

Рекомендации по установке и настройке Astra Linux 1.5

Белугин Артём
Панченко Александр
Бельков Константин

Появление документа: 03.08.2017
Последняя актуализация: 26.11.2019

Содержание

- 1 Графическая установка
- 2 Установка пароля для пользователя root
- 3 Создание пользователей и групп
- 4 Настройка сетевых интерфейсов и их автостарт
- 5 Создание и использование локального репозитория
- 6 Использование сетевого репозитория
- 7 Настройка grub
- 8 Библиотеки
- 9 Запись дисков
- 10 Настройка vsftpd
- 11 Настройка ssh
- 12 Отключение блокировки пользователей после неудачных попыток логина
- 13 Включение общепринятых alias'ов
- 14 Установка драйверов Nvidia
- 15 Установка rpm-пакетов
 - 15.1 Установка 32-х битных rpm-пакетов
 - 15.2 Пример установки драйверов для Lscan
 - 15.3 Корректировка старт-стоп скриптов
- 16 Тонкая настройка mc
- 17 Создание ярлыков на рабочем столе

Графическая установка

В ходе графической установки обязательно нужно создать пользователя, присвоить ему пароль, а также задать пароль на grub. Рекомендуется создавать фиктивного пользователя user с паролем useruser и таким же паролем на grub.

В выборе пакетов, доступных для установки, смело выбирайте всё (СУБД, Графические пакеты и т.д.). А вот режим Киоска или тачскрин монитор включать не стоит. Не стоит ставить галочки и на отключение автоматического определения сети и так далее.

Установка пароля для пользователя root

Нигде во время установки не предлагается установить пароль для пользователя "root". Чтобы это

сделать после установки, следует залогиниться пользователем, созданным во время установки, а затем выполнить следующую команду для получения root-терминала и смены пароля:

```
sudo su -  
passwd root
```

Создание пользователей и групп

Перед созданием группы "afis" с GID 1001, нужно изменить GID группы astra-admin на 1101 в */etc/group*:

```
Было: astra-admin:x:1001:user
```

```
Стало: astra-admin:x:1101:user
```

Далее создать группу afis как обычно:

```
groupadd -g 1001 afis
```

Перед созданием пользователей "admin" и "st" с UUID 1000 и 1001 соответственно, нужно удалить фиктивного пользователя user:

```
userdel -r user
```

Далее создать пользователей "admin" и "st" с обязательным использованием ключа "-m", иначе домашняя директория пользователя создана не будет:

```
useradd -g afis -u 1000 -G disk, tty, uucp, cdrom, floppy, audio, dip, video, plugdev, fuse, scanner, netdev, bluetooth -m admin  
useradd -g afis -u 1001 -G disk, tty, uucp, cdrom, floppy, audio, dip, video, plugdev, fuse, scanner, netdev, bluetooth -m st
```

Следует также выполнить особую команду по назначению дефолтных уровней безопасности. Без них часть функций в Астре заблокировано:

```
usermac -z st  
usermac -z admin
```

Не забудьте назначить им пароли:

```
passwd admin  
passwd st
```

Настройка сетевых интерфейсов и их автостарт

Замечено, что если использовать стандартный апплет настройки сети (в правом нижнем углу после логина в систему), то сеть не стартует до тех пор, пока пользователь не залогинится. Что может быть губительным для серверов, где никто никогда не логинится. Можно конечно включить автологин, но это небезопасно. Поэтому есть следующее решение:

1. Отключаем запуск этого апплета.

Открываем его главное окно.

Настройки

На вкладке "Общие настройки" поставьте галку "Всегда показывать проводной интерфейс", там же снимите галку "Автоматически переключаться при потере сетевого соединения".

На вкладке "Внешние программы" снимите галку "Включить автозапуск"

2. Настраиваем сеть в стандартных для Астры конфигурационных файлах.

Отредактируйте файл `/etc/network/interfaces`, добавив в его конец следующие строки:

```
auto eth0
iface eth0 inet static
    address 192.168.37.251
    netmask 255.255.252.0
    broadcast 192.168.39.255
    gateway 192.168.36.1
```

Обратите внимание, поднятие интерфейса "eth0" (как и любых других) должно быть после поднятия интерфейса "lo". Иначе инициализация сети занимает очень продолжительное время в момент загрузки.

Отредактируйте файл `/etc/resolv.conf`, добавив в него следующие строки (данный файл может отсутствовать, создайте его):

```
domain papillon.ru
search papillon.ru
nameserver 192.168.36.3
nameserver 192.168.36.4
```

Изменения вступят в силу после перезагрузки

```
reboot
```

Создание и использование локального репозитория

Создаем каталог репозитория, например `./1/astra`:

```
mkdir ./1/astra
```

В нем создаем каталог `conf`:

```
mkdir ./1/astra/conf
```

В `conf` создаем файл `distributions` с таким содержимым:

```
Codename: smolensk
Suite: stable
Version: 1.5
Origin: Astra
Label: Astra 1.5
Description: Astra Stable Repository
Architectures: amd64
Components: main contrib non-free
DebIndices: Packages Release . .gz .bz2
```

```
DscIndices: Sources Release . .gz .bz2
Contents: . .gz .bz2
```

Затем с первого диска дистрибутива ставим пакет репрепро:

```
mount /dev/cdrom /mnt/cdrom
dpkg -i /mnt/cdrom/pool/main/r/reprepro/reprepro_4.12.5-1_amd64.deb
```

После чего инициализируем репозиторий:

```
cd /.1/astra
reprepro export
reprepro createsymlinks
```

Затем монтируем первый диск дистрибутива в /mnt/cdrom и копируем пакеты в репозиторий:

```
reprepro -b /.1/astra/ includedeb smolensk /mnt/cdrom/pool/*/*/*.deb
```

По завершении отмонтируем первый диск и монтируем второй, затем также копируем все файлы:

```
reprepro -b /.1/astra/ includedeb smolensk /mnt/cdrom/pool/*/*/*.deb
```

Репозиторий готов.

Чтобы подключить его, нужно добавить в /etc/apt/sources.list (предварительно выкопировав все, что там было):

```
deb file:///1/astra smolensk contrib main non-free
```

Затем нужно выполнить:

```
apt-get update
```

Теперь можно ставить пакеты из репозитория.

Использование сетевого репозитория

Для использования сетевого репозитория (можно использовать созданный в предыдущем пункте репозиторий) нужно добавить в /etc/apt/sources.list (предварительно выкопировав все, что там было):

```
deb ftp://linux.papillon.ru/astra15 smolensk contrib main non-free
```

(если репозиторий на ftp). Затем нужно выполнить:

```
apt-get update
```

Теперь можно ставить пакеты из репозитория.

Настройка grub

Отредактируйте файл /boot/grub/grub.cfg

Найдите строку с фразой `/boot/vmlinuz-4.2.0-23-generic` - именно это ядро грузится по-умолчанию. В конце этой строки уберите параметры `quiet splash`. Это позволит видеть процесс загрузки системы.

Что именно загружается по-умолчанию указывается параметрами `set default` (19я строка файла), который указывает на уникальный идентификатор со строкой `menuentry`, например:

```
set default="gnulinux-4.2.0-23-generic-advanced-794b6382-124f-4a1f-9bc8-87cb9c81e4e0"
```

указывает на строку

```
menuentry 'AstraLinuxSE GNU/Linux, with Linux 4.2.0-23-generic' --class astralinuxse --class gnu-linux --class gnu --class os
```

Библиотеки

Живой сканер (64 бит) требует для своей работы `libtiff.so.3`, `libjpeg.so.62`, `libcurl.so.4`. Нужно сделать линк и установить пакет:

```
ln -s /usr/lib/x86_64-linux-gnu/libtiff.so.5.2.0 /usr/lib/x86_64-linux-gnu/libtiff.so.3
wget http://it.papillon.ru/astra15/libjpeg62_6b1-3+deb7u1_amd64.deb
dpkg -i libjpeg62_6b1-3+deb7u1_amd64.deb
aptitude install libcurl3
```

Запись дисков

В Астре нет привычных для ЖС команд `cdrecord` и `mkisofs`. Поэтому следует создать линки на их аналоги:

```
ln -s /usr/bin/wodim /usr/bin/cdrecord
ln -s /usr/bin/genisoimage /usr/bin/mkisofs
```

Настройка vsftpd

Для нормальной работы FTP-сервера `vsftpd` нужно отредактировать конфигурационный файл `/etc/vsftpd.conf`:

Заменить

```
listen=NO
```

на

```
listen=YES
```

Заменить

```
listen_ipv6=YES
```

на

```
listen_ipv6=NO
```

Раскомментировать

```
local_enable=YES  
write_enable=YES  
xferlog_file=/var/log/vsftpd.log
```

Добавить в автозагрузку

```
chkconfig vsftpd on
```

Запустить прямо сейчас

```
/etc/init.d/vsftpd restart
```

Примечание: vsftpd иногда плохо реагирует на команды vsftpd stop/restart. Контролируйте изменение PID'a процесса через ps ax.

Настройка ssh

По-умолчанию, после входа через ssh запускать графические приложения не получится. Чтобы разрешить это, нужно проверить наличие установленного пакета *xauth*:

```
dpkg -l | grep xauth
```

Если его нет, то установить командой

```
apt-get install xauth
```

Затем отредактировать файл */etc/ssh/sshd_config*, выставив в нём следующие переменные:

```
X11Forwarding yes  
X11DisplayOffset 10  
X11UseLocalhost no
```

После чего перезапустить ssh:

```
/etc/init.d/ssh restart
```

Заходить следует с ключом *-X*:

```
ssh -X username@hostname
```

Отключение блокировки пользователей после неудачных попыток логина

В Астре по-умолчанию включена блокировка логина пользователя после 10 неудачных попыток логина. Чтобы избежать этого следует отредактировать файл `/etc/pam.d/common-auth`, закомментировав следующую строку (строка 17):

```
auth [success=ignore default=die] pam_tally.so per_user deny=10
```

Включение общепринятых alias'ов

По-умолчанию, у пользователей выключены общепринятые alias'ы, такие как ll. Для их активации рекомендуется отредактировать файл `.bashrc`, который находится в домашнем каталоге пользователя.

Для пользователя root отредактируйте файл `/root/.bashrc`
Раскомментируйте в нём следующие строки:

```
alias ls='ls $LS_OPTIONS'  
alias ll='ls $LS_OPTIONS -l'  
alias l='ls $LS_OPTIONS -lA'
```

Для всех остальных пользователей отредактируйте файл `/home/<имя_пользователя>/.bashrc`
Раскомментируйте в нём следующие строки:

```
alias ll='ls -l'  
alias la='ls -lA'  
alias l='ls -lCF'
```

Установка драйверов Nvidia

Разработчик операционной системы советует использовать готовые deb-пакеты:

GPU серий GeForce 8xxx – GeForce 8xx

<http://it.papillon.ru/astra1.5/nvidia-drivers-astra-smolensk-1.5-340.98-signed.deb>

GPU серий GeForce 4xx – GeForce 9xx

http://it.papillon.ru/astra1.5/nvidia-drivers-astra-smolensk-1.5-361.28_signed.deb

GPU серий GeForce 4xx – GeForce 9xx, GeForce 10 TiTAN X (Pascal)

http://it.papillon.ru/astra1.5/nvidia-drivers-astra-smolensk-1.5-367.35_signed.deb

GPU серий GeForce 4xx – GeForce 9xx, TiTAN X, GeForce GTX 1050, GeForce GTX 1060

<http://it.papillon.ru/astra1.5/nvidia-drivers-astra-smolensk-1.5-375.20-signed.deb>

Скачайте подходящий для Вашей видеокарты пакет и установите его при помощи команды

```
dpkg -i <имя_пакета.deb>
```

Установка rpm-пакетов

В некоторых случаях возможна ситуация когда у вас есть только rpm-пакет, но нет deb-пакета

(например, rpm-драйвера для живых сканеров). Чтобы поставить такой пакет, необходимо установить утилиту *alien*:

```
aptitude install debhelper rpm
wget http://it.papillon.ru/astra1.5/alien_8.92_all.deb
dpkg -i alien_8.92_all.deb
```

После этого следует преобразовать rpm-пакет в deb-пакет, выполнив команду:

```
alien --to-deb /path/to/file.rpm
```

Полученный deb-файл можно установить командой:

```
dpkg -i file.deb
```

Обратите внимание, такой способ установки далеко не всегда гарантирует правильность установки и работоспособности rpm-пакета.

