

Создание сервера терминалов LTSP на Astra Linux CE ОС ОН Орел 2.12

Сейчас мы попробуем создать сервер терминалов LTSP на базе Astra Linux Common Edition ОС ОН Орел с загрузочным образом для бездисковых станций.

- [Режимы работы](#)
- [Настройка сервера](#)
 - [Настройка сети](#)
 - [Установка основных компонентов](#)
 - [Настройка DHCP сервера](#)
 - [Настройка tftpd сервера](#)
- [Создание образа тонкого клиента](#)
 - [Настройка образа](#)
- [Создание образа толстого клиента](#)
 - [Настройка образа](#)



Данная статья применима к:

- ОС ОН Орёл 2.12

Режимы работы

Существует два режима работы клиентов:

- тонкий клиент - По сети загружается минимальный образ с базовой системой и графическим сервером. Все приложения выполняются на сервере терминалов, результаты работы передаются по сети графическому серверу на клиенте.
- толстый клиент - Загружается полноценный образ с ОС. Приложения выполняются непосредственно на клиенте, а сервер просто предоставляет доступ к пользовательским файлам и программам.

Настройка сервера

Настройка сети

В примере на сервере используется 2 сетевых интерфейса:

- eth0 - смотрит в интернет.
- eth1 - во внутреннюю сеть, в которой находятся тонкие клиенты.

Сеть на сервере настроена следующим образом:

```
eth0 - DHCP
eth1 - static
ip: 192.168.56.100
mask: 255.255.255.0
```

Включение удаленного доступа

Разрешим удаленный вход в систему:

Панель управления Система Вход в систему Дополнительно - Разрешить удаленный вход в систему

Установка основных компонентов

Предполагается, что репозитории правильно настроены и установка пакетов из сети Интернет работает.

```
apt install isc-dhcp-server ltsp-server tftpd-hpa
```

- [isc-dhcp-server](#) - DHCP сервер. Используется для выдачи клиентам IP-адресов и информации о tftp-сервере и пути к загрузчику pxelinux.
- [tftpd-hpa](#) - сервер tftpd. Отдает по tftp-протоколу загрузчик, ядро и главный конфиг `lts.conf`.
- [ltsp-server](#) - скрипты для создания сервера терминалов

Настройка DHCP сервера

В файле `/etc/default/isc-dhcp-server` в строке `INTERFACЕCv4=` " " указать интерфейс: `eth1`

```
/etc/default/isc-dhcp-server
```

```
INTERFACЕCv4="eth1"
```

Создадим конфигурацию для DHCP сервера:

```
ltsp-config --overwrite isc-dhcp-server
```

```
mv /etc/dhcp/dhcpd.conf /etc/dhcp/dhcpd.bak
```

```
cp /etc/ltsp/dhcpd.conf /etc/dhcp/dhcpd.conf
```

Заменяем адреса в `dhcpd.conf` на нужные:

```
#
# Default LTSP dhcpd.conf config file.
#

authoritative;

subnet 192.168.56.0 netmask 255.255.255.0 {
    range 192.168.56.20 192.168.56.250;
    option domain-name "example.com";
    option broadcast-address 192.168.56.255;
    option routers 192.168.56.100;
    next-server 192.168.56.100;
    # get-lease-hostnames true;
    option subnet-mask 255.255.255.0;
    option root-path "/opt/ltsp/amd64";
    if substring( option vendor-class-identifier, 0, 9 ) = "PXEClient" {
        filename "/ltsp/amd64/pxelinux.0";
    } else {
        filename "/ltsp/amd64/nbi.img";
    }
}
```

Перезапустим dhcp:

```
systemctl restart isc-dhcp-server
```

Настройка tftpd сервера

В файле `/etc/default/tftpd-hpa` изменим путь к корню тфп сервера:

```
/etc/default/tftpd-hpa
```

```
TFTP_DIRECTORY="/var/lib/tftpboot"
```

Перезапустим:

```
systemctl restart tftpd-hpa
```

Создание образа тонкого клиента

Скопируем конфигурацию Debian (т.к. для орла конфигурацию еще не придумали)

```
cp -r /usr/share/ltsp/plugins/ltsp-build-client/Debian /usr/share/ltsp/  
plugins/ltsp-build-client/AstraLinuxCE
```

Директория должна называться именно AstraLinuxCE

Поправим файл:

```
/usr/share/ltsp/plugins/ltsp-build-client/AstraLinuxCE/010-debootstrap
```

Строку

```
LC_ALL=C ${DEBOOTSTRAP:-"debootstrap"} $DEBOOTSTRAPOPTS --arch $ARCH $DIST $ROOT $MIRROR
```

меняем на

```
LC_ALL=C ${DEBOOTSTRAP:-"debootstrap"} --components "main, contrib, non-free" $DEBOOTSTRAPOPTS --arch $ARCH  
$DIST $ROOT $MIRROR
```

Соберем образ:

```
ltsp-build-client --dist orel --purge-chroot --mirror "http://dl.astralinux.  
ru/astra/current/orel/repository" --components "contrib, main, non-free" --  
kernel-packages linux-image-generic --early-packages vim
```

Разберем опции:

--dist orel - собрать образ с системой "Орел"

--purge-chroot - очищать директорию с файлами образа

--mirror "<http://dl.astralinux.ru/astra/current/orel/repository>" - указываем репозиторий, откуда ставить систему.

