# Лекция 6. Основы пользовательской работы в ОССН Astra Linux

Оглавление

[Лекция 6. Основы пользовательской работы в ОССН Astra Linux 1](#_Toc495263993)

[1. Варианты загрузки и экран регистрации в ОССН 1](#_Toc495263994)

[2. Администрирование параметров графического входа в систему 8](#_Toc495263995)

[3. Основные приёмы работы с защищённой графической подсистемой Fly. 9](#_Toc495263996)

[4. Завершение пользовательского сеанса и завершение работы 16](#_Toc495263997)

## 1. Варианты загрузки и экран регистрации в ОССН

После включения питания компьютера запускается процедура самотестирования (Power On Self Test, POST), проверяющая основные компоненты системы: видеокарту, оперативную память, жесткие диски и т. д. Затем начинается загрузка операционной системы. Компьютер ищет на жестком диске (и других носителях) программу-загрузчик операционной системы. Если такая программа найдена, то ей передается управление, если же такая программа не найдена ни на одном из носителей, выдается сообщение с просьбой вставить загрузочный диск.

Задача загрузчика — предоставить пользователю возможность выбрать нужную операционную систему (ведь кроме Linux на компьютере может стоять и другая операционная система) и передать ей управление. В случае с Linux загрузчик загружает ядро операционной системы и передает управление ему. Все последующие действия по загрузке системы (монтирование корневой файловой системы, запуск программы инициализации) выполняет ядро Linux.

В настоящее время популярны два загрузчика Linux: LILO и GRUB. GRUB является более современным и используется по умолчанию в большинстве дистрибутивов. Так что после установки Linux начальным загрузчиком будет именно GRUB (если вы самостоятельно не выберете другой загрузчик). Некоторые дистрибутивы имