

Фёдоров С. А.

СОЗДАНИЕ ИНСТРУМЕНТАЛЬНОЙ ПЛАТФОРМЫ НА БАЗЕ GNU/LINUX

Санкт-Петербург
Редакция от 25.12.2017

Содержание

Обновления.....	4
Введение.....	5
1 Установка дистрибутива GNU/Linux.....	6
1.1 Требования к дистрибутиву для создания инструментальной платформы.....	6
1.2 Подготовка носителя с дистрибутивом Debian.....	7
1.2.1 Определение архитектуры и выбор дистрибутива.....	7
1.2.2 Запись образа на носитель.....	8
1.3 Порядок установки дистрибутива.....	8
1.3.1 Запуск установщика на системе.....	8
1.3.2 Выбор языка / Choose language.....	9
1.3.3 Настройка клавиатуры.....	9
1.3.4 Поиск и монтирование CD-ROM.....	10
1.3.5 Загрузка компонентов программ установки с компакт-диска.....	10
1.3.6 Определение сетевой карты.....	10
1.3.7 Настройка сети.....	10
1.3.8 Настройка учётных записей пользователей и паролей.....	11
1.3.9 Настройка времени.....	11
1.3.10 Разметка дисков.....	11
1.3.11 Установка базовой системы.....	12
1.3.12 Настройка менеджера пакетов.....	12
1.3.13 Выбор и установка программного обеспечения.....	13
1.3.14 Установка системного загрузчика GRUB на жёсткий диск.....	13
1.3.15 Установка системных часов.....	14
2 Запуск системы с помощью загрузчика GRUB.....	15
2.1 Разрешение сложностей с запуском загрузчика GRUB.....	15
2.2 Установка загрузчика GRUB в раздел EFI.....	15
2.2.1 Запуск одной из систем Linux.....	16
2.2.2 Установка загрузчика GRUB2 в раздел EFI.....	16
2.2.3 Возможные сложности.....	18
3 Настройка инструментальной платформы.....	19
3.1 Настройка времени.....	19
3.2 Настройка менеджера пакетов.....	19
3.3 Настройка хранилищ и приоритета дистрибутива.....	20
3.3.1 Подключение основных хранилищ.....	20
3.3.2 Подключение тестируемого хранилища.....	21
3.3.3 Подключение хранилища Dropbox.....	21
3.3.4 Настройка используемых по умолчанию пакетов.....	21
3.4 Установка подключения к WiFi или Ethernet.....	22
3.4.1 Определение отсутствующих микропрограмм для WiFi, Ethernet и	

Bluetooth.....	23
3.4.2 Установка необходимых пакетов с микропрограммами при наличии подключения к Интернет.....	23
3.4.3 Установка необходимых пакетов с микропрограммами без подключения к Интернет.....	25
3.4.4 Настройка WiFi.....	27
3.5 Установка пакетов.....	28
3.6 Загрузчик.....	29
3.7 Обновление графического сервера X.....	29
3.8 Автоматический вход в виртуальную консоль.....	29
3.9 Автоматический запуск графического режима из под пользователя.....	30
3.10 Клиент Transmission.....	30
3.11 Клиент Яндекс.Диск.....	30
3.12 Клиент Dropbox.....	31
3.13 Офисный пакет LibreOffice.....	31
3.14 Поддержка модуля Flash PPAPI в Icedove (a la Firefox).....	31
3.15 Подключение хранилищ PPA.....	31
3.15.1 Вариант 1. Создание отдельного файла со списком хранилищ.....	31
3.15.2 Вариант 2. Добавление хранилища в существующий список.....	32
3.16 Менеджер файлов Emelfm2.....	32
3.17 Поддержка HiDPI.....	33
3.18 Настройка оконного менеджера awesome.....	33
3.18.1 Создание снимков экрана.....	33
4 Qt.....	35
5 GTK+.....	36
5.1 Тема.....	36
6 GIMP.....	37

Обновления

25.11.16

- добавлен раздел «Запуск системы с помощью загрузчика GRUB».

Введение

В этом руководстве по шагам описано, как создать эффективную инструментальную среду на базе ОС GNU/Linux. Описание не требует дополнительного опыта, но оно приводится как для будущих, так и для текущих разработчиков ПО.

Не стоит бояться, ведь самостоятельная установка и настройка дистрибутива с самого начала является прекрасным и общепринятым способом вхождения в работу с системой.

В самом начале вашей работы необязательно выполнять действия, направленные на повышения эффективности работы с системой, изложенные в дополнительных главах. Вы можете возвращаться к этому руководству по мере вашего продвижения и накопления опыта.

Никогда не пасуйте перед трудностями при настройке и работе с вашим инструментом, как то уход на другой дистрибутив или вообще использование виртуальной машины. Использование компьютера как инструмента, а не бытового устройства — это часть вашей профессии.

Пытливости интереса вам и большого удовольствия!

1 Установка дистрибутива GNU/Linux

1.1 Требования к дистрибутиву для создания инструментальной платформы

Инструментальная платформа (instrumental platform) — программная платформа по разработке программного обеспечения, являющаяся инструментом в работе *разработчика ПО* (software developer).

Инструментальная платформа разработчика во многом отличается от *целевой платформы* (target platform), используемой пользователями и тестирующими ПО. Как и в любой другой профессии инструмент специалиста отличается от того, что обычно используют обыватели.

Требования к дистрибутиву для создания на его основе инструментальной платформы для разработки программного обеспечения:

1. Надёжность работы дистрибутива, обеспечиваемая политикой безопасности и политикой по работе с пакетами.
2. Безопасность работы с дистрибутивом.
3. Обширное число пакетов, входящих в официальную ветку дистрибутива.
4. Поддержка большого числа аппаратных платформ.
5. Применение дистрибутива в ответственных задачах.
6. Наличие пакетов с широко используемыми статическими и динамическими библиотеками для разработки ПО.
7. Эффективность работы с дистрибутивом.

В данном руководстве рассматривается дистрибутив Debian GNU/Linux, обеспечивающий приведённые выше требования следующим образом (кратко):

1. Политика Debian Policy (www.debian.org/doc/debian-policy/). Наличие следующих веток в дистрибутиве: стабильная, тестируемая, нестабильная, экспериментальная. Специальная политика по работе с пакетами и их миграции между ветками.
2. Собственная команда безопасности Debian, заслужившая уважение среди компаний (например, эксклюзивное разрешение от Mozilla называть обозре-

ватель Firefox с наложенными на него заплатами от Debian также Firefox – итог работы за многие годы, см. bugs.debian.org/cgi-bin/bugreport.cgi?bug=815006).

3. Поддерживает в основной ветке около 60 000 пакетов.
4. Поддерживает 8 платформ, включая ARM, IA-32, IA-64, MIPS, PowerPC, x86-64, z/Architecture. Применяется во встраиваемых, настольных, серверных решениях, а также в высокопроизводительных системах и системах специального назначения.
5. Используется в серверах и ответственных настольных решениях (включая даже облачный сервис Azure от Microsoft). Используется многими коммерческими компаниями, включая Google, а также компаниями, разрабатывающими решения ответственного применения, включая предприятия Минобр. Изначально разрабатывался для надёжной работы, а не для обывателей-пользователей, использующих компьютер как бытовое устройство (обывателей).
6. Включает как статические, так и динамические библиотеки (в отличие от, например, Arch Linux).
7. Не требует сборки пакетов из исходных кодов на инструментальной платформе (в отличие от, например, Gentoo).

1.2 Подготовка носителя с дистрибутивом Debian

1.2.1 Определение архитектуры и выбор дистрибутива

Определите, процессор какой архитектуры используется в системе. Чаще всего это AMD64 (независимо от производителя процессора). Для этого можно обратиться к таким ресурсам как ark.intel.com или www.amd.com .

Рекомендуется следовать «Руководству по установке Debian GNU/Linux»: www.debian.org/releases/stable/amd64/.

У дистрибутива ОС Debian GNU/Linux есть множество образов, которые можно использовать для установки. Ниже представлены некоторые из них.

1. При наличии подключения к Интернет рекомендуется использовать сетевую установку с минимальным образом компакт-диска: www.debian.org/CD/netinst/ При этом основные пакеты при установке будут

загружаться из сети.

2. При отсутствии подключения к Интернет можно использовать образ первого CD- или DVD-диска: www.debian.org/CD/http-ftp/#testing Это позволит вам установить базовую систему без доступа к Интернет, а затем провести её последующую настройку с использованием этой сети.

1.2.2 Запись образа на носитель

Одним из удобных способов установки является запись образа на носитель USB с последующей загрузки с него. Для записи образа на носитель на системах POSIX (Linux, OS X, Unix) можно следовать указаниям в «Руководстве по установке Debian GNU/Linux» (параграф 4.3. 1). Для записи на системах Windows Nt можно использовать такое ПО, как [Win32 Disk Imager](#) и [Rufus](#).

При желании проводить установку с использованием оборудования WiFi или Ethernet, производители которых поставляют для него *драйвера с закрытым исходным кодом*, необходимо предоставить системе *микропрограммы* для этой аппаратуры. Из-за того, что они являются *несвободными*, они не поставляются на установочных образах. Для включения несвободных микропрограмм в образ, достаточно скопировать пакеты с ними на носитель с образом: <http://cdimage.debian.org/cdimage/unofficial/non-free/firmware/> Также можно использовать неофициальный образ Netinst с изначально включёнными в него микропрограммами: <http://cdimage.debian.org/cdimage/unofficial/non-free/cd-including-firmware/> Дополнительную информацию о микропрограммах и порядке работы с ними можно почерпнуть здесь: <https://wiki.debian.org/Firmware>

1.3 Порядок установки дистрибутива

1.3.1 Запуск установщика на системе

При загрузке в режиме UEFI может оказаться невозможным загрузиться с внешних носителей и приводов. Для обеспечения такой возможности достаточно в BIOS установить *унаследованный режим загрузки* (legacy boot mode), использовавшийся прежде – до систем с разделами UEFI. Также необходимо снять режим Secured Boot, раз будет устанавливаться сторонняя ОС.

После загрузки с установочного накопителя установщика ОС Debian GNU/Linux нужно *обязательно* выбрать Advanced options → Expert install. Этим следуют разработчики ПО и администраторы по следующим причинам:

- установка проводится в псевдографике и исключает ошибки, связанные с поддержкой видеокарты на системе;
- установка проводится эффективно с использованием клавиатуры;
- установщик в псевдографике намного лучше оттестирован и отлажен; он разрабатывается в первую очередь.

Ниже приведены рекомендации по настройке на всех этапах установщика. Не забывайте *обращаться к официальному Руководству* и никогда не пасуйте перед трудностями при настройке и работе с вашим инструментом.

1.3.2 Выбор языка / Choose language

Выбор языка (Choose language)

Рекомендуется выбрать родной язык в качестве языка используемого по умолчанию. Это позволит нам эффективнее работать в системе, локализованной под язык. Например, для перехода к русскому языку достаточно два раза нажать «г», перепрыгнув через румынский язык.

Выбор местонахождения

Установка региональных настроек

На ряду с нашей родной локалью добавим *en_US.utf-8*.

По умолчанию лучше использовать родную для вас локаль.

1.3.3 Настройка клавиатуры

Выберим раскладку, используемую на вашей клавиатуре. Она остаётся русской даже если мы используем, например, ноутбук не локализованный для России, но с выгравированными лазером русскими буквами на его клавиатуре.

Это может показаться странным, но наиболее удобным способом переключения между национальной раскладкой и стандартной латинской раскладкой является клавиша «AltGr» или «Caps Lock». В отличие от общепринятых способов на других системах такой способ требует только одной клавиши, которая при этом не связана с другими клавиатурными комбинациями в приложениях, как например, Alt, Win, Alt Gr. При использовании «Caps Lock» переключение между заглавными и строчными буквами проводится по «Shift + Caps Lock», что, заметим, требуется крайне редко по сравнению с переключением раскладки, которую теперь

можно проводить нажатием большую кнопку мизинцем.

В этом и состоит эффективность этого метода, к которому можно очень быстро привыкнуть и от которого не захочется потом уходить.

1.3.4 Поиск и монтирование CD-ROM

Если вы используете носители или накопители на дисках, то все они будут найдены для дальнейшего использования при установке.

1.3.5 Загрузка компонентов программ установки с компакт-диска

Если у нас нет необходимости проводить при установке дополнительные действия, как то работа через OpenSSH или ручная разметка накопителя с использованием другого ПО, то ненужно здесь что-то выбирать.

1.3.6 Определение сетевой карты

При определении нашего сетевого оборудования, возможно, понадобится установка несвободного ПО. Например, микропрограммы для модуля WiFi `iwlfwif-xxxx-x.ucode`. Если мы хотим *уже на этапе установки* иметь подключение к Интернету через WiFi, то необходимые микропрограммы можно разместить на носителе USB перед установкой (см. Подготовка носителя с дистрибутивом Debian). В противном случае это ПО можно установить после установки дистрибутива, используя хранилище пакетов с несвободным ПО (см. Установка подключения к WiFi или Ethernet). Здесь же можно записать, какие микропрограммы требуются для работы WiFi (например, `rtl8106e-1.fw` (при настройке сети установщик на второй системе также запрашивал файл `rt3290.bin`)).

1.3.7 Настройка сети

Если ваш маршрутизатор или провайдер раздаёт IP-адреса автоматически, то можно использовать клиент DHCP для их получения. В противном случае необходимо настроить сеть вручную, указав: IP-адрес, маску подсети, шлюз и адреса серверов DNS.

Имя компьютера обычно совпадает с моделью ноутбука или прозвищем системы. Имя домена можно опустить.