Установка 32-х битных rpm-пакетов

Иногда необходимо преобразовать 32хбитный rpm-пакет:

```
alien --to-deb xxkb-1.10-c4e15.i386.rpm
```

И можно увидеть следующее сообщение

```
xxkb-1.10-c4e15.i386.rpm is for architecture i386 ; the package cannot be built on this system
```

В этом случае следует выполнить следующие шаги:

```
alien -g xxkb-1.10-c4e15.i386.rpm
cd xxkb-1.10
mcedit debian/control
```

Найдите строку *Architecture: i386* и добавьте в неё *amd64* через запятую

```
Было: Architecture: i386
```

```
Стало: Architecture: i386, amd64
```

Выполните сборку пакета:

```
./debian/rules binary
```

Установите пакет

```
dpkg -i ../xxkb_1.10-1_amd64.deb
```

Пример установки драйверов для Lscan

Например, драйвер для DS45 установится, но работать не будет. Нужно сделать следующие действия:

Убедитесь, что у вас установлен *gcc* с сопутствующим обвесом. Выполните команду *dpkg -l* и проверьте, что установлены следующие 5 пакетов:

```
root@belugintmp-251:~# dpkg -l | grep gcc
ii gcc                4:4.7.0-6astra1.se1      amd64      GNU C compiler
ii gcc-4.7            4.7.2-5astra.se1        amd64      GNU C compiler
ii gcc-4.7-base:amd64 4.7.2-5astra.se1        amd64      GCC, the GNU Compiler Collection (base pac
ii lib32gcc1          1:4.7.2-5astra.se1      amd64      GCC support library (32 bit Version)
ii libgcc1:amd64     1:4.7.2-5astra.se1      amd64      GCC support library
```

Убедитесь что у вас установлен "linux-headers-4.2.0-23-generic" (либо его аналог если используется нестандартное ядро). Выполните команду "dpkg -l" и проверьте, что установлен следующий пакет:

```
root@belugintmp-251:~# dpkg -l | grep linux-headers-4.2.0-23-generic
ii linux-headers-4.2.0-23-generic 4.2.0-23.28astra39      amd64      Linux kernel headers for version 4.2.0 on
```

Если какого-либо пакета нет, то доустановите его при помощи команды:

```
aptitude install <packagename>
```

После этого можно переходить к установке драйверов:

```
dpkg -i ds45-usb_4.4.8.06-1.1_all.deb lcd-usb_4.4.8.06-1.1_all.deb papsound_4.4.8.06-2.04_all.deb
cd /usr/src/ds45_usb-4.4.8.06
make
cp ds45_usb.ko /lib/modules/4.2.0-23-generic/kernel/drivers/char

cd /usr/src/lcd_usb-4.4.8.06
make
cp lcd_usb.ko /lib/modules/4.2.0-23-generic/kernel/drivers/char

cd /usr/src/papsound-4.4.8.06
make
cp papsound.ko /lib/modules/4.2.0-23-generic/kernel/drivers/char

depmod -a

/etc/init.d/ds45_usb start
/etc/init.d/lcd_usb start
/etc/init.d/papsound start

chkconfig ds45_usb on
chkconfig lcd_usb on
chkconfig papsound on
```

Корректировка старт-стоп скриптов

Команда *chkconfig* из примера выше может вызвать ошибку из-за того, что штатные старт-стоп скрипты пока не рассчитаны на использование в Астра 1.5

Чтобы это исправить нужно отредактировать старт-стоп скрипт, изменив в нём шапку на (пример для DS45):

```
### BEGIN INIT INFO
```

```
!# Provides:      ds45_usb
!# Required-Start: $all
!# Required-Stop:
!# Default-Start: 2 3 4 5
!# Default-Stop:  0 1 6
!# Short-Description: Papillon DS45 driver initscript
!# Description:    Papillon DS45 driver initscript
!### END INIT INFO
```

Таким образом (разумеется изменяя значения в шапке на соответствующие) следует поступить с каждым скриптом, на который ругается система

Тонкая настройка mc

Если вы планируете использовать mc, рекомендуется в конец файла */etc/bash.bashrc* добавить следующие строки:

```
!if [ -f /usr/lib/mc/mc.sh ]; then
!  . /usr/lib/mc/mc.sh
!fi
```

Это поможет при выходе из mc оставаться в той директории, где был произведен выход, а не переходить в директорию, откуда была дана команда на запуск mc.

Создание ярлыков на рабочем столе

Вариант 1 - через графику

ПКМ на рабочем столе -> Создать -> Ярлык.

Заполните поля, отмеченные жирным шрифтом (Имя и Команда).

Вариант 2 - созданием файла в домашнем каталоге пользователя:

```
!~/home/st/.fly/startmenu/recent/shortcut0.desktop
```

Со следующим содержимым:

```
! [Desktop Entry]
! Name = Живой сканер
! Type = Application
! NoDisplay = false
! Exec = /home/lscanner/bin/q9.lscan.sh
! Hidden = false
! Terminal = false
! StartupNotify = false
```

Обратите внимание на имя файла - в нем есть порядковый номер, который следует менять.