом соединении с Интернет

Вы хотите подключить к интернет компьютер с FreeBSD, работающий в локальной сети. Компьютер с FreeBSD будет почтовым шлюзом для локальной сети. PPP соединение не выделенное.

Существует как минимум два пути, чтобы сделать это. Один способ это использование UUCP.

Другой способ это использование постоянно работающего интернет сервера для обеспечения вторичного MX сервиса вашего домена. Например, домен вашей компании example.com, и провайдер интернет настроил example.net для обеспечения вторичного MX сервиса:

```
example.com.  MX    10    example.com.
              MX    20    example.net.
```

Только один хост должен быть указан в качестве последнего получателя (добавьте запись Cw example.com в файл /etc/mail/sendmail.cf на машине example.com).

Когда программа sendmail (со стороны отправителя) "захочет" доставить почту, она попытается соединиться с вашим хостом (example.com) через модемное подключение. Скорее всего, ей это не удастся (вы, вероятнее всего, не будете подключены к интернет). Программа sendmail автоматически перейдет ко вторичному MX серверу, т.е. вашему провайдеру (example.net). Вторичный MX сервер будет периодически пытаться соединиться с вашим хостом и доставить почту на основной сервер MX (example.com).

Вы можете воспользоваться следующим сценарием, чтобы забирать почту каждый раз, когда вы входите в систему:

```
#!/bin/sh
# Put me in /usr/local/bin/pppmyisp
( sleep 60 ; /usr/sbin/sendmail -q ) &
/usr/sbin/ppp -direct pppmyisp
```

Если же вы хотите написать отдельный пользовательский скрипт, лучше воспользоваться командой sendmail -qRexample.com вместо вышеприведенного сценария, так как в этом случае вся почта в очереди для хоста example.com будет обработана немедленно.

Рассмотрим эту ситуацию подробнее:

Пример 4.1. - Сообщение из freebsd-isp

> Мы предоставляем вторичный MX для наших клиентов. Вы соединяетесь

> с нашим сервером несколько раз в день, чтобы забрать почту для вашего

> первичного (главного) MX (мы не соединяемся с ним каждый раз, когда

> приходит новая почта для его доменов). Далее, sendmail отправляет

> почту, находящуюся в очереди каждые 30 минут, и клиент должен быть > подключен к Интернет в течении 30 минут, чтобы удостовериться, что

> вся почта "ушла" на основной MX-сервер.

>

> Может быть, есть какая-либо команда, которая заставит sendmail

> немедленно отправить все почту, находящуюся в очереди? Естественно,

> пользователи не обладают какими-либо повышенными привилегиями на

> нашем сервере.

В разделе "privacy flags" файла sendmail.cf, определяется опция

Opgoaway,restrictqrun

Уберите restrictqrun, чтобы разрешить рядовым пользователям инициировать работу с очередью. Вам также может понадобиться изменить порядок MX-серверов. Так, если вы предоставляете первый (основной)

MX-сервер для ваши пользователей, мы указываем:

 # If we are the best MX for a host, try directly instead of generating

 # local config error.

OwTrue

Таким образом, удаленный хост будет доставлять почту непосредственно к вам, не пытаясь установить соединение с клиентом. Затем уже вы, в свою очередь, отсылаете ее клиенту. Удостоверьтесь, что в DNS есть записи про

"customer.com" и "hostname.customer.com". Просто добавьте запись A в DNS для "customer.com".

Получение ошибки “Relaying Denied” при отправке почты через другие хосты

В установке FreeBSD по умолчанию, sendmail настроен для отправки почты только от хоста, на котором он работает. Например, если доступен POP сервер, пользователи смогут проверять почту из школы, с работы или других удаленных точек, но не смогут отправлять письма. Обычно, через некоторое время после попытки будет отправлено письмо от MAILER-DAEMON с сообщением об ошибке “5.7 Relaying Denied”.

Есть несколько путей разрешения этой ситуации. Самый прямой путь это использование адреса вашего провайдера в файле relay-domains, расположенном в /etc/mail/relay-domains. Быстрый способ сделать это:

 # echo "your.isp.example.com" > /etc/mail/relay-domains

После создания или редактирования этого файла вы должны перезапустить sendmail. Это отлично работает, если вы администратор сервера и не хотите отправлять почту локально, или хотите воспользоваться почтовым клиентом/системой на другом компьютере или даже через другого провайдера. Это также очень полезно, если у вас настроены одна или две почтовые записи. Если необходимо добавить несколько адресов, вы можете просто открыть этот файл в текстовом редакторе и добавить домены, по одному на строку:

your.isp.example.com
other.isp.example.net
users-isp.example.org
www.example.org

Теперь будет отправляться любая почта, посылаемая через вашу систему любым хостом из этого списка (предоставляемого пользователем, имеющим учетную запись в вашей системе). Это отличный способ разрешить пользователям отправлять почту через

вашу систему удаленно, одновременно он блокирует отправку спама.

РАСШИРЕННОЕ РУКОВОДСТВО

Базовая конфигурация

Изначально, вы можете отправлять почту "во внешний мир" если правильно составлен файл /etc/resolv.conf или запущен свой сервер имен. Если вы хотите, чтобы почта, предназначенная для хоста в вашем домене, доставлялась MTA (например, sendmail) на вашем хосте FreeBSD, есть два пути:

- Запустите свой собственный сервер DNS, тем самым организовав собственный домен, например, FreeBSD.org
- Получайте почту для вашего хоста непосредственно. Это работает при доставке почты непосредственно на DNS имя вашей машины. Например, example.FreeBSD.org.

Независимо от выбранного из предложенных выше вариантов, для доставки почты непосредственно на ваш хост у него должен быть постоянный IP адрес (а не динамический, как у большинства PPP соединений). Если вы находитесь за брандмауэром, то последний должен пропускать SMTP-пакеты. Если вы хотите, чтобы почта приходила непосредственно на ваш хост, необходимо убедиться в одном из двух:

- Убедитесь, что запись (с наименьшим номером) MX в DNS соответствует IP адресу вашего хоста.
- Убедитесь, что в DNS для вашего хоста вообще отсутствует MX-запись.

Выполнение любого из перечисленных условий обеспечит доставку почты для вашего хоста.

Попробуйте это:

 # hostname example.FreeBSD.org
 # host example.FreeBSD.org
example.FreeBSD.org has address 204.216.27.XX

Если вы это видите, то можно без проблем посылать почту на <yourlogin@example.FreeBSD.org> (предполагается, что sendmail на example.FreeBSD.org работает правильно).

Однако, если вы видите это:

 # host example.FreeBSD.org
example.FreeBSD.org has address 204.216.27.XX
example.FreeBSD.org mail is handled (pri=10) by hub.FreeBSD.org

то вся почта, посланная на example.FreeBSD.org будет собираться на hub (для того же пользователя), вместо того, чтобы быть отосланной непосредственно на ваш хост.

Эта информация обрабатывается вашим DNS сервером. Соответствующая запись DNS, указывающая, через какой хост будет проходить ваша почта, называется MX (Mail eXchanger). Если для хоста отсутствует такая запись, почта будет приходить прямо на этот хост.

Допустим, что запись MX для хоста freefall.FreeBSD.org в какой-то момент выглядела так:

freefall MX 30 mail.crl.net
freefall MX 40 agora.rdrop.com
freefall MX 10 freefall.FreeBSD.org
freefall MX 20 who.cdrom.com

Вы видите, что для хоста freefall существуют несколько MX-записей. Запись с наименьшим номером соответствует хосту, получающему почту непосредственно, если он доступен; если он недоступен по каким-то причинам, другие сервера (иногда называемые ("резервными MX") временно получают почту, и

хранят ее пока не станут доступны хосты с меньшими номерами, в конечном итоге отправляя почту на эти хосты.

Чтобы альтернативные MX-хосты использовались наиболее эффективно, они должны быть независимо подключены к Интернет. Ваш провайдер (или дружественный сайт) скорее всего без проблем сможет оказать подобные услуги.

Почта для вашего домена

Для настройки "почтового хоста" (почтовый сервер) вам потребуется, чтобы почта, направляемая различным рабочим станциям, пересылалась этому хосту. Обычно вам необходима доставка всей почты для любого хоста вашего домена (в данном случае *.FreeBSD.org) на почтовый сервер, чтобы пользователи могли получать свою почту с этого сервера.

Чтобы облегчить себе (и другим) жизнь, создайте на обеих машинах учетные записи с одинаковыми именами пользователей, например, с помощью команды adduser.

Сервер, который вы будете использовать в качестве почтового, должен быть объявлен таковым для каждой машины в домене. Вот фрагмент примерной конфигурации:

```
example.FreeBSD.org   A  204.216.27.XX     ;Рабочая станция
                      MX 10 hub.FreeBSD.org ;Почтовый шлюз
```

Таким образом, вся корреспонденция, адресованная рабочей станции, будет обрабатываться вашим почтовым сервером, независимо от того, что указано в A-записи.

Все это можно реализовать только в том случае, если вы используете сервер DNS. Если вы по каким-либо причинам не имеете возможности установить свой собственный сервер имен, необходимо договориться с провайдером или теми, кто поддерживает ваш DNS.

Если вы хотите поддерживать несколько виртуальных почтовых серверов, может пригодиться следующая информация. Допустим, что ваш клиент зарезервировал домен, например, customer1.org, и вам требуется, чтобы почта, предназначенная для customer1.org приходила на ваш хост, например, mail.myhost.com. В таком случае, DNS должен выглядеть так:

customer1.org *MX 10 mail.myhost.com*

Заметьте, что если вам требуется только получать почту для домена, соответствующая Азапись не нужна.

Примечание

Помните, что если вы попытаетесь каким-либо образом обратиться к хосту customer1.org, у вас вряд ли что-либо получится, если нет А-записи для этого хоста.