1.3.8 Настройка учётных записей пользователей и паролей

Рекомендуется включить поддержку теневого паролей. Обязательно разрешим

вход в систему суперпользователю (выполнять команды от имени суперпользователя через `sudo` крайне не рекомендуется). Пароль суперпользователя не должен быть тривиальным и при этом забываемым. Конечно, опыт переустановки системы из-за забытого пароля или взлома системы из-за его банальности поможет нам выработать правильное отношение к этому вопросу. Но представим, что это уже с нами происходило, и предложим сразу грамотный пароль.

Создаём учётную запись обычного пользователя. Именно под ней, а не под суперпользователем мы будем работать большую часть своего времени, даже собирая ПО для марсохода из сверхсекретных исходных кодов. Всегда нужно разделять разработку и администрирование системы. Полным именем пользователя в нашей культуре принято считать имя, отчество и фамилию, например: «Акакий А. Башмачкин». Имя учётной записи состоит из одного слова на латинском языке и является, как правило, именем пользователя или его прозвищем. Рекомендации к выбору пароля те же.

1.3.9 Настройка времени

Настройку времени лучше проводить, используя NTP.

1.3.10 Разметка дисков

Крайне не рекомендуется выполнять автоматическую разметку диска. В этом случае вы можете потерять данные, находящиеся на других разделах. Эта практика касается установки любой ОС. Если от разметки у разработчика возникает страх как у хирурга от операционного стола, то стоит сперва разрешить это, а потом вернуться к этому вопросу.

Итак мы выбираем установку вручную. Мы видим все ваши накопители, а также разделы на них. Для установки и последующей эффективной работы рекомендуется выделить 20 ГиБ (потребуется для сборки ядра Linux). Выбираем раздел, который мы хотим уменьшить в размере (данные на нём сохранятся). Уменьшив его размер на, например, 20 ГиБ. В свободном пространстве необходимо создать раздел с журналируемой файловой системой Ext4 и точкой монтирования /. Именно на нём будет размещаться корень файловой системы. Его необходимо сделать загрузочным, чтобы через загрузчик GRUB запускать все ОС в системе.

При большом числе пользователей на системе рекомендуется вынести директорию `/home` с файлами пользователей в отдельный раздел.

Если используется накопитель на жёстких магнитных дисках, а не твердотельный накопитель, то в целях защиты от частой фрагментации данных имеет смысл создать отдельный раздел с часто изменяемыми файлами с точками монтирования `/var` и `/tmp`.

При наличии нескольких разделов *рекомендуется* создавать их в специальном разделе LVM. Это позволит затем *менять* размеры разделов, расположенных в нём (например `/home`, `/var` и т. д.).

Раздел для пространства *подкачки* (swap), расширяющего виртуальную память системы, имеет смысл создавать, только если для профессиональной работы не достаточно имеющегося объёма оперативной памяти. При наличии ОЗУ от 2 ГиБ и использовании профессиональных менеджеров окон (например, awesome — см. ниже) обычно он *не требуется*. Это позволяет системе эффективно размещать страницы памяти только в ОЗУ.

1.3.11 Установка базовой системы

При наличии подключения к сети Интернет основные пакеты будут загружаться именно оттуда.

При выборе ядра установки необходимо выбрать пакет с общим названием без указания версии. Например, `linux-image-amd64`. Он зависит от актуальной версии пакета с образом ядра и потому будет обновляться по мере обновления последнего. При выборе же пакета с конкретной версией ядра такого может не происходить.

Initrd рекомендуется выбирать специализированным. Это позволит системе загружаться эффективнее. К тому же разработчик всегда знает обо всех изменениях в аппаратной части системы.

1.3.12 Настройка менеджера пакетов

Необходимо установить использование зеркала архива из сети. Зеркало должно территориально располагаться как можно ближе к провайдеру. Например, можно использовать зеркало Яндекс или, что надёжнее, — `ftp.ru.debian.org`. Протокол FTP предпочтительнее.

Использование несвободного ПО явно выделено согласно политике Debian и оно рекомендуется (например, для работы WiFi и проигрывания содержимого

Flash в обозревателе Firefox). Это всегда можно будет изменить.

Рекомендуется при установке подключить как хранилища с обновлениями безопасности и обновлениями выпуска, так и с бэкпортированным ПО (backported software). Бэкпортированное ПО – это ПО, перенесённое назад – из тестируемой в стабильную ветку, то есть пакеты популярного ПО из тестовой ветки с более поздними и менее проверенными версиями, специально собранные для стабильной ветки. Это хранилище полезно использовать, например, для установки свежей версии LibreOffice, что будет описано ниже. Более того, будет показано, как одновременно использовать пакеты из стабильной, тестируемой и при желании нестабильной ветки дистрибутива, оставляя при этом саму систему на стабильной ветке.

1.3.13 Выбор и установка программного обеспечения

Рекомендуется участвовать в опросе популярности пакетов. В этом случае разработчик может влиять на отображение тех пакетов, которые он использует.

При создании инструментальной платформы разработчика необходима установка *только* «Стандартных системных утилит» и при необходимости ПО для ноутбука. Установка графического сервера, а также менеджера окон для разработчиков будет проведена отдельно.

1.3.14 Установка системного загрузчика GRUB на жёсткий диск

Рекомендуется установить системный загрузчик GRUB в *главную загрузочную запись* (main boot record, MBR) для того, чтобы потом проводить с его помощью загрузку всех ОС на системе. Иначе придётся редактировать другой загрузчик для возможности запуска с его помощью ОС GNU/Linux (загрузчики, поставляемые с ОС Windows Nt, делают это с меньшим успехом).

Также при запуске системы через EFI рекомендуется выполнить принудительную установку GRUB в путь для съёмных носителей EFI, тогда при запуске системы в худшем случае можно будет выбирать загрузчик через меню загрузки (см. руководство к ситсеме).

1.3.15 Установка системных часов

Часто система не использует часы в формате UTC, указывая лишь местное время.

2 Запуск системы с помощью загрузчика GRUB

Если система способна загружать сторонние ОС, то после перезагрузки запустится загрузчик GRUB, где можно будет выбрать нужную ОС.

2.1 Разрешение сложностей с запуском загрузчика GRUB

В системах с разделами EFI и «защитой» по запуску сторонних ОС могут возникать сложности по запуску загрузчика GRUB. Необходимо реализовать следующие подходы, пока они не разрешат проблемы с загрузкой сторонней ОС на используемой системе.

1. Поставить в BIOS запуск в унаследованном режиме загрузки (Boot BIOS – Legacy) и установить *первой по приоритету* загрузку с винчестера (накопителя) (см. руководство по системе). Если загрузчик GRUB смог прописаться в MBR, то запустится именно он.
2. Если при выборе в *меню загрузки* (см. руководство по системе, обычно по F9) другого загрузчика в UEFI появляется предупреждение системы безопасности: «Can not authenticate boot image», запрещающей запуск сторонних загрузчиков и ОС, то достаточно убрать защищённую загрузку BIOS (Secure boot BIOS: disable).
3. Обновить BIOS (см. руководство по системе). Возможно, в BIOS появится опция выбора загрузчика EFI (OS Boot manager или debian).
4. Установить загрузчик GRUB в раздел EFI (см. следующий параграф).
5. Обновить загрузчик GRUB из тестовой ветки дистрибутива (см. в руководстве ниже) и повторить действия п. 4, переустановив GRUB в раздел EFI.

2.2 Установка загрузчика GRUB в раздел EFI

Большинство систем имеет «защиту» от установки и запуска сторонних ОС. Порой эта защита обеспечивается тем, что система всегда загружается с помощью загрузчика, который установлен в раздел EFI системы, независимо от того, что записано в главной загрузочной записи. Для того, чтобы загрузчик GRUB2 мог загружаться после BIOS, его необходимо установить именно в этот раздел после установки системы.

Подробности:

https://wiki.debian.org/GRUBEFIReinstall#Boot_your_computer_with_the_Refind_media

2.2.1 Запуск одной из систем Linux

Установка загрузчика в раздел EFI производится с помощью любой системы Linux. Для этих целей можно использовать как минимум один из вариантов, представленных ниже.

1. Собственная система.

1. При загрузке системы перейти в меню выбора загрузки (обычно F9, см. руководство по системе). Выбрать там раздел с системой Linux (не менеджер загрузки ОС – OS Boot manager), тем самым загрузив свою систему.
2. Если загрузчик ОС Linux не виден в списке, то можно попробовать в BIOS снять (если стоял) режим Secured Boot, установить режим загрузки Legacy и повторить п. 1.

2. Загрузить систему Live CD/DVD/USB.

3. Загрузить с USB менеджер загрузки The rEFInd Boot Manager:
<http://www.rodsbooks.com/refind/getting.html>

2.2.2 Установка загрузчика GRUB2 в раздел EFI

Запустив систему одним из предложенных выше способов, откройте терминал, в котором необходимо провести установку загрузчика в раздел EFI:

- Перейдите на суперпользователя, введя его пароль:

```
su
```

- Проверьте, в каком режиме запущена система:

```
[ -d /sys/firmware/efi ] && echo "EFI" || echo "Legacy"
```

должна вернуть "EFI".

- Определите узел, который отображает раздел EFI (если хотите, аналог меток C, D в системах Windows Nt). Если он расположен на первом накопителе, то он называется /dev/sda*. Если на накопителе с системой не поставлялся раздел для восстановления, то у него будет первый номер (в противном случае, вероятно, что второй):

```
fdisk -l
```

```
...  
/dev/sda1 ... .. EFI System
```

- Проверьте, подмонтирован ли этот раздел в директорию вроде /boot/efi:

```
df  
/dev/sda1 ... .. xx% /boot/efi
```

- Если нет, то создайте директорию для его подключения:

```
mkdir /boot/efi
```

и подмонтируйте раздел EFI к этой директории:

```
mount /dev/sda1 /boot/efi
```

- Переустановите пакет grub-efi.
 - Для этого вам придётся вновь подмонтировать носитель, с которого проводилась установка, или подключиться к Интернет (см. разделы ниже). Для первого варианта необходимо установить носитель в систему и определить узел, на который он отображается:

```
fdisk -l
```

```
...  
/dev/sdb1 ... .. FAT или Empty
```

- После этого можно смело его монтировать:

```
mount /dev/sdb1 /media/cdrom
```

- Как только у вас появился доступ к хранилищу – локальному или удалённому – вы можете переустановить пакет grub-efi:

```
apt-get install --reinstall grub-efi
```

- Установить загрузчик в раздел EFI, указав не раздел EFI, а весь накопитель, где он располагается, и создать соответствующую запись в NVRAM:

```
grub-install /dev/sda
```

- Обновите конфигурационный файл загрузчика согласно новой схеме разбиения ваших разделов:

```
update-grub
```

- Проверьте, что загрузчик установлен корректно:
 - Убедитесь в наличии загрузчика GRUB в разделе EFI (используйте Tab для автодополнения путей, что не вводить всё целиком):

```
file /boot/efi/EFI/debian/grubx64.efi
```

- Убедитесь в корректной записи в NVRAM:


```
efibootmgr --verbose
```

и

```
efibootmgr --verbose | grep -i debian
```

- Перезагрузите систему, чтобы увидеть приветствие GRUB. Позже вы сможете изменить его стандартно поведение (время ожидания, пункт загрузки по умолчанию и т. д.).

2.2.3 Возможные сложности

Даже если программа `efibootmgr --verbose` отображает адекватный порядок запуска загрузчиков, система может *сознательно* не следовать требованиям EFI и игнорировать установленный порядок запуска. В этом случае можно попробовать другие средства из параграфа 2.1.

3 Настройка инструментальной платформы

После установки и успешного запуска системы необходимо провести её настройку *как инструментальной платформы*.

3.1 Настройка времени

Определить текущее время можно по команде `date`.

Если наше время отличается от системного, то, возможно, вы неправильно задали параметры при установке. Настройки времени можно поменять с помощью суперпользователя:

```
su
# dpkg-reconfigure tzdata
```

3.2 Настройка менеджера пакетов

При разработке ПО на инструментальной платформе часто приходится использовать *разные* хранилища пакетов наряду со стабильными хранилищами – как *тестовые*, так и вообще хранилища *других дистрибутивов*. ОС Debian позволяет эффективно работать со множеством хранилищ без необходимости обновлять все пакеты до самых свежих из них, т. е. *нестабильных*. Чтобы оперировать такими пакетами на инструментальной платформе используется менеджер пакетов `aptitude`, обладающий необходимой *интерактивностью* и *надёжностью*.

После запуска менеджера пакетов `aptitude` рекомендуется его настроить, выставив следующие настройки (в меню (Ctrl+T): Параметры → Предпочтения):

- убрать «Показывать некоторые из доступных команд вверху экрана»: рекомендуется убрать, как только вы познакомитесь с основными клавишами управления менеджером;
- убрать «Скрывать строку меню, когда она не используется»: рекомендуется для эффективного задействования пространства экрана (вызов меню по Ctrl+T);
- убрать «Запрашивать подтверждение при выходе»: экономит время;
- убрать «Автоматически устанавливать рекомендуемые пакеты»: при удалении определённого пакета, будут также удаляться ненужные пакеты, которые были установлены автоматически для удовлетворения его зависимо-

стей; рекомендуемые пакеты при этом оставляются; снятие этой опции позволяет избавиться от засорения системы.

Автоматизация настройки. Достаточно скопировать файл config со следующим содержанием:

```
aptitude "";
aptitude::Keep-Unused-Pattern "";
aptitude::Delete-Unused-Pattern "";
aptitude::UI "";
aptitude::UI::Menubar-Autohide "true";
aptitude::UI::Prompt-On-Exit "false";
aptitude::UI::HelpBar "false";
Apt "";
Apt::Install-Recommends "false";
в директорию /root/.aptitude/:
cp -f ./aptitude/config /root/.aptitude/config
```

3.3 Настройка хранилищ и приоритета дистрибутива

3.3.1 Подключение основных хранилищ

Используемые системой *хранилища* (repositories) указываются в файле /etc/apt/sources.list и в файлах директории /etc/apt/sources.list.d. Рекомендуемый список хранилищ состоит из:

- стабильной ветки – **stretch** (Debian 9.x);
- обновлений безопасности stretch – **stretch/updates**;
- предлагаемых обновлений для stretch – **stretch-updates**;
- ПО, *перенесённого назад* (backported software) – **stretch-backports**, т. е. ПО, перенесённого назад из тестируемой ветки stretch в стабильную stretch (обычно это очень популярные пакеты, например, такие как LibreOffice, ядро Linux, Firefox, тестируемые версии которых непременно хочется иметь в стабильной ветке, без переноса в неё *зависимостей* этих пакетов из тестируемой ветки).

При этом для каждого хранилища можно указать, пакеты какого рода использовать из него – *основные* (main), *несвободные* (non-free) и *внесённые* (contrib). В файле sources.list это задаётся следующим образом (с помощью текстового редактора vim или nano):

```
deb ftp://ftp.ru.debian.org/debian/ stretch main non-free contrib
deb-src ftp://ftp.ru.debian.org/debian/ stretch main non-free contrib
```

```
deb http://security.debian.org/ stretch/updates main contrib non-free
deb-src http://security.debian.org/ stretch/updates main contrib non-free
```

```
# stretch-updates, previously known as 'volatile'
deb ftp://ftp.ru.debian.org/debian/ stretch-updates main contrib non-free
deb-src ftp://ftp.ru.debian.org/debian/ stretch-updates main contrib non-free

# stretch-backports, previously on backports.debian.org
deb ftp://ftp.ru.debian.org/debian/ stretch-backports main contrib non-free
deb-src ftp://ftp.ru.debian.org/debian/ stretch-backports main contrib non-free
```

Совет. Не стоит удалять хранилище, связанное с носителем, использованным при установке, если вы хотите установить ПО именно с него (например, не имея доступа к Интернет).

3.3.2 Подключение тестируемого хранилища

В директорию `/etc/apt/sources.list.d` рекомендуется добавить отдельный файл `testing.list`, содержащий закомментированные хранилища с тестируемыми пакетами – `buster` и `buster-updates`:

```
# buster
#deb ftp://ftp.ru.debian.org/debian/ buster main non-free contrib
#deb-src ftp://ftp.ru.debian.org/debian/ buster main non-free contrib

# buster-updates
#deb ftp://ftp.ru.debian.org/debian/ buster-updates main contrib non-free
#deb-src ftp://ftp.ru.debian.org/debian/ buster-updates main contrib non-free
```

При необходимости задействовать тестовые хранилища их можно будет раскомментировать и обновить информацию о пакетах:

```
aptitude -u
```

3.3.3 Подключение хранилища Dropbox

Для установки клиента Dropbox необходимо также создать файл с хранилищем для него `/etc/apt/sources.list.d/dropbox.list`:

```
deb [arch=i386,amd64] http://linux.dropbox.com/debian stretch main
```

3.3.4 Настройка используемых по умолчанию пакетов

Какие бы ни были подключены хранилища на инструментальной платформе необходимо оставаться преимущественно на *стабильных* пакетах. При наличии хранилищ со свежими – тестируемыми или нестабильными – пакетами менеджер пакетов будет предлагать обновляться до них («g» в `aptitude`). Чтобы этого не происходило, необходимо сообщить системе дистрибутив для установки пакетов

по умолчанию, создав файл /etc/apt/apt.conf:

```
APT::Default-Release "stretch";
```

Также необходимо создать файл /etc/apt/preferences.d/pinning.pref, определяющий приоритет при установке пакетов из хранилищ:

```
Package: *  
Pin: release a=stable  
Pin-Priority: 900  
Package: *  
Pin: release a=stretch-backports  
Pin-Priority: 0  
Package: *  
Pin: release a=testing  
Pin-Priority: -500
```

Теперь после обновления информации о пакетах их, можно будет устанавливать пакеты с любых подключённых хранилищ, но менеджер пакетов не будет предлагать обновляться до тестируемых и нестабильных пакетов и будет их устанавливать только при отсутствии другого решения.

Используя такой подход, можно подключить к системе *тестируемые* (testing), *нестабильные* (unstable) и *экспериментальные* (experimental) хранилища, а также хранилища от *других* дистрибутивов ОС GNU/Linux, оставаясь при этом в основном на *стабильной* системе. Так очень часто поступают профессиональные разработчики и администраторы систем. По этой причине для них выражения о том, что Debian содержит не последние версии пакетов, выглядят по меньшей мере наивными и потребительскими.

Автоматизация настройки. При наличии необходимых файлов sources.list, sources.list.d/testing.list, sources.list.d/dropbox.list, apt.conf, preferences.d/pinning.pref в резервной директории /apt для автоматизации приведённой настройки достаточно их скопировать на место конфигурационных файлов системы:

```
cp -Rf ./apt/ /etc/
```

3.4 Установка подключения к WiFi или Ethernet

Если дальнейшую установку решено проводить, получая пакеты с хранилищ из Интернет, используя подключение по WiFi, то прежде необходимо настроить такое соединение.

3.4.1 Определение отсутствующих микропрограмм для WiFi, Ethernet и Bluetooth

Если при установке на образ не были помещены *несвободные* микропрограмм-

мы, то возможно их отсутствие в системе. Проверить *отсутствующие* микропрограммы:

```
cat /var/log/syslog.1 | grep -i failed
```

Пример 1. Контроллер WiFi и Bluetooth Intel.

```
iwlwifi 0000:03:00.0: firmware: failed to load iwlwifi-7265-9.ucode (-2)
iwlwifi 0000:03:00.0: Direct firmware load failed with error -2
iwlwifi 0000:03:00.0: firmware: failed to load iwlwifi-7265-8.ucode (-2)
iwlwifi 0000:03:00.0: Direct firmware load failed with error -2
bluetooth hci0: firmware: failed to load intel/ibt-hw-37.8.10-fw-1.10.3.11.e.bseq (-2)
bluetooth hci0: Direct firmware load failed with error -2
Bluetooth: hci0 failed to open Intel firmware file: intel/ibt-hw-37.8.10-fw-1.10.3.11.e.bseq(-12)
bluetooth hci0: firmware: failed to load intel/ibt-hw-37.8.bseq (-2)
bluetooth hci0: Direct firmware load failed with error -2
Bluetooth: hci0 failed to open default Intel fw file: intel/ibt-hw-37.8.bseq
```

Необходимые файлы с микропрограммами: iwlwifi-7265-9.ucode, iwlwifi-7265-8.ucode, ibt-hw-37.8.10-fw-1.10.3.11.e.bseq.

Пример 2. Контроллер WiFi Realtek.

```
r8169 0000:05:00.0: firmware: failed to load rtl_nic/rtl8106e-1.fw (-2)
r8169 0000:05:00.0: Direct firmware load failed with error -2
```

Необходимые файлы с микропрограммами: rtl8106e-1.fw.

Пример 3. Контроллер Ethernet Ralink.

При настройке сети установщик Debian запрашивал файл rt3290.bin.

Теперь нужно найти пакеты, в которых располагаются необходимые микропрограммы.

3.4.2 Установка необходимых пакетов с микропрограммами при наличии подключения к Интернет

При наличии подключения к Интернет, например, по Ethernet для поиска пакета, содержащего необходимую микропрограмму, эффективнее всего использовать информацию с хранилищ. Для этого вначале нужно установить пакет apt-file, позволяющий проводить поиск по содержимому пакетов:

```
aptitude install apt-file
```

Рекомендуется устанавливать последние версии микропрограмм, взяв их из тестируемой ветки, поэтому необходимо раскомментировать это хранилище в файле /etc/apt/sources.list.d/testing.list.

После этого достаточно обновить *информацию* обо всех пакетах с хранилищ (сами пакеты не загружаются):

```
apt-file update
```

Для поиска пакета с необходимым файлом теперь достаточно использовать apt-file.

Пример 1. Контроллер WiFi и Bluetooth Intel.

```
apt-file -x search "ibt-hw-37.8.10-fw-1.10.3.11.e.bse"  
firmware-iwlwifi: /lib/firmware/intel/ibt-hw-37.8.10-fw-1.10.3.11.e.bseq
```

Если менеджер подключён к *нескольким* хранилищам, то важно уточнить, в *какой конкретно ветке* дистрибутива содержится нужный пакет, *указав файл с хранилищами*, которые нужно использовать. В начале достаточно искать в пакетах *стабильного* выпуска:

```
apt-file -x -s /etc/apt/sources.list search "ibt-hw-37.8.10-fw-1.10.3.11.e.bse"  
firmware-iwlwifi: /lib/firmware/intel/ibt-hw-37.8.10-fw-1.10.3.11.e.bseq  
а затем тестируемого:
```

```
apt-file -x -s /etc/apt/sources.list.d/testing.list search "ibt-hw-37.8.10-fw-1.10.3.11.e.bse"  
firmware-iwlwifi: /lib/firmware/intel/ibt-hw-37.8.10-fw-1.10.3.11.e.bseq
```

В данном случае создаётся впечатление, что необходимый пакет имеется как в стабильной, так и в тестовой ветке. На самом же деле стабильная ветка его не содержит — он находится в ветке с ПО, перенесённым назад (backported software). Эта ветка указана в файле /etc/apt/sources.list и поэтому дала положительный результат.

Приведённые микропрограммы находятся в пакете firmware-iwlwifi. При этом только в тестовой версии этого пакета, поэтому необходимо установить именно её:

```
aptitude install -t testing firmware-iwlwifi
```

Пример 2. Контроллер WiFi Realtek.

```
apt-file -x search "rtl8106e-1.fw"  
firmware-realtek: /lib/firmware/rtl_nic/rtl8106e-1.fw
```

Если менеджер подключён к *нескольким* хранилищам, то важно уточнить, в *какой конкретно ветке* дистрибутива содержится нужный пакет, *указав файл с хранилищами*, которые нужно использовать. В начале достаточно искать в пакетах *стабильного* выпуска:

```
apt-file -x -s /etc/apt/sources.list search "rtl8106e-1.fw"
firmware-realtek: /lib/firmware/rtl_nic/rtl8106e-1.fw
```

а затем *тестируемого*:

```
apt-file -x -s /etc/apt/sources.list.d/testing.list search "rtl8106e-1.fw"
firmware-realtek: /lib/firmware/rtl_nic/rtl8106e-1.fw
```

В данном случае микропрограмма rtl8106e-1.fw предоставляется пакетом firmware-realtek как в стабильной, так и в тестовой ветке. Рекомендуется устанавливать последние версии микропрограмм, взяв их из *тестируемой* ветки:

```
aptitude install -t testing firmware-realtek
```

Пример 3. Контроллер Ethernet Ralink.

```
apt-file -x search "rt3290.bin"
firmware-ralink: /lib/firmware/rt3290.bin
```

Если менеджер подключён к *нескольким* хранилищам, то важно уточнить, в *какой конкретно ветке* дистрибутива содержится нужный пакет, *указав файл с хранилищами*, которые нужно использовать. В начале достаточно искать в пакетах *стабильного* выпуска:

```
apt-file -x -s /etc/apt/sources.list search "rt3290.bin"
firmware-ralink: /lib/firmware/rt3290.bin
```

а затем *тестируемого*:

```
apt-file -x -s /etc/apt/sources.list.d/testing.list search "rt3290.bin"
firmware-misc-nonfree: /lib/firmware/rt3290.bin
```

В данном случае микропрограмма rt3290.bin предоставляется пакетом firmware-ralink в стабильной ветке и пакетом firmware-misc-nonfree в тестируемой ветке. Рекомендуется установить последние версии микропрограмм:

```
aptitude install -t testing firmware-misc-nonfree
```

3.4.3 Установка необходимых пакетов с микропрограммами без подключения к Интернет

Пакеты с нужными файлами можно искать также и на сайте packages.debian.org с помощью другой системы, имеющей доступ к Интернет. Это *менее профессионально*, чем работа с информацией прямо на системе.

Выбрав «пакеты, содержащие файлы с таким именем», стабильный/тестируемый дистрибутив, нужную архитектуру, а также указав нужный файл для поиска, мы находим необходимые пакеты.

Пример 1. Контроллер WiFi и Bluetooth Intel.

В стабильной ветке файл ibt-hw-37.8.10-fw-1.10.3.11.e.bseq не найден, а в тестируемой его предоставляет пакет:

```
/lib/firmware/intel/ibt-hw-37.8.10-fw-1.10.3.11.e.bseq    firmware-iwlwifi
```

Значит необходимо установить этот пакет именно из тестовой ветки.

Пример 2. Контроллер WiFi Realtek.

Микропрограмма rtl8106e-1.fw предоставляется пакетом firmware-realtek как в стабильной, так и в тестовой ветке:

```
/lib/firmware/rtl_nic/rtl8106e-1.fw    firmware-realtek
```

Рекомендуется установить последние версии микропрограмм, взяв их из *тестируемой* ветки.

Пример 3. Контроллер Ethernet Ralink.

Микропрограмма rt3290.bin предоставляется пакетом firmware-ralink в стабильной ветке и пакетом firmware-misc-nonfree в тестируемой ветке.

```
/lib/firmware/rt3290.bin    firmware-ralink
/lib/firmware/rt3290.bin    firmware-misc-nonfree
```

Рекомендуется установить последние версии обеих микропрограмм, взяв их из *тестируемой* ветки.

Установка пакетов. Переходя по названию найденных пакетов, можно убедиться, что они не зависят от других пакетов и что они одинаковы для всех архитектур (all). Перейдя по всем архитектурам, можно скачать необходимые пакеты с выбранного зеркала.

После скачивания необходимых пакетов (например, firmware-iwlwifi_20160824-1_all.deb или firmware-realtek_20160824-1_all.deb), предоставляющих нужные микропрограммы, можно установить их с носителя USB в систему из-под суперпользователя:

```
su
```

Создание директории для монтирования USB:

```
# mkdir /media/usb
```

Определение узла, отвечающего за носитель:

```
# fdisk -l
```

```
...
/dev/sdb1 ... .. FAT
```

Монтирование накопителя:

```
# mount /dev/sdb1 /media/usb
```

Установка пакетов с накопителя, например:

```
# dpkg -i /media/usb/firmware-iwlwifi_20160824-1_all.deb
```

Демонтирование накопителя:

```
# umount /dev/sdb1 /media/usb
```

3.4.4 Настройка WiFi

Для использования протоколов шифрования WPA и WPA2 (IEEE 802.11i) необходимо установить пакет `wpa_supplicant`. При выборе в установщике Debian утилит для ноутбука на ряду со стандартными системными утилитами он устанавливается автоматически. Проверить его наличие можно вызовом `wpa_supplicant`, а установить одним из следующих способов:

- при наличии подключения к Интернет (например, через Ethernet):

```
aptitude install wpa_supplicant
```

- без подключения к Интернет можно зайти на сайт packages.debian.org и указать там пакет `wpa_supplicant`. Перейдя по нужной архитектуре в описании пакета, необходимо скачать его, а также пакеты xxx, от которых он зависит, но которые могут отсутствовать в системе с базовой установкой.

Подробную информацию по настройке WiFi можно почерпнуть отсюда: <https://wiki.debian.org/WiFi>

Конфигурационный файл `/etc/network/interfaces` отвечает за используемые интерфейсы.

Пример. Автоматическое получение адреса IP из сети Home_name.

```
iface home inet dhcp
    wpa-ssid home_wifi_name
    wpa-psk XXXX
```

```
iface phone inet dhcp
    wpa-ssid phone_nwifi_ame
    wpa-psk YYYY
```

```
iface work inet dhcp
    wireless-essid work_wifi_name
```

Поднять интерфейс:

```
ifup wlan0=home
```

Опустить интерфейс:

```
ifdown wlan0
```

Проверка интерфейса:

```
ifconfig
```

Сканирование сетей:

```
iwlist scan
```

3.5 Установка пакетов

После установки базовой системы и подключения к Интернет для создания инструментальной платформы потребуется целый ряд пакетов, позволяющих эффективно заниматься в ней разработкой. Сюда относятся как средства разработки, так и специальные *менеджеры окон* (window managers), эргономично располагающие приложения на экране и позволяющие ими управлять с клавиатуры.

Для установки пакетов необходимо использовать менеджер пакетов aptitude.

Рекомендуемые пакеты для установки приведены ниже:

```
pmount sudo apt-file zip unzip p7zip-full unrar \
reportbug \
software-properties-common \
linux-headers-amd64 firmware-linux intel-microcode debian-kernel-
handbook \
build-essential gcc gfortran automake gcc-doc gfortran-doc gcc-multilib
gfortran-multilib gdb libtool make gcc-4.9-locales \
xorg xfonts-cyrillic awesome \
bash-completion vim-gtk cscope vim-doc gnome-terminal \
dia inkscape/$BP gimp evince-gtk eog zathura zathura-djvu zathura-ps \
firefox-esr{,-l10n-ru} icedove{,-l10n-ru} browser-plugin-freshplayer-
pepperflash pepperflashplugin-nonfree \
mpv imagemagick \
transmission-daemon transmission-remote-cli \
alsa-base alsa-utils alsa-tools \
xscreensaver{,-gl} \
libreoffice/stretch-backports \
libreoffice-{help-ru,ogltrans,pdfimport,gtk3}/stretch-backports \
libreoffice-lightproof{-en,-ru-ru} \
{mythes,hunspell,hyphen}{-en-us,-ru} fonts-liberation \
{libreoffice-{core,common,l10n-ru,style-galaxy,java-common,base-
core,base,base-drivers,calc,draw,impress,math,writer},python3-uno,ure,uno-
libs3}/stretch-backports+M \
ttf-mscorefonts-installer
```

3.6 Загрузчик

- После установки нужной версии ядра в файле /etc/default/grub изменить время ожидания:

GRUB_TIMEOUT=0

- Обновить конфигурацию:

update-grub2

3.7 Обновление графического сервера X

В случае необходимости поддержки последних видеокарт бывает полезным обновить графический сервер X. Для этого рекомендуется *через aptitude* обновить следующие пакеты:

Состояние в aptitude ¹	Пакет
iu	xorg
iuA	xserver-common
iuA	xserver-xorg
iuA	xserver-xorg-core
iuA	xserver-input-all
iuA	xserver-input-evdev
iuA	xserver-input-mouse
iuA	xserver-input-synaptics
iuA	xserver-xorg-video-vesa

Последним оставлен драйвер, обеспечивающий стандартный интерфейс VESA, предоставляемый всеми видеокартами. Необходимо дополнительно установить драйвер для используемой в системе видеокарты.

3.8 Автоматический вход в виртуальную консоль

Инструментальная платформа почти всегда является личной для разработчика. Для защиты системы рекомендуется поставить пароль на уровне BIOS, а для защиты *доступа к накопителю с системой и данными* – пароль на накопитель в BIOS. При этом данные будут защищены даже в случае утери системы.

При таком подходе будут защищены все ОС и данные на накопителе, а вход в систему будет проходить быстрее – без ненужной авторизации. Для этого под именем разработчика необходимо создать файл /etc/systemd/system/getty\@tty1.service:

```
[Service]
ExecStart=
ExecStart=-/sbin/agetty --noclear -a user_name %I $TERM
```

¹ i – предыдущая версия пакета уже установлена, u – пакет предлагается к обновлению, A – пакет устанавливается автоматически для удовлетворения зависимости.

Здесь вместо *user_name* нужно указать имя пользователя для автоматического входа в систему под ним.

3.8.1.1 Автоматизация настройки

При наличии подготовленного файла `systemd/system/getty@tty1.service` будет достаточно скопировать его в нужное место:

```
cp -fR ./systemd/ /etc/
```

3.9 Автоматический запуск графического режима из под пользователя (Debian jessie)

Пользователем создать файл `~/.bash_profile`:

```
if [ -z "$DISPLAY" ] && [ $(tty) == /dev/tty1 ]; then
    yandex-disk start &      # дополнительно
    dropbox start &          # дополнительно
    startx
fi
```

3.10 Поддержка модуля Flash PPA в Firefox

Установить пакеты `browser-plugin-freshplayer-pepperflash` и `pepperflashplugin-nonfree` (включенные в сценарий).

3.11 Подключение хранилищ PPA

Установить пакет `software-properties-common`:

3.11.1 Вариант 1. Создание отдельного файла со списком хранилищ.

1. Найти необходимое хранилище и определить расположение PPA для вашего пакета в формате `ppa:user/ppa-name`. Например, для `Emelfm2` это `ppa:emelfm2/release`.
2. Добавить хранилище (заменить `ppa:user/ppa-name` на нужное расположение):

```
add-apt-repository ppa:user/ppa-name
```

3. Хранилище будет добавлено в отдельный файл в каталоге `/etc/apt/sources.list.d/`, но с именем используемого дистрибутива Debian. В каталоге `/etc/apt/sources.list.d/` открыть *добавленный файл* и поменять название дистрибутива Debian на одно из названий для Ubuntu, для которого этот пакет доступен. Например, с `stretch` на `trusty`:

```
deb http://ppa.launchpad.net/emelfm2/release/ubuntu stretch main
```

```
deb http://ppa.launchpad.net/emelfm2/release/ubuntu trusty main
```

Это можно автоматизировать:

```
sed -i 's/stretch/trusty/g' /etc/apt/sources.list.d/emelfm2-ppa-stretch.list
```

4. Обновите информацию о пакетах:

```
aptitude update
```

3.11.2 Вариант 2. Добавление хранилища в существующий список.

1. Найти необходимое хранилище на Launchpad (например, launchpad.net/~emelfm2/+archive/ubuntu/release).
2. Открыть технические детали (Technical details about this PPA), выбрать нужную версию дистрибутива Ubuntu и добавить хранилище:

```
add-apt-repository "deb http://ppa.launchpad.net/emelfm2/release/ubuntu trusty main"
```

3. Хранилище добавится в основной файл `/etc/apt/sources.list`.
4. Если ключ PPA GPG не был добавлен автоматически, добавить его, например:

```
apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 42C9B570
```

5. Обновите информацию о пакетах:

```
aptitude update
```

3.12 Менеджер файлов Emelfm2

1. Добавить хранилище PPA `ppa:emelfm2/release`
2. Установить пакет `emelfm2`.

Можно привязывать файлы к приложениям для их открытия по умолчанию из контекстного меню: Настроить типы файлов...

3.13 Поддержка HiDPI

```
echo Xft.dpi: 144 | xrdp -merge
```

<https://wiki.debian.org/accessibility>

<https://wiki.debian.org/MonitorDPI>

3.14 Настройка оконного менеджера awesome

3.14.1 Создание снимков экрана

Бывает полезным иметь горячую клавишу для создания снимка экрана. Для этого достаточно создать сценарий `capscr`, создающий файл снимка в директории `$HOME/Pictures/screenshots`:

```
#!/bin/bash
timestamp="$(date +%d.%m.%Y"_shot_"%H:%M:%S)"
targetbase="$HOME/Pictures/screenshots"
mkdir -p $targetbase
[ -d $targetbase ] || exit 1
import -window root -quality 98 $targetbase/$timestamp.png
```

Этот сценарий удобнее поместить в одну из директорий с исполняемыми файлами:

```
cp -f ./awesome/capscr /bin/capscr
```

Также его обязательно нужно сделать исполняемым для всех пользователей:

```
chmod a+x /bin/capscr
```

Вызов этого сценария можно осуществлять по горячей клавише `PrtScr`, добавив указание на это в файл конфигурации `awesome` для пользователя `$HOME/.config/awesome/rc.lua`:

```
-- {{{ Own bindings
globalkeys = awful.util.table.join(
  -- bind PrintScrn to capture a screen
  awful.key(
    {},
    "Print",
    function()
      awful.util.spawn("capscr", false)
    end
  )
)
```

3.14.1.1 Автоматизация настройки

При наличии сценария `capscr` и файла конфигурации `user/rc.lua` автоматизацию настройки можно проводить следующим способом:

- от имени суперпользователя:

```
cp -f ./awesome/capscr /bin/capscr
chmod a+x /bin/capscr
```

- от имени пользователя:

```
mkdir -p /home/$USER/.config/awesome
```

```
cp -f ./user/rc.lua /home/$USER/.config/awesome/rc.lua
```


4 Настройка дополнительного ПО

4.1 Клиент Transmission

- Установить необходимые пакеты с помощью сценария.
- Остановить демон: `/etc/init.d/transmission-daemon stop`
- В файле `/etc/transmission-daemon/settings.json` отключить аутентификацию:

`"rpc-authentication-required": false,`

- Запустить демон: `/etc/init.d/transmission-daemon start`

4.2 Клиент Яндекс.Диск

- Установить пакет `yandex-disk`:

`yandex-disk setup`

4.3 Клиент Dropbox

- Установить пакет `dropbox`.
- Для пользователя выполнить:

```
dropbox start -i
dropbox autostart
```

4.4 Офисный пакет LibreOffice

В опциях Сервис → Параметры:

- поставить: большой размер значков (Вид), единицы измерения сантиметры (LibreOffice Writer → Общие), Автоматически проверять орфографию и Автоматическая расстановка переносов (Настройки языка → Лингвистика);
- рекомендуется заменить шрифт по умолчанию на Times NewRoman, а для заголовков – Arial (LibreOffice Writer → Основные шрифты). Это позволит при сохранении документов в закрытых форматах DOC, DOCX применять в них стандартные шрифты и передавать такие документы специалистам, использующим Microsoft Word;
- убрать: использование виртуальной машины Java (Специальные возможности);
- при использовании тёмной темы можно также поставить распознавание

контрастного режима операционной системы и автоматически подбирать цвет для отображения текста (LibreOffice→ Специальные возможности).

4.5 Qt

- **qt5-doc**
- qt5-doc-html
- qtcreator
- qtcreator-doc
- qttools5-dev
- **qttools5-dev-tools**
- qttranslations5-l10n
- **qt5-default**

4.6 GTK+

4.6.1 Тема

Для профессиональной работы рекомендуется использовать тему, создающую наименьшее напряжение на глаза. К таким темам относятся тёмные. Для установки тёмной темы для конкретного пользователя достаточно создать файл с настройками GTK+ 3:

```
mkdir ~/.config/gtk-3.0/  
vim ~/.config/gtk-3.0/settings.ini
```

Потребуются следующие установки:

```
[Settings]  
gtk-theme-name = Adwaita  
gtk-application-prefer-dark-theme = true
```

Установка тёмной темы для `gnome-terminal` осуществляется вручную: Правка → Параметры.

4.7 GIMP

Для эффективной работы бывает удобнее использовать однооконный режим, как это делается во многих средах:

Окна > Однооконный режим