--components "contrib, main, non-free" - используем все компоненты

--kernel-packages linux-image-generic - установить ядро generic по-умолчанию

--early-packages vim - установить редактор vim на стадии сборки

Все опции обязательны, иначе при сборки возникнут ошибки (хотя можно поправить сборочные скрипты в `/usr/share/ltsp/plugins/ltsp-build-client/AstraLinuxCE`)

По умолчанию образ собирается в `/opt/ltsp/amd64`

Настройка образа

```
ltsp-chroot apt update
```

```
ltsp-chroot apt install ltsp-client-core
```

зададим пароль для root:

```
ltsp-chroot passwd root
```

В `/opt/ltsp/amd64/etc/ltsp/update-kernels.conf` укажем тип загружаемого образа: NBD

```
/opt/ltsp/amd64/etc/ltsp/update-kernels.conf
```

```
# Supported booth methods  
BOOT_METHODS="NBD"
```

Выполним:

```
ltsp-chroot bash /usr/share/ltsp/update-kernels
```

В `/opt/ltsp/amd64/etc/initramfs-tools/initramfs.conf` указать использовать `busybox`:

```
/opt/ltsp/amd64/etc/initramfs-tools/initramfs.conf
```

```
BUSYBOX=y
```

Установим `busybox`:

```
ltsp-chroot apt install busybox
```

затем пересоберем `initramfs`:

```
ltsp-chroot update-initramfs -uk all
```

Установим `fly-dm` (без него показывает черный экран) и `xserver`:

```
ltsp-chroot apt install xserver-xorg fly-dm
```

Запретим автозапуск `fly-dm`:

```
ltsp-chroot systemctl disable fly-dm
```

Обновим собранный образ:

```
ltsp-update-image
```

И перезапустим `nbd-server`:

```
systemctl restart nbd-server
```

Создаем пользователей на сервере терминалов, включаем тонкий клиент и выполняем загрузку по сети.

Создание образа толстого клиента

Установка дополнительных компонентов

Для корректной работы образа потребуется установка дополнительных компонентов на сервер терминалов:

```
wget https://mirror.yandex.ru/debian/pool/main/l/ldm/ldm-server_2.2.18-2_all.deb
```

```
wget https://mirror.yandex.ru/debian/pool/main/l/ltspfs/ltspfs_1.4-2%  
2Bb1_amd64.deb
```

```
dpkg -i ldm-server_2.2.18-2_all.deb
```

```
dpkg -i ltspfs_1.4-2%2Bb1_amd64.deb

apt install -f

apt install openssh-server squashfs-tools x11-utils xauth
```

Собираем образ: [Создание образа](#)

Настройка образа

```
ltsp-chroot apt update
```

```
ltsp-chroot apt install ltsp-client-core wget alsa-utils cron cups-bsd
libasound2-plugins netcat ntpdate pulseaudio python-serial sane-utils sshfs
x11-xserver-utils xorg xserver-xorg busybox
```

зададим пароль для root:

```
ltsp-chroot passwd root
```

Загрузим доп компоненты

```
ltsp-chroot

cd /root

wget https://mirror.yandex.ru/debian/pool/main/l/ldm/ldm_2.2.18-2_amd64.deb

wget https://mirror.yandex.ru/debian/pool/main/l/ltspfs/ltspfsd_1.4-2_all.
deb

wget https://mirror.yandex.ru/debian/pool/main/l/ltspfs/ltspfsd-core_1.4-2%
2Bb1_amd64.deb

dpkg -i *.deb

apt install -f

exit
```

Установим основные пакеты Fly

```
ltsp-chroot apt install fly-all-main
```

Так же можно установить метапакеты (или же необходимые пакеты по отдельности)

- fly-all-optional - ark, clementine, compton, fly-admin-iso, fly-fm-crypt, gparted, guvcview, kggpg, libsvg2-bin, qapt-deb-installer, qbat, recoll, speedcrunch, synaptic, vlc, vlc-astra, vlc-l10n, vlc-plugin-access-extra, xca, yelp
- fly-all-games - klines, kmines, kpat, supertux, warzone2100, xonotic
- libreoffice - кроссплатформенный, свободно распространяемый офисный пакет с открытым исходным кодом

Запретим автозапуск fly-dm:

```
ltsp-chroot systemctl disable fly-dm
```

В `/var/lib/ftptboot/ltsp/amd64/lts.conf` меняем значение сессии на default, и добавим параметр `LDM_PASSWORD_HASH` для корректной работы блокировщика экрана:

```
# If you're using Unity, you might want to install gnome-session-fallback and
# use it by default instead, as it performs better on LTSP thin clients.
LDM_SESSION="default"
LDM_PASSWORD_HASH=True
```

В `/opt/ltsp/amd64/etc/ltsp/update-kernels.conf` укажем тип загружаемого образа: NBD

```
/opt/ltsp/amd64/etc/ltsp/update-kernels.conf
```

```
# Supported booth methods
BOOT_METHODS="NBD"
```

Выполним:

```
ltsp-chroot bash /usr/share/ltsp/update-kernels
```

В `/opt/ltsp/amd64/etc/initramfs-tools/initramfs.conf` указать использовать busybox:

```
/opt/ltsp/amd64/etc/initramfs-tools/initramfs.conf
```

```
BUSYBOX=y
```

затем пересоберем initramfs:

```
ltsp-chroot update-initramfs -uk all
```

Обновим собранный образ:

```
ltsp-update-kernels && ltsp-update-image
```

Создаем пользователей на сервере терминалов, включаем тонкий клиент и выполняем загрузку по сети.