Последнее, что вы должны сделать - это сказать программе sendmail, для каких доменов и/или хостов она должна принимать почту. Это можно сделать несколькими способами: • Добавьте названия этих хостов в файл /etc/mail/local-host-names, если вы используете FEATURE(use_cw_file). Если у вас sendmail версии ниже 8.10, необходимо отредактировать файл /etc/sendmail.cw.

• Добавьте строку Cwyour.host.com в файл /etc/sendmail.cf или /etc/mail/sendmail.cf (если у вас sendmail версии 8.10 или более поздней).

SMTP через UUCP

астройка поставляемого с FreeBSD sendmail предназначена для сайтов, подключенных к интернет непосредственно. Сайты, осуществляющие обмен почтой через UUCP, должны использовать другой файл настройки sendmail.

Редактирование /etc/mail/sendmail.cf вручную это сложная задача. sendmail версии 8 генерирует файлы настройки через

препроцессор m4, реально настройка выполняется на более высоком уровне абстракции. Файлы настройки m4 можно найти в /usr/share/sendmail/cf. Файл README в каталоге cf содержит введение в основы настройки m4.

Лучшим способом настройки поддержки передачи по UUCP является использование возможности mailertable. При этом создается база данных, которая помогает sendmail решать вопросы маршрутизации.

Во-первых, создайте файл .mc. В каталоге /usr/share/sendmail/cf/cf находятся несколько примеров. Возьмем для примера имя файла foo.mc. Все, что потребуется для преобразования его в sendmail.cf, это:

```
# cd /etc/mail
# make foo.cf
# cp foo.cf /etc/mail/sendmail.cf
```

Типичный .mc файл может выглядеть примерно так:

```
VERSIONID(`Your version number') OSTYPE(bsd9.1)

FEATURE(accept_unresolvable_domains)
FEATURE(nocanonify)
FEATURE(mailertable, `hash -o /etc/mail/mailertable')
define(`UUCP_RELAY',your.uucp.relay)
define(`UUCP_MAX_SIZE',200000)
define(`confDONT_PROBE_INTERFACES')

MAILER(local)
MAILER(smtp)
MAILER(uucp)

Cw      your.alias.host.name
Cw      youruucpnodename.UUCP
```

Строки, содержащие accept_unresolvable_domains, nocanonify, и confDONT_PROBE_INTERFACES, предотвратят использование DNS для доставки почты. Пункт UUCP_RELAY необходим для поддержки доставки по UUCP. Просто поместите сюда имя хоста в интернет, способного работать с .UUCP адресами псевдо-доменов; скорее всего, вы введете сюда основной сервер пересылки почты провайдера.

Как только вы сделаете это, потребуется файл /etc/mail/mailertable. Если вы используете для всей почты только одно внешнее соединение, подойдет следующий файл:

```
# makemap hash /etc/mail/mailertable.db < /etc/mail/mailertable
.                       uucp-dom:your.uucp.relay
```

Более сложный пример может выглядеть так:

```
# makemap hash /etc/mail/mailertable.db < /etc/mail/mailertable
#
horus.interface-business.de     uucp-dom:horus
.interface-business.de      uucp-dom:if-bus
interface-business.de       uucp-dom:if-bus
.heep.sax.de                    smtp8:%1
horus.UUCP                      uucp-dom:horus
if-bus.UUCP                     uucp-dom:if-bus
.                               uucp-dom:
```

В первых трех строках обрабатываются специальные случаи, когда почта для домена должна отправляться не на маршрут по умолчанию, а на ближайшее соединение UUCP для сокращения пути доставки. Следующая строка обрабатывает почту, которая может быть доставлена по SMTP для локального Ethernet домена. Наконец, определены маршруты UUCP в нотации псевдо-доменов .UUCP, для включения перезаписи правил по умолчанию правилом uucp-neighbor !recipient. Последняя строка всегда содержит одиночную точку, означающую "все остальное", с отправкой через UUCP, являющимся универсальным почтовым шлюзом. Все имена

узлов после ключевого слова uucp-dom: должны представлять существующие маршруты UUCP, проверить их можно с помощью команды uuname.

Напоминаем, что этот файл должен быть преобразован в базу данных DBM перед использованием. Командную строку для этой задачи лучше всего поместить в качестве комментария в верхней части файла mailertable. Всегда выполняйте эту команду после правки файла mailertable.

И наконец: если вы не уверены, что некоторые отдельные почтовые маршруты будут работать, запомните параметр sendmail -bt. С этим параметром sendmail запускается в режиме тестирования адреса; просто введите 3,0 и адрес, который вы хотите протестировать. В последней строке появится сообщение об используемом внутреннем почтовом агенте, хосте назначения, с которым вызывается этот агент, и (возможно транслированный) адрес. Выход из этого режима происходит при нажатии Ctrl+D.

% sendmail -bt
ADDRESS TEST MODE (ruleset 3 NOT automatically invoked)
Enter <ruleset> <address>
> 3,0 foo@example.com
canonify input: foo @ example . com
parse returns: $# uucp-dom $@ your.uucp.relay $: foo < @ example . com . >
> ^D

Почтовые программы пользователей

Почтовая программа пользователя (Mail User Agent, MUA) это приложение, используемое для отправки и получения почты. Кроме того, поскольку почта "эволюционирует" и становится более сложной, MUA совершенствуют свои функции по обработке почты, становятся более удобны в использовании. FreeBSD поддерживает множество различных пользовательских почтовых

программ, каждая из которых может быть легко установлена из Коллекции Портов FreeBSD. Пользователи могут выбирать между графическими почтовыми клиентами, такими как evolution или balsa, консольными клиентами, такими как mutt, pine или mail, или Web-интерфейсами, используемыми в некоторых больших организациях.

mail

В FreeBSD в качестве MUA по умолчанию используется mail. Это консольный MUA, предоставляющий все основные функции, необходимые для отправки и получения текстовых сообщений, хотя его возможности по работе с вложениями ограничены и он может работать только с локальными почтовыми ящиками.

Хотя mail не поддерживает работу с серверами POP или IMAP, эти почтовые ящики могут быть загружены в локальный файл mbox с помощью fetchmail.

Для отправки и получения почты просто выполните команду mail, как в этом примере:

```
% mail
```

Содержимое почтового ящика в каталоге /var/mail будет автоматически прочитано утилитой mail. Если почтовый ящик пуст, утилита завершит работу с сообщением о том, что почта не была обнаружена. После чтения почтового ящика запустится интерфейс программы и будет отображен список сообщений. Сообщения нумеруются автоматически и будут выглядеть как в этом примере:

```
Mail version 8.1 6/6/93.  Type ? for help.
"/var/mail/marcs": 3 messages 3 new
>N  1 root@localhost       Mon Mar  8 14:05  14/510   "test"
 N  2 root@localhost    Mon Mar  8 14:05  14/509   "user account"
 N  3 root@localhost       Mon Mar  8 14:05  14/509   "sample"
```

Теперь сообщения могут быть прочитаны с помощью команды t, завершаемой номером сообщения, которое должно быть отображено. В этом примере мы прочтем первое сообщение:

& t 1
Message 1:
From root@localhost Mon Mar 8 14:05:52 2004
X-Original-To: marcs@localhost
Delivered-To: marcs@localhost
To: marcs@localhost
Subject: test
Date: Mon, 8 Mar 2004 14:05:52 +0200 (SAST)
From: root@localhost (Charlie Root)

This is a test message, please reply if you receive it.

Как видно в примере выше, клавиша t выводит сообщение со всеми заголовками. Для повторного вывода списка сообщений необходимо использовать клавишу h.

Если требуется ответить на сообщение, используйте для ответа mail, нажав клавишу R или r. Клавиша R используется в mail для ответа только отправителю, а r для ответа и отправителю, и другим получателям сообщения. Вы можете также завершить эти команды номером письма, на которое хотите составить ответ. После этого необходимо ввести ответ, конец сообщения должен быть завершен символом . на новой строке. Пример можно увидеть ниже:

& R 1
To: root@localhost Subject: Re: test

Thank you, I did get your email.
.
EOT

Для отправки нового сообщения используйте клавишу m и введите адрес получателя. Несколько получателей могут быть указаны через запятую. Введите тему сообщения и его содержимое. Конец сообщения отмечается помещением символа. на новой строке.

& mail root@localhost
Subject: I mastered mail

Now I can send and receive email using mail ... :)
.
EOT

В утилите mail для вызова справки в любой момент может быть использована команда ?, для получения помощи по mail необходимо также обратиться к странице справочника mail.

Замечание: Как упоминалось выше, команда mail не была первоначально предназначена для работы с вложениями, и поэтому их поддержка довольно слабая. Современные MUA, такие как mutt, работают с вложениями гораздо более уверенно. Но если вы все же предпочитаете использовать mail, установите порт converters/mpack.

mutt

mutt это небольшая но очень мощная почтовая программа с отличными возможностями, в числе которых:

- Возможность сортировки сообщений по дискуссиям;
- Поддержка PGP для подписи и шифрования сообщений;
- Поддержка MIME;
- Поддержка Maildir;
- Широкие возможности настройки. Все эти возможности делают mutt одним из самых лучших почтовых клиентов.

Обратитесь к http://www.mutt.org за дополнительной информацией по mutt.

Стабильная версия mutt может быть установлена из порта mail/mutt. После установки порта, mutt может быть запущен следующей командой:

% mutt

mutt автоматически прочтет содержимое пользовательского почтового ящика в каталоге /var/mail и отобразит почту, если она имеется в наличии. Если почты в ящике пользователя нет, mutt будет ожидать команд от пользователя. В примере ниже показан mutt со списком сообщений:

Рис. 1 – mutt со списком сообщений

Для чтения почты просто выберите сообщение с помощью клавиш навигации и нажмите Enter. Пример mutt, отображающего сообщение, показан ниже:

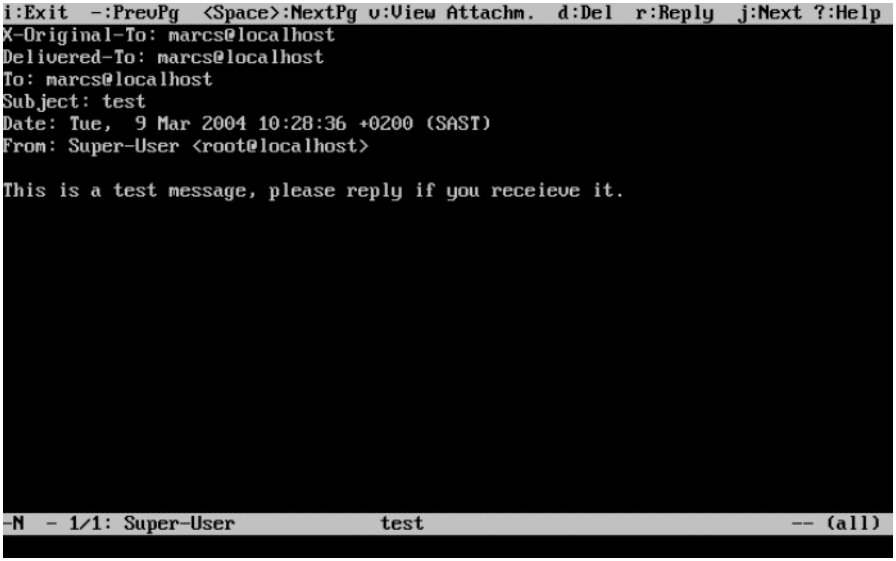

Рис. 2 – mutt, отображающий сообщение

Как и команда mail, mutt позволяет пользователям отвечать как только отправителю, так и всем получателям. Для ответа только отправителю почты, используйте клавишу r. Для группового ответа и отправителю сообщения и всем получателям используйте клавишу g.

Примечание

mutt использует vi в качестве редактора для создания писем и ответа на них. Редактор можно заменить путем создания или редактирования собственного .muttrc в своем домашнем каталоге и установки переменной editor, или установкой переменной окружения EDITOR. Обратитесь к http://www.mutt.org/ за более подробной информацией о настройке mutt.

Для создания нового почтового сообщения нажмите m. После введения темы mutt запустит vi для создания письма. Как только письмо будет завершено, сохраните его и закройте vi, mutt

продолжит работу, отобразив окно с сообщением, которое должно быть отправлено. Для отправки сообщения нажмите y. Пример окна с сообщением показан ниже:

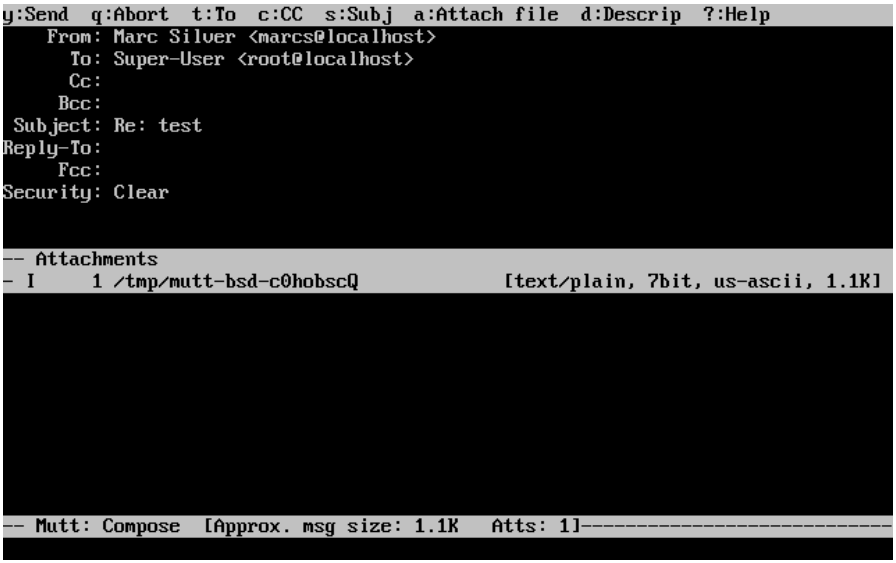

Рис. 3. – mutt с сообщением

mutt также содержит исчерпывающий справочник, к которому можно обратиться из большинства меню, нажав клавишу ?. Верхняя строка также показывает клавиатурные сокращения, которые могут быть использованы.

Использование fetchmail

fetchmail это полноценный IMAP и POP клиент, позволяющий пользователям автоматически загружать почту с удаленных серверов IMAP и POP в локальные почтовые ящики; так доступ к почтовым ящикам упрощается. fetchmail может быть установлен из порта mail/fetchmail и предоставляет различные возможности, в том числе:

- Поддержка протоколов POP3, APOP, KPOP, IMAP, ETRN и ODMR.

- Возможность пересылки почты через SMTP, что позволяет использовать функции фильтрации, перенаправления и синонимов.
- Может быт запущен в режиме даемона для периодической проверки поступающих сообщений.
- Может забирать почту с нескольких почтовых ящиков и рассылать ее различным локальным пользователям в зависимости от настроек.

Утилита fetchmail требует наличия файла настройки .fetchmailrc. Этот файл включает информацию о сервере, а также информацию для аутентификации. Поскольку этот файл содержит важную информацию, правильно будет сделать его доступным для чтения только владельцем с помощью следующей команды:

% chmod 600 .fetchmailrc

В следующем примере файл .fetchmailrc предназначен для загрузки одного почтового ящика по протоколу POP. Этот файл указывает fetchmail соединиться с example.com с именем пользователя joesoap и паролем XXX. В примере подразумевается, что пользователь joesoap существует также и в локальной системе

. poll example.com protocol pop3 username "joesoap" password "XXX"

В следующем примере производится подключение к нескольким POP и IMAP серверам, при необходимости почта перенаправляется другим локальным пользователям:

poll example.com proto pop3:
user "joesoap", with password "XXX", is "jsoap" here;
user "andrea", with password "XXXX"; poll example2.net proto imap:
user "john", with password "XXXXX", is "myth" here;

Утилита fetchmail может работать в режиме даемона с флагом -d, заданным с интервалом (в секундах), через который fetchmail должен опрашивать серверы, перечисленные в .fetchmailrc. В

следующем примере fetchmail будет забирать почту каждые 600 секунд:

% fetchmail -d 600

Дополнительную информацию о fetchmail можно найти на сайте http://fetchmail.berlios.de/.

Использование procmail

Утилита procmail это невероятно мощное приложение, используемое для фильтрации входящей почты. Она позволяет пользователям определять "правила", которые могут быть сопоставлены входящим письмам для выполнения определенных действий или для перенаправления почты в альтернативные почтовые ящики и/или на почтовые адреса. procmail может быть установлен с помощью порта mail/procmail. После установки он может быть непосредственно интегрирован в большинство MTA; сверьтесь с документацией на ваш MTA. Другой способ интеграции procmail - добавление в файл .forward, находящийся в домашнем каталоге пользователя, следующей строки:

"|exec /usr/local/bin/procmail || exit 75"

В этом разделе будут показаны основы настройки правил procmail, а также краткое описание их действия. Эти и другие правила должны быть помещены в файл .procmailrc, который должен находиться в домашнем каталоге пользователя.

Большую часть этих правил также можно найти на странице справочника procmailex.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Настроить sendmail.
2. Настроить файл /etc/mail/access
3. Настроить файл /etc/mail/aliases
4. Настроить файл /etc/mail/local-host-names

5. Настроить файл /etc/mail/sendmail.cf
6. Настроить файл /etc/mail/virtusertable
7. Заместить sendmail как почтовую программу по умолчанию.
8. Установить сервис POP3 и/или IMAP.
9. Настроить сервис POP3 и/или IMAP.
10. Проверить работоспособность.
11. Завершить работу FreeBSD.
12. Ответить на контрольные вопросы и подготовить отчет.

ФОРМА ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ

.

Отчет на защиту предоставляется в печатном или электронном виде.

Структура отчета (на отдельном листе(-ах)): титульный лист, формулировка задания (вариант), этапы выполнения работы (со скриншотами).

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Опишите назначение программы Sendmail.
2. Перечислите протоколы доставки почты и какие из них реализуются Sendmail.
3. Укажите утилиту, которой можно посмотреть почту в ОС FreeBSD.
4. Укажите символы, которые обозначают конец почтового сообщения.
5. Дайте определение и назначение Base64.
6. Перечислите основные части, задействованные в работе почтовой системы.
7. Охарактеризуйте пользовательский почтовый клиент.
8. Дайте определение почтового демона.
9. Опишите способ получения доступа к удаленным почтовым ящикам по протоколам POP и IMAP.

10. Опишите способ получения доступа к локальным почтовым серверам.
11. Дайте определения почтового хоста.
12. Опишите назначение файлов настройки sendmail.
13. Укажите способ отключения sendmail.
14. Укажите способ установки sendmail как программы по умолчанию.
15. Перечислите неисправности, которые могут возникнуть и пути их устранения
16. Опишите способ настройки почты для локального домена.
17. Опишите способ настройки SMTP через UUCP
18. Перечислите пользовательские почтовые программы.
19. Укажите назначение procmail.
20. Для чего нужен файл /etc/mail/virtusertable?
21. Как можно отключить sendmail?

ЛАБОРАТОРНАЯ РАБОТА №4
НАСТРОЙКА ГРАФИЧЕСКОЙ ОБОЛОЧКИ

ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Целью выполнения лабораторной работы является приобретение практических навыков по настройке графического режима и сервера X Window.

Основными задачами выполнения лабораторной работы являются:

1. Научиться настраивать драйверы графической карты.
2. Научиться настраивать X-сервер для запуска графической оболочки.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

X Window System. Обзор

FreeBSD использует X11 для того, чтобы дать пользователям мощный графический интерфейс. X11 является свободно доступной версией *X Window System*, реализованной в *Xorg* и *XFree86* (а также других программных пакетах, здесь не рассматриваемых). В версиях FreeBSD до и включая FreeBSD 5.2.1-RELEASE сервером X11 по умолчанию был XFree8, выпускаемый The XFree86 Project, Inc. Начиная с FreeBSD 5.3-RELEASE, официальной версией X11 по умолчанию стал *Xorg*, разработанный *X.Org Foundation* под лицензией, очень похожей на ту, которая используется FreeBSD. Под FreeBSD существуют также коммерческие X серверы.

Эта лабораторная работа посвящена установке и настройке X11 в системе FreeBSD, с акцентом на релиз *Xorg 7.7*. За информацией о настройке XFree86™ (в более старых релизах FreeBSD XFree8 был реализацией X11 по умолчанию), или более старых релизов *Xorg*, всегда можно обратиться к старым версиям Руководства FreeBSD по адресу *http://docs.FreeBSD.org/doc/*.

Основы X

Первое знакомство с X может оказаться чем-то вроде шока для тех, кто работал с другими графическими системами, такими, как Microsoft Windows или Mac OS. Хотя нет необходимости вникать во все детали различных компонентов X и их взаимодействия, некоторые базовые знания делают возможным использование сильных сторон X.

Почему именно X?

X не является первой оконной системой для UNIX, но она самая популярная из них. До работы над X команда ее разработчиков трудилась над другой оконной системой. Та система называлась «W» (от «Window»). X была просто следующей буквой в романском алфавите. X можно называть «X», «X Window System», «X11» и

множеством других терминов. Факт использования названия «X Windows» для X11 может задеть интересы некоторых людей.

Модель клиент/сервер в X

X изначально разрабатывалась, чтобы быть системой, ориентированной на работу в сети с использованием модели «клиент- сервер».

В модели работы X «X-сервер» работает на компьютере с клавиатурой, монитором и мышью. Область ответственности сервера включает управление дисплеем, обработку ввода с клавиатуры, мыши и других устройств ввода или вывода (например, «планшет» может быть использован в качестве устройства ввода, а видеопроектор в качестве альтернативного устройства вывода). Каждое X-приложение (например, XTerm или Netscape) является «клиентом». Клиент посылает сообщения серверу, такие, как «Пожалуйста, нарисуй окно со следующими координатами», а сервер посылает в ответ сообщения типа «Пользователь только что щёлкнул мышью на кнопке OK».

В случае использования дома или в офисе, сервер и клиенты X как правило будут работать на том же самом компьютере. Однако реально возможно запускать X-сервер на менее мощном настольном компьютере, а приложения X (клиенты) на, скажем, мощной и дорогой машине, обслуживающей целый офис. В этом сценарии X- клиент и сервер

общаются через сеть.

Некоторых это вводит в заблуждение, потому что терминология X в точности обратна тому, что они ожидают. Они полагают, что «X- сервер» будет большой мощной машиной, стоящей на полу, а «X- клиентом» является машина, стоящая на их столах.

Важно помнить, что X-сервером является машина с монитором и клавиатурой, а Хклиенты являются программами, выводящими окна.

В протоколе нет ничего, что заставляет машины клиента и сервера работать под управлением одной и той же операционной системы, или даже быть одним и тем же типом компьютера. Определённо возможно запускать X-сервер в Microsoft Windows или Mac OS от Apple, и есть множество свободно распространяемых и коммерческих приложений, которые это реализуют.

Оконный менеджер

Философия построения X очень похожа на философию построения UNIX, «инструменты, не политика». Это значит, что X не пытаются диктовать то, как должна быть выполнена работа. Вместо этого пользователю предоставляются инструменты, а за пользователем остается принятие решения о том, как использовать эти инструменты.

Этот подход расширен в X тем, что не задается, как окна должны выглядеть на экране, как их двигать мышью, какие комбинации клавиш должны использоваться для переключения между окнами (то есть Alt+Tab, в случае использования Microsoft Windows), как должны выглядеть заголовки окон, должны ли в них быть кнопки для закрытия, и прочее.

Вместо этого X делегирует ответственность за это приложению, которое называется «Window Manager» (Менеджер Окон). Есть десятки оконных менеджеров для X:

AfterStep, Blackbox, ctwm, Enlightenment, fvwm, Sawfish, twm, WindowMaker и другие.

Каждый из этих оконных менеджеров предоставляет различные внешние виды и удобства; некоторые из них поддерживают «виртуальные рабочие столы»; некоторые из них позволяют изменять назначения комбинаций клавиш, используемых для управления рабочим столом; в

некоторых есть кнопка «*Start*» или нечто подобное; некоторые поддерживают «темы», позволяя изменять внешний вид, поменяв тему. Эти оконные менеджеры, а также множество других, находятся в категории *X11-wm коллекции портов*. Кроме того, оболочки KDE и GNOME имеют собственные оконные менеджеры, которые интегрированы в оболочку.

Каждый оконный менеджер также имеет собственный механизм настройки; некоторые предполагают наличие вручную созданного конфигурационного файла; некоторые предоставляют графические инструменты для выполнения большинства работ по настройке; по крайней мере один (*Sawfish*) имеет конфигурационный файл, написанный на диалекте языка *Lisp*.

Примечание

Политика фокусирования

Другой особенностью, за которую отвечает оконный менеджер, является «политика фокусирования» мыши. Каждая оконная система должна иметь некоторый способ выбора окна для активации получения нажатий клавиш, а также визуальную индикацию того, какое окно активно.

Широко известная политика фокусировки называется «щелчок-для-фокуса» («click-tofocus»). Эта модель используется в Microsoft Windows, когда окно становится активным после получения щелчка мыши.

X не поддерживает никакой конкретной политики фокусирования. Вместо этого менеджер окон управляет тем, какое окно владеет фокусом в каждый конкретный момент времени.

Различные оконные менеджеры поддерживают разные методы фокусирования.

Самыми популярными политики фокусирования являются: фокус следует за мышью (focus-follows-mouse)

Фокусом владеет то окно, что находится под указателем мыши. Это не обязательно будет окно, которое находится поверх всех остальных. Фокус меняется при указании на другое окно, при этом также нет нужды щёлкать на нём.

Нечеткий фокус (sloppy-focus)

С политикой focus-follows-mouse если мышь помещается поверх корневого окна (или заднего фона), то никакое окно фокус не получает, а нажатия клавиш просто пропадают. При использовании политики

нечёткого фокуса он меняется только когда курсор попадает на новое окно, но не когда уходит с текущего окна. щелчок для выбора фокуса (click-to-focus)

Активное окно выбирается щелчком мыши. Затем окно может быть «поднято» и появится поверх всех других окон. Все нажатия клавиш теперь будут направляться в это окно, даже если курсор переместится к другому.

Виджеты

Подход X, заключающийся в предоставлении инструментов, а не политики, распространяется и на виджеты, которые располагаются на экране в каждом приложении.

«Виджет» (widget) является термином для всего в пользовательском интерфейсе, на чём можно щёлкать или каким-то образом управлять; кнопки, зависимые (radio buttons) и независимые (check boxes) опции, иконки, списки и так далее. В Microsoft Windows это называется «элементами управления» («controls»).

Microsoft Windows и Mac OS от Apple имеют очень жёсткую политику относительно виджетов. Предполагается, что разрабатываемые приложения обязательно должны иметь похожий внешний вид. Что касается X, то было решено, что не нужно требовать обязательного использования какого-то определённого графического стиля или набора виджетов.

В результате не стоит ожидать от X-приложений похожести во внешнем виде. Существует несколько популярных наборов виджетов и их разновидностей, включая оригинальный набор виджетов Athena от MIT, Motif, OpenLook и другие.

В большинстве появляющихся в настоящее время приложений для X будет использоваться современно выглядящий набор виджетов, либо *Qt*, используемый в KDE, либо GTK+, используемый проектом GNOME. В этом отношении наблюдается унификация внешнего вида рабочего стола в UNIX, что определённо облегчает жизнь начинающему пользователю.

УСТАНОВКА X11

Версией X11 по умолчанию для FreeBSD является Xorg. Xorg это

сервер X дистрибутива открытой реализации X Window System, выпущенной *X.Org Foundation*. *Xorg* основан на коде *XFree8 4.4RC2* и *X11R6.6*. Версия *Xorg*, доступная на данный момент из *коллекции портов* FreeBSD: 7.7.

Для сборки и установки Xorg из Коллекции портов, выполните:

```
#    cd /usr/ports/x11/xorg
#    make install clean
```

Примечание

Перед сборкой полной версии Xorg удостоверьтесь в наличии хотя бы 4 GB свободного места.

Кроме того, X11 может быть установлен непосредственно из пакетов. Бинарные пакеты, устанавливаемые pkg_add, доступны и для X11. Когда pkg_add используется для удаленной загрузки пакетов, номер версии пакета необходимо удалить. pkg_add автоматически установит последнюю версию приложения.

Таким образом, для загрузки и установки пакета Xorg, просто наберите:

```
#    pkg_add -r xorg
```

Примечание

В примерах выше будет установлен полный дистрибутив X11, включая серверы, клиенты, шрифты и так далее. Также доступны и отдельные пакеты, и порты для различных частей X11.

КОНФИГУРАЦИЯ X11

Перед настройкой X11 необходима следующая информация о конфигурируемой системе:

- Характеристики монитора
- Набор микросхем, используемый в видеоадаптере
- Объём видеопамяти

Характеристики монитора используются в X11 для определения

рабочего разрешения и частоты. Эти характеристики обычно могут быть получены из документации, которая прилагается к монитору или с сайта производителя. Тут нужны два диапазона значений, для частоты горизонтальной развёртки и для частоты вертикальной синхронизации.

Набор микросхем графического адаптера определяет, модуль какого драйвера использует X11 для работы с графическим оборудованием. Объём видеопамяти графического адаптера определяет разрешение и глубину цвета, с которым может работать система. Это важно, чтобы пользователь знал ограничения системы.

Конфигурирование X11

Начиная с версии 7.3, Xorg зачастую может работать без какого- либо файла настройки, для его запуска достаточно просто набрать:

- `startx`

Начиная с версии 7.4, Xorg может использовать HAL для автоматического поиска клавиатуры и мыши. Порты *sysutils/hal* и *devel/dbus* будут инсталлированы как зависимости *x11/xorg*, но для их включения необходимо иметь следующие записи в */etc/rc.conf file:*

```
haldenable="YES" dbusenable="YES"
```

Эти сервисы должны быть запущены до последующей загрузки Xorg конфигурации.

Автоматическая конфигурация не всегда может сработать на некотором оборудовании, либо создать не совсем ту настройку, которая желаема. В этих случаях, необходима ручная настройка конфигурации.

Примечание

Такие оконные менеджеры, как GNOME, KDE или Xfce имеют собственные утилиты, позволяющие пользователю легко устанавливать такие параметры, как разрешение экрана. Поэтому, если конфигурация по умолчанию не подходящая и вы планируете инсталлировать эти оконные менеджеры, можете продолжить настройку рабочей среды, используя их собственные утилиты для установок параметров экрана.

Процесс настройки X11 является многошаговым. Первый шаг заключается в построении начального конфигурационного файла. Работая с правами супер-пользователя, просто запустите:

```
#    Xorg -configure
```

При этом в каталоге */root* будет создан скелет конфигурационного файла X11 под именем *xorg.conf.new*. Программа X11 сделает попытку распознать графическое оборудование системы и запишет конфигурационный файл, загружающий правильные драйверы для обнаруженного оборудования в системе.

Следующим шагом является тестирование существующей конфигурации для проверки того, что *Xorg* может работать с графическим оборудованием в настраиваемой системе. Для этого выполните:

```
#    Xorg -config xorg.conf.new
```

Начиная с Xorg 7.4 и выше, это тестирование покажет лишь черный экран, что делает диагностику не совсем полноценным. Старое поведение будет доступно при использовании опции *retro*

```
#    Xorg -config xorg.conf.new -retro
```

Если появилась чёрно-белая сетка и курсор мыши в виде X, то настройка была выполнена успешно. Для завершения тестирования просто нажмите одновременно *Ctrl+Alt+Backspace*

Примечание

Данная комбинация включена по умолчанию до Xorg версии 7.3. Для включения этого в версии 7.4 и выше, вы должны ввести следующую команду в любом эмуляторе X терминала:

```
% setxkbmap -option terminate: ctrl_alt_bksp
```

или создать конфигурационный файл клавиатуры для hald называемый x11-input.fdi и сохранить его в /usr/local/etc/hal/fdi/policy

директории. Данный файл должен содержать следующие строчки:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<deviceinfo version="0.2">
<device>
<match key="info.capabilities" contains="input. keyboard">
<merge key="input.x11_options.XkbOptions"
type="string">terminate: ctrl_alt_bksp</merge>
</match>
</device>
</deviceinfo>
```

Вам может потребоваться перезагрузка системы для вступления параметров hald в силу.

Если мышь не работает, ее необходимо настроить. Дополнительно, начиная с версии 7.4, секция InputDevice в xorg.conf игнорируется в

пользу автоматического поиска устройств. Для возвращения старого поведения, добавьте следующие строчки в секции ServerLayout или ServerFlags:

```
Option "AutoAddDevices" "false"
```

Устройства ввода могут конфигурированы затем как в предыдущих версиях, вместе с другими необходимыми опциями (такими, как переключение раскладок клавиатуры).

Примечание

Как ранее уже сообщалось, начиная с версии 7.4, по умолчанию, hald демон будет пытаться распознать вашу клавиатуру автоматически. Есть возможность, что раскладка вашей клавиатуры или ее модель будут определены некорректно. Такие оконные менеджеры как GNOME, KDE или Xfce содержат свои инструменты для конфигурирования клавиатур. Тем не менее, можно установить параметры клавиатуры непосредственно с помощью утилиты setxkbmap или через hald конфигурационные правила.

Например, если вы хотите использовать клавиши PC 102 клавиатуры,

идущая с французской раскладкой, мы должны создать конфигурационный файл клавиатуры для hald называемый x11- input.fdi и сохранить в /usr/local/etc/hal/fdi/policy директории. Этот файл должен содержать следующие строчки:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<deviceinfo version="0.2">
<device>
<match key="info.capabilities" contains="input.keyboard">
<merge key="input.x11_options.XkbModel"
type="string">pc102</merge>
<merge key="input.x11_options.XkbLayout" type="string">fr</merge>
</match>
</device>

</deviceinfo>
```

Если этот файл уже существует, просто скопируйте и добавьте эти строчки в файл данный файл. Вы должны будете перезагрузить систему, чтобы заставить hald применить настройки. Есть возможность проделать ту же конфигурацию из X терминала или скрипт следующей командой:

```
% setxkbmap -model pc102 -layout fr
```

Файл /usr/local/share/X11/xkb/rules/base.lst содержит список различных клавиатур, доступные опции и раскладки

Теперь выполните тонкую настройку в файле *xorg.conf.new* по своему вкусу. Сначала задайте частоты для монитора. Они обычно обозначаются как частоты горизонтальной и вертикальной синхронизации. Эти значения добавляются в файл *xorg.conf.new* в раздел "*Monitor*":

```
Section "Monitor"
Identifier "Monitor0"
VendorName "Monitor Vendor"
ModelName "Monitor Model"
```

HorizSync 30-107
VertRefresh 48-120
EndSection

X позволяет использовать возможности технологии DPMS с поддерживающими её мониторами. Программа xset управляет временными задержками и может явно задавать режимы ожидания, останова и выключения. Если вы хотите включить использование возможностей DPMS вашего монитора, вы должны добавить следующую строку в раздел, описывающий монитор:

```
Option "DPMS"
```

Пока файл конфигурации *xorg.conf.new* открыт в редакторе, выберите желаемые разрешение и глубину цвета, которые будут использоваться по умолчанию. Они задаются в разделе "*Screen*":

Section "Screen"
Identifier "Screen0"
Device "Card0"
Monitor "Monitor0"
DefaultDepth 24
SubSection "Display" Viewport 0 0
Depth 24
Modes "1024x768"
EndSubSection
EndSection

Ключевое слово DefaultDepth описывает глубину цвета, с которой будет работа по умолчанию. Это значение может быть переопределено при помощи параметра командной строки *-depth* для *Xorg*. Ключевое слово *Modes* описывает разрешение, с которым нужно работать при данной глубине цвета.

Наконец, запишите конфигурационный файл и протестируйте его при помощи тестового режима, описанного выше.

Примечание

При решении проблем могут помочь лог файлы X11, в которых находится информация по каждому устройству, к которому подключен сервер X11. Лог файлам Xorg названия даются в формате /var/log/Xorg.0.log . Имена лог файлам могут даваться от Xorg.0.log до Xorg.8.log и так далее.

Если все в порядке, то конфигурационный файл нужно установить в общедоступное место, где его сможет найти *Xorg*. Обычно это */etc/X11/xorg.conf* или */usr/local/etc/X11/xorg.conf*.

```
# cp xorg.conf.new /etc/X11/xorg.conf
```

Теперь процесс настройки X11 завершен. *Xorg* теперь можно запустить с помощью *startx*. X11 можно также запустить через *xdm*.

Тонкие вопросы настройки

Конфигурирование при работе с графическими чипсетами Intel i810

Конфигурирование при работе с интегрированными наборами микросхем *Intel i810* требует наличия *agpgart* , программного интерфейса AGP, посредством которого X11 будет управлять адаптером.

Это позволит конфигурировать графическое оборудование точно так же, как и любой другой графический адаптер. Заметьте, что для систем, у которых драйвер *agp* в ядро не вкомпилирован, попытка погрузить модуль с помощью *kldload* окончится неудачно. Этот драйвер должен оказаться в ядре во время загрузки, либо вкомпилированным,либо подгруженным посредством */boot/loader.conf* .

Настройка широкоэкранного режима

Для этого раздела необходимо несколько больше навыков настройки. Если после использования описанных выше инструментов настройки в результате рабочей конфигурации не получается, в лог файлах достаточно информации для доведения конфигурации до рабочего уровня. Для настройки используется текстовый редактор.

Существующие широкоэкранные стандарты (*WSXGA, WSXGA+*,

WUXGA, WXGA, WXGA+, и т.д.) поддерживают форматы изображения 16:10 и 10:9, которые могут быть проблемными. Для формата 16:10, например, возможны следующие разрешения экрана:

- 2560x1600
- 1920x1200
- 1680x1050
- 1440x900
- 1280x800

Иногда достаточно добавить одно из этих разрешений в качестве параметра *Mode* в раздел Section "*Screen*" вот так:

Section "Screen"
Identifier "Screen0"
Device "Card0"
Monitor "Monitor0"
DefaultDepth 24
SubSection "Display"
Viewport 0 0
Depth 24
Modes "1680x1050"
EndSubSection
EndSection

Xorg может извлечь информацию о разрешении из монитора посредством *I2C/DDC*, так что у него есть данные, какие частоты и разрешения может поддерживать монитор.

Если эти *ModeLines* не определены в драйверах, может потребоваться дополнительная настройка *Xorg*. Используя */var/log/Xorg.0.log* , можно извлечь достаточно информации для создания рабочей строки *ModeLine* вручную. Просто обратитесь к следующей информации:

(11) MGA(0): Supported additional Video Mode:

(12) MGA(0): clock: 146.2 MHz Image Size: 433 x 271 mm (II) MGA(0): h_active: 1680 h_sync: 1784 h_sync_end 1960 h_blank_end 2240 h_border: 0
(13) MGA(0): v_active: 1050 v_sync: 1053 v_sync_end 1059 (J v_blanking: 1089 v_border: 0

(14) MGA(0): Ranges: V min: 48 V max: 85 Hz, H min: 30 H max: 94 (J kHz, PixClock max 170 MHz

Эта информация называется *EDID*. Создание *ModeLine* из сводится к расположению номеров в правильном порядке:

```
ModeLine <name> <clock> <4 horiz. timings> <4 vert. timings>
```

Для нашего примера *ModeLine* в *Section "Monitor"* будет выглядеть так:

Section "Monitor"
Identifier "Monitor1"
VendorName "Bigname"
ModelName "BestModel"
ModeLine "1680x1050" 146.2 1680 1784 1960 2240 1050 1053 (J 1059 1089
Option "DPMS"
EndSection

После завершения редактирования конфигурации, X должен запуститься в новом широкоэкранном разрешении.

ИСПОЛЬЗОВАНИЕ ШРИФТОВ В X11

Шрифты Type1

Шрифты, используемые по умолчанию и распространяемые вместе с X11, вряд ли можно назвать идеально подходящими для применения в обычных издательских приложениях. Большие презентационные шрифты выглядят рвано и непрофессионально, а мелкие шрифты в Netscape вообще невозможно разобрать. Однако есть некоторое количество свободно распространяемых высококачественных шрифтов *Type1* (*PostScript*), которые можно без изменений использовать с X11. К примеру, в наборе шрифтов URW (*x11fonts/urwfonts*) имеются высококачественные версии стандартных шрифтов *Type1* (*Times Roman, Helvetica, Palatino* и другие). В набор Freefonts (*x11-fonts/freefonts*) включено ещё больше шрифтов, однако большинство из них предназначено для использования в программном обеспечении для работы с графикой, например, *Gimp,* и они не вполне пригодны для использования в качестве экранных шрифтов. Кроме того, X11 с минимальными усилиями может быть настроена на использование шрифтов *TrueType*.

Для установки вышеупомянутых коллекций шрифтов *Type1* из *коллекции портов* выполните следующие команды:

```
# cd /usr/ports/x11-fonts/urwfonts # make install clean
```

То же самое нужно будет сделать для коллекции freefont и других. Чтобы X-сервер обнаруживал этих шрифты, добавьте соответствующую строку в файл настройки X сервера (/etc/X11/xorg.conf), которая должна выглядеть так:

FontPath "/usr/local/lib/X11/fonts/URW/"

Либо из командной строки при работе с X выполните:

```
#    xset fp+ /usr/local/lib/X11/fonts/URW
#    xset fp rehash
```

Это сработает, но будет потеряно, когда сеанс работы с X будет

закрыт, если эта команда не будет добавлена в начальный файл *~/.xinitrc* в случае обычного сеанса через *startx* или *~/.xsession* при входе через графический менеджер типа *XDM*). Третий способ заключается в использовании нового файла */usr/local/etc/fonts/local.conf*.

Шрифты TrueType

В *Xorg* имеется встроенная поддержка шрифтов *TrueType*. Имеются два модуля, которые могут обеспечить эту функциональность. В нашем примере используется модуль *freetype*, потому что он в большей степени похож на другие механизмы для работы с шрифтами.

Для включения модуля *freetype* достаточно в раздел "*Module*" файла */etc/X11/xorg.conf* добавить следующую строчку.

Load "freetype"

Теперь создайте каталог для шрифтов TrueType (к примеру, */usr/local/lib/X11/fonts/TrueType*) и скопируйте все шрифты *TrueType* в этот каталог. Имейте в виду, что напрямую использовать шрифты *TrueType* с *Macintosh* нельзя; для использования с X11 они должны быть в формате *UNIX/MS-DOS/Windows*. После того, как файлы будут скопированы в этот каталог, воспользуйтесь утилитой *ttmkfdir* для создания файла *fonts.dir* , который укажет подсистеме вывода шрифтов X на местоположение этих новых файлов. *ttmkfdir* имеется в *Коллекции Портов* FreeBSD: *x11-fonts/ttmkfdir*.

```
#    cd /usr/local/lib/X11/fonts/TrueType
#    ttmkfdir -o fonts.dir
```

После этого добавьте каталог со шрифтами *TrueType* к маршруту поиска шрифтов. Это делается точно также, как описано выше для шрифтов *Type1*, то есть выполните

```
% xset fp+ /usr/local/lib/X11/fonts/TrueType
% xset fp rehash
```

или добавьте строку *FontPath* в файл *xorg.conf* .

Это всё. Теперь *Netscape*, *Gimp*, *StarOffic* и все остальные X- приложения должны увидеть установленные шрифты *TrueType*. Очень маленькие (как текст веб-страницы на дисплее с высоким разрешением) и очень большие (в *StarOffice*) шрифты будут теперь выглядеть гораздо лучше.

Антиалиасинг шрифтов

Антиалиасинг присутствует в X11 начиная с *XFree86*, версии 4.0.2. Однако настройка шрифтов была довольно громоздка вплоть до появления XFree8 4.3.0.

Начиная с версии XFree86 4.3.0, все шрифты, расположенные в каталогах */usr/local/lib/X11/fonts/ и ~/.fonts/*, автоматически становятся доступными для применения антиалиасинга в приложениях, использующих *Xft*. Не все приложения могут использовать Xft, но во многих его поддержка присутствует. Примерами приложений, использующих Xft, является Qt версий 2.3 и более поздних (это инструментальный пакет для оболочки KDE), GTK+ версий 2.0 и более поздних (это инструментальный пакет для оболочки *GNOME*), а также *Mozilla* версий 1.2 и более поздних.

Для применения к шрифтам антиалиасинга, а также для настройки параметров антиалиасинга, создайте (или отредактируйте, если он уже существует) файл */usr/local/etc/fonts/local.conf* . Некоторые мощные возможности системы шрифтов *Xft* могут быть настроены при помощи этого файла; в этом разделе описаны лишь некоторые простые возможности.

Этот файл должен быть сформирован в формате XML. Обратите особое внимание на регистр символов, и удостоверьтесь, что все тэги корректно закрыты. Файл начинается обычным заголовком *XML*, за которым следуют *DOCTYPE* и тэг *<fontconfig>*:

```
<?xml version="1.0"?>
<!DOCTYPE fontconfig SYSTEM "fonts.dtd">
<fontconfig>
```

Как и говорилось ранее, все шрифты из каталога /

usr/local/lib/X11/fonts / , а также *~/.fonts* / уже доступны для приложений, использующих Xft. Если вы хотите добавить каталог, отличный от этих двух, добавьте строчку, подобную следующей, в файл

/usr/local/etc/fonts/local.conf :

<dir>/path/to/my/fonts</dir>

После добавления новых шрифтов, и особенно новых каталогов со шрифтами, вы должны выполнить следующую команду для перестроения кэшей шрифтов:

 # fc-cache -f

Антиалиасинг делает границы несколько размытыми, что делает очень мелкий текст более читабельным и удаляет «лесенки» из текста большого размера, но может вызвать нечёткость при применении к тексту обычного размера. Для исключения размеров шрифтов, меньших 14, из антиалиасинга, добавьте такие строки:

```
<match target="font">
<testname="size"compare="less"> <double>14</double>
</test>

<edit name="antialias" mode="assign"> <bool>false</bool>
</edit>
</match>
<match target="font">
<test name="pixelsize" compare="less" qual="any">
<double>14</double>
</test>
<edit mode="assign" name="antialias"> <bool>false</bool>
</edit>
</match>
```

Для некоторых моноширинных шрифтов антиалиасинг может также

оказаться неприменимым при определении межсимвольного интервала. В частности, эта проблема возникает с KDE. Одним из возможных решений для этого является жесткое задание межсимвольного интервала в 100. Добавьте следующие строки:

```
<match target="pattern" name="family">
<test qual="any" name="family">
<string>fixed</string>
</test>
<edit name="family" mode="assign"> <string>mono</string>
</edit>
</match>
<match target="pattern" name="family">
<test qual="any" name="family"> <string>console</string>
</test>
<edit name="family" mode="assign"> <string>mono</string>
</edit>
</match>
```

это создаст алиасы "mono" для других общеупотребительных имён шрифтов фиксированного размера, а затем добавьте:

```
<match target="pattern" name="family"> <test qual="any" name="family"> <string>mono</string>
</test>
<edit name="spacing" mode="assign"> <int>100</int>
</edit>
</match>
```

С некоторыми шрифтами, такими, как *Helvetica*, при антиалиасинге могут возникнуть проблемы. Обычно это проявляется в виде шрифта, который наполовину вертикально обрезан. Хуже того, это может привести к сбоям таких приложений, как *Mozilla*. Во избежание этого следует добавить следующее в файл *local.conf* :

```
<match target="pattern" name="family">
<test qual="any" name="family">
```

<string>Helvetica</string>
</test>
<edit name="family" mode="assign"> <string>sans-serif</string>
</edit>
</match>

После того, как вы закончите редактирование *local.conf,* удостоверьтесь, что файл завершен тэгом *</fontconfig>*.

Набор шрифтов по умолчанию, поставляемый с X11, не очень подходит, если включается антиалиасинг. Гораздо лучший набор шрифтов, используемых по умолчанию, можно найти в порте *x11- fonts/bitstream-vera*. Этот порт установит файл */usr/local/etc/fonts/local.conf,* если такого ещё не существует. Если файл существует, то порт создаст файл */usr/local/etc/fonts/local.conf- vera*. Перенесите содержимое этого файла в */usr/local/etc/fonts/local.conf,* и шрифты Bitstream автоматически заменят используемые по умолчанию в X11 шрифты *Serif, Sans Serif* и *Monospaced*.

Наконец, пользователи могут добавлять собственные наборы посредством персональных файлов *.fonts.conf* . Для этого каждый пользователь должен просто создать файл *~/.fonts.conf* . Этот файл также должен быть в формате XML.

И последнее замечание: при использовании дисплея *LCD* может понадобиться включение разбиения точек. При этом компоненты красного, зелёного и голубого цветов, рассматриваются как отдельные точки для улучшения разрешения экрана по горизонтали; результат может оказаться потрясающим. Для включения этого механизма добавьте такую строчку где-нибудь в файле *local.conf*:

<match target="font">
<test qual="all" name="rgba"> <const>unknown</const>
</test>
<edit name="rgba" mode="assign"> <const>rgb</const>
</edit>

</match>

Примечание

В зависимости от типа дисплея, rgb может потребоваться заменить на bgr, vrgb или vbgr: пробуйте и смотрите, что работает лучше.

Антиалиасинг должен быть включен при следующем запуске X- сервера. Однако программы должны знать, как использовать его преимущества. В настоящее время инструментальный пакет Qt умеет ими пользоваться, так что вся оболочка KDE может использовать шрифты с антиалиасингом. GTK+ и GNOME также можно заставить использовать антиалиасинг посредством капплета «Font». По умолчанию Mozilla версий 1.2 и выше будет автоматически использовать антиалиасинг. Для отмены использования антиалиасинга перестройте *Mozilla* с флагом *-DWITHOUT_XFT*.

МЕНЕДЖЕРЫ ЭКРАНОВ (DISPLAY MANAGERS) X

Вступление

Менеджер Экранов X (XDM) это необязательный компонент X Window System, который используется для управления входом пользователей в систему. Это полезно в ряде ситуаций, например для минимальных «X Терминалов», десктопов, больших сетевых серверов экранов. Так как X Window System не зависит от сетей и протоколов, то существует множество различных конфигураций для X клиентов и серверов, запущенных на различных компьютерах, подключенных к сети. XDM предоставляет графический интерфейс для выбора сервера, к которому вы желаете подключится, и введения информации, авторизующей пользователя, например комбинации логина и пароля.

XDM можно рассматривать как аналог программы *getty*, предоставляющий такие же возможности для пользователей. И это именно так, XDM производит вход в систему для подключенного пользователя и запускает управляющую сессию для пользователя (обычно это менеджер окон X). После этого XDM ожидает завершения приложения, означающее завершение пользователем работы и отключает управляющую сессию. Затем XDM может снова вывести приглашение к входу в систему и ожидать

входа другого пользователя.

Использование XDM

Программой даемона XDM является */usr/local/bin/xdm*. Эта программа может быть запущена от пользователя root в любой момент, и она начнёт управлять дисплеем X на локальной машине. Если XDM нужно запускать в фоновом режиме каждый раз при запуске компьютера, то наиболее правильный способ — это добавить новую запись в */etc/ttys*.

Вот строка, которую необходимо добавить в файл */etc/ttys* для того, чтобы запустить даемон XDM на виртуальном терминале:

```
ttyv8 "/usr/local/bin/xdm -nodaemon" xterm off secure
```

По умолчанию эта запись отключена; для её включения нужно заменить пятое поле с *off* на *on* и перезапустить *init*. Первое поле это название терминала, которым будет управлять программа, ttyv8. Это означает, что XDM будет запущен на 9ом виртуальном терминале.

Конфигурирование XDM

Конфигурационные файлы XDM находятся в каталоге */usr/local/lib/X11/xdm*. В нём размещаются насколько файлов, которые используются для изменения поведения и внешнего вида XDM. Обычно это следующие файлы:

Таблица 1. Файлы для изменения поведения и внешнего вида XDM

Файл	Описание
Xaccess	Правила авторизации клиентов.
Xresources	Значения ресурсов X по умолчанию.
Xservers	Список локальных и удаленных экранов
Xsession	Сценарий сессии по умолчанию.
Xsetup_ *	Скрипт для запуска приложений до появления приглашения к входу в систему
xdm-config	Глобальный конфигурационный файл для всех экранов, запущенных на локальной машине

xdm-errors	Ошибки, сгенерированные серверной программой.
xdm-pid	ID процесса, запущенного XDM

В этом каталоге также находятся несколько командных сценариев и программ, используемых для настройки рабочего стола (*desktop*) при запуске XDM. Назначение каждого из этих файлов будет вкратце

описано. Точный синтаксис и информация по их использованию находятся в *xdm*.

В конфигурации по умолчанию выводится простое прямоугольное окно приглашения ко входу в систему с именем компьютера, написанным сверху большим шрифтом, и строками ввода «*Login*:» и «*Password*:» внизу. Это хорошая отправная точка для изменения внешнего вида экранов *XDM*.

Xaccess

Протокол, по которому происходит подключение дисплеев, управляемых XDM, называется *X Display Manager Connection Protocol* (*XDMCP*). Этот файл представляет собой набор правил для управления *XDMCP* соединениями с удалёнными машинами. Он игнорируется, пока стандартный файл *xdm-config* не содержит указаний по обслуживанию удалённых соединений.

Xresources

Это файл содержит установки по умолчанию для приложений, запущенных в экране выбора серверов и экране приглашения к входу в систему. В нем может быть изменён вид программы входа в систему. Формат этого файла идентичен файлу *appdefaults*, описанному в документации к X11.

Xservers

Это список удаленных экранов, которые XDM должен предоставить как варианты для входа в систему

Xsession

Этот файл представляет из себя командный сценарий по умолчанию для пользователей, вошедших в систему с использованием XDM. Обычно каждый пользователь имеет собственный сценарий входа в файле *~/.xsession* , который используется вместо этого сценария.

Xsetup_*

Они запускаются автоматически перед тем, как показывается экран выбора сервера или экран входа в систему. Для каждого экрана (*display*) есть свой сценарий с именем *Xsetup_*, за которым следует локальный номер экрана (например, *Xsetup_0*). Обычно эти сценарии запускают одну или две программы в фоновом режиме, например *xconsole* .

xdm-config

Здесь содержатся настройки в формате *app-defaults,* которые применимы ко всем экранам данного компьютера.

xdm-errors

Здесь находится выдача X серверов, которые XDM пытается запустить. Если экран, который XDM пытается открыть, отключается по некоторым причинам, то это хорошее место для поиска сообщений об ошибках. Эти сообщения также записываются в пользовательский файл *~/.xsession-errors* для каждого сеанса.

Использование сетевого сервера дисплеев

Для того, чтобы позволить другим клиентам подключаться к серверу дисплеев, необходимо отредактировать правила контроля доступа и включить обслуживание сетевых соединений. По умолчанию они выключены, что является хорошим решением с точки зрения обеспечения безопасности. Для того, чтобы позволить XDM принимать сетевые соединения, в первую очередь закомментируйте строку в файле *xdm-config*:

БЕЗОПАСНОСТЬ: do not listen for XDMCP or Chooser requests

! Закомментируйте эти линии, если вы хотите управлять X терминалами с xdm

DisplayManager.requestPort: 0

затем перезапустите XDM. Помните, что комментарии в файлах *app-defaults* начинаются с символа «*!*», а не как обычно, «*#*». Может потребоваться более жёсткий контроль доступа — взгляните на примеры из *Xaccess*.

Замены для XDM

Существует несколько программ, заменяющих XDM. Одна из них, *kdm* (поставляемая вместе с *KDE*), описана далее в этой главе. В *kdm* имеется много визуальных и косметических улучшений, а также функциональность, позволяющая пользователям выбирать собственные оконные менеджеры во время входа в систему.

ГРАФИЧЕСКИЕ ОБОЛОЧКИ

В этом разделе описываются различные графические оболочки, доступные в X для FreeBSD. Термин «графическая оболочка» может использоваться для чего угодно, от простого менеджера окон до полнофункционального набора приложений для рабочего стола, типа *KDE* или *GNOME*.

GNOME

GNOME является дружественной к пользователю графической оболочкой, позволяющей пользователям легко использовать и настраивать свои компьютеры. В *GNOME* имеется панель (для запуска приложений и отображения их состояния), рабочий стол (где могут быть размещены данные и приложения), набор стандартных инструментов и приложений для рабочего стола, а также набор соглашений, облегчающих совместную работу и согласованность приложений. Пользователи других операционных систем или оболочек при использовании такой мощной графической оболочки, какую обеспечивает *GNOME*, должны чувствовать себя в родной среде.

Установка GNOME

Программу проще всего установить из пакета или *коллекции портов*:

Для установки пакета GNOME из сети, просто наберите:

```
#    pkg_add -r gnome2
```

Для построения GNOME из исходных текстов используйте дерево портов:

```
#    cd /usr/ports/x11/gnome2
#    make install clean
```

После установки GNOME нужно указать X-серверу на запуск *GNOME* вместо стандартного оконного менеджера.

Самый простой путь запустить GNOME - это использовать GDM

(*GNOME Display Manager*). GDM, который устанавливается, как часть GNOME (но отключен по умолчанию), может быть включён путём добавления *gdm_enable="YES" в /etc/rc.conf*.

После перезагрузки, GNOME запустится автоматически после того, как вы зарегистрируйтесь в системе. Никакой дополнительной конфигурации не требуется.

GNOME может также быть запущен из командной строки с помощью конфигурирования файла *.xinitrc* . Если файл *.xinitrc* уже откорректирован, то просто замените строку, в которой запускается используемый менеджер окон, на ту, что вызовет */usr/local/bin/gnome- session*. Если в конфигурационном файле нет ничего особенного, то будет достаточно просто набрать:

```
% echo "/usr/local/bin/gnome-session" > ~/.xinitrc
```

Теперь наберите *startx* , и будет запущена графическая оболочка GNOME.

Примечание

Если используется более старый менеджер дисплеев типа XDM, то это не сработает. Вместо этого создайте выполнимый файл .xsession с той же самой командой в нём. Для этого отредактируйте файл, заменив существующую команду запуска оконного менеджера на /usr/local/bin/gnome-session:

```
% echo "#!/bin/sh" > ~/.xsession
% echo "/usr/local/bin/gnome-session" >> ~/.xsession
% chmod +x ~/.xsession
```

Ещё одним вариантом является настройка менеджера дисплеев таким образом, чтобы он позволял выбирать оконный менеджер во время входа в систему.

Шрифты с антиалиасингом и GNOME

X11 поддерживает антиалиасинг посредством своего расширения «RENDER». GTK+ 2.0 и более поздние версии (это инструментальный

пакет, используемый GNOME) могут использовать такую функциональность. Таким образом, при наличии современного GNOME, возможно использование антиалиасинга. Просто перейдите в *Applications → Desktop Preferences → Font* и выберите либо Best shapes, Best contrast, либо Subpixel smoothing (LCDs). Для приложений GTK+, которые не являются частью оболочки GNOME, задайте в качестве значения переменной окружения GDK_USE_XFT 1 перед запуском программы.

KDE

О KDE

KDE является простой в использовании современной графической оболочкой. Вот лишь некоторые из преимуществ, которые даёт пользователю KDE:

- Прекрасный современный рабочий стол
- Рабочий стол, полностью прозрачный для работы в сети
- Интегрированная система помощи, обеспечивающая удобный и согласованный доступ к системе помощи по использованию рабочего стола KDE и его приложений
- Единообразный внешний вид и управление во всех приложениях KDE
- Стандартизированные меню и панели инструментов, комбинации клавиш, цветовые схемы и так далее.
- Интернационализация: в KDE поддерживается более 40 языков
- Централизованное единообразное конфигурирование рабочего стола в диалоговом режиме
- Большое количество полезных приложений для KDE

Совместно с KDE поставляется веб-браузер под названием Konqueror, который является серьезным соперником другим браузерам для UNIX-систем.

Имеется две версии KDE доступные на FreeBSD. Версия 3 была доступна очень долгое время и она является очень зрелой. Версия 4 - это следующее поколение, также доступное через Коллекцию Портов. Обе версии могут быть инсталлированы одновременно.

Установка KDE

Как и в случае с GNOME или любой другой графической оболочкой, программное обеспечение можно легко установить из пакета или из Коллекции Портов:

Для установки пакета KDE3 из сети, просто наберите:

```
#    pkg_add -r kde
```

Для установки пакета KDE4 из сети, просто наберите:

```
#    pkg_add -r kde4
```

pkg_add автоматически загрузит самую последнюю версию приложения. Для построения KDE3 из исходных текстов, воспользуйтесь деревом портов:

```
#    cd /usr/ports/x11/kde3
#    make install clean
```

Для построения KDE4 из исходных текстов, воспользуйтесь деревом портов:

```
#    cd /usr/ports/x11/kde4
   # make install clean
```

После установки KDE нужно указать X-серверу на запуск этого приложения вместо оконного менеджера, используемого по умолчанию. Это достигается редактированием файла *.xinitrc*:

Для *KDE3*:

```
% echo "exec startkde" > ~/.xinitrc
```

Для *KDE4*:

```
% echo "exec /usr/local/kde4/bin/startkde" > ~/.xinitrc
```

Теперь при вызове X Window System по команде *startx* в качестве оболочки будет использоваться KDE.

При использовании менеджера дисплеев типа XDM настройка несколько отличается. Вместо этого нужно отредактировать файл *.xsession* . Указания для *kdm* описаны далее в этой главе.

Более подробно о KDE

Теперь, когда KDE установлена в системе, можно узнать много нового из её справочных страниц или просто указанием и щелканьем по различным меню. Пользователи Windows или Mac будут чувствовать себя как дома.

Лучшим справочником по KDE является онлайновая документация. KDE поставляется с собственным веб-браузером, который называется *Konqueror*, десятками полезных приложений и подробной документацией. В оставшейся части этого раздела обсуждаются технические вопросы, трудные для понимания при случайном исследовании.

Менеджер дисплеев KDE

Администратору многопользовательской системы может потребоваться графический экран для входа пользователей в систему. 37

Вы можете использовать *XDM,* как это описано ранее. Однако в *KDE* имеется альтернативный менеджер kdm, который был разработан более привлекательным и с большим количеством настраиваемых опций для входа в систему. В частности, пользователи могут легко выбирать (посредством меню), какую оболочку (*KDE, GNOME* или что-то ещё) запускать после входа в систему.

Для того, чтобы разрешить запуск *kdm,* измените в файле */etc/ttys* строку, относящуюся к консоли *ttyv8*:

Для *KDE3*:

```
ttyv8 "/usr/local/bin/kdm -nodaemon" xterm on secure
```

Для *KDE4*:

```
ttyv8 "/usr/local/kde4/bin/kdm -nodaemon" xterm on se-
cure
```

XFce

О XFce

XFce является графической оболочкой, построенной на основе инструментального пакета GTK+, используемого в GNOME, но она гораздо легче и предназначена для тех, кому нужен простой, эффективно работающий рабочий стол, который легко использовать и настраивать. Визуально он выглядит очень похоже на CDE, который есть в коммерческих UNIX-системах. Вот некоторые из достоинств *XFce*:

Простой, лёгкий в обращении рабочий стол

- Полностью настраиваемый при помощи мыши, с интерфейсом *drag and drop* и так далее
- Главная панель похожа на CDE, с меню, апплетами и возможностями по быстрому запуску приложений
- Интегрированный оконный менеджер, менеджер файлов, управление звуком, модуль совместимости с *GNOME* и прочее
- Возможность использования тем (так как использует *GTK+)*
- Быстрый, легкий и эффективный: идеален для устаревших/слабых машин или для машин с ограниченной памятью

Установка XFce

Для XFce имеется (на момент написания этого текста) бинарный пакет. Для его установки просто наберите:

```
• pkg_add pkg_add -r xfce4
```

Либо, в случае построения из исходных текстов, используйте Коллекцию Портов:

```
• cd /usr/ports/x11-wm/xfce4
• make install clean
```

Теперь укажите X-серверу на запуск *XFce* при следующем запуске X. Просто наберите:

```
• echo "/usr/local/bin/startxfce4" > ~/.xinitrc
```

При следующем запуске X в качестве рабочего стола будет использоваться *XFce*. Как сказано выше, если используется менеджер дисплеев, такой, как *XDM*, создайте файл *.xsession* так, как это описано в разделе о GNOME, но с командой */usr/local/bin/startxfce4* , либо настройте менеджер дисплеев так, чтобы он разрешил выбор рабочего стола во время входа в систему.

ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Порядок выполнения:

1. Ознакомиться с предложенным материалом для получения информации о настройке графического режима в ОС FreeBSD.
2. Настроить систему и драйверы для поддержки графического режима.
3. Настроить сервер X11 (оболочка KDE).
4. Настроить сервер X11 (оболочка GNOME).

5. Запустить систему в графическом режиме.
6. Ответить на контрольные вопросы и подготовить отчет.

ФОРМА ОТЧЕТА О ВЫПОЛНЕННОЙ РАБОТЕ

Отчет на защиту предоставляется в печатном или электронном виде.

Структура отчета (на отдельном листе(-ах)): титульный лист, формулировка задания (вариант), этапы выполнения работы (со скриншотами), результаты выполнения работы (скриншоты и содержимое файлов).

КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

1. Раскройте понятие X-сервер.
2. Дайте определение термину X-клиент.
3. Опишите роль оконного менеджера.
4. Назовите существующие политики фокусирования.
5. Раскройте понятие виджет.
6. Объясните, что такое Xorg.
7. Опишите алгоритм установки X.
8. Опишите алгоритм, как сконфигурировать X.
9. Перечислите шрифты, которые используются в X.
10. Опишите алгоритм установки дополнительных шрифтов.
11. Опишите назначение менеджеров экранов.
12. Перечислите файлы, участвующие в конфигурировании XDM.
13. Перечислите графические оболочки.
14. Объясните, что такое GNOME.
15. Объясните, что такое KDE.
16. Объясните, что такое XFCE.

ОСНОВНАЯ ЛИТЕРАТУРА

1. Вирт, Н. Разработка операционной системы и компилятора. Проект Оберон [Электронный ресурс] / Н. Вирт, Ю. Гуткнехт. — Москва: ДМК Пресс, 2012. 560 с. Режим доступа: https://e.lanbook.com/book/39992
2. Войтов, Н.М. Основы работы с Linux. Учебный курс [Электронный ресурс]: учебное пособие / Н.М. Войтов. — Москва : ДМК Пресс, 2010. — 216 с. — Режим доступа: URL: https://e.lanbook.com/book/1198
3. Стащук, П.В. Краткое введение в операционные системы [Электронный ресурс] : учебное пособие / П.В. Стащук. — 3-е изд., стер. — Москва : ФЛИНТА, 2019. — 124 с.— URL: https://e.lanbook.com/book/125385

ДОПОЛНИТЕЛЬНАЯ ЛИТЕРАТУРА

4. Войтов, Н.М. Администрирование ОС Red Hat Enterprise Linux. Учебный курс [Электронный ресурс] : учеб. пособие — Москва: ДМК Пресс, 2011. 192 с. Режим доступа: https://e.lanbook.com/book/1081
5. Стащук П.В. Администрирование и безопасность рабочих станций под управлением Mandriva Linux: лабораторный практикум. [Электронный ресурс]: учебно-методическое пособие / П.В. Стащук. — 2-е изд., стер. - М: Флинта, 2015. https://e.lanbook.com/book/70397

ЭЛЕКТРОННЫЕ РЕСУРСЫ:

1. Научная электронная библиотека http://eLIBRARY.RU.
2. Электронно-библиотечная система http://e.lanbook.com.
3. Электронно-библиотечная система «Университетская библиотека онлайн» http://biblioclub.ru.

4. Электронно-библиотечная система IPRBook http://www.iprbookshop.ru/
5. Losst - Linux Open Source Software Technologies https://losst.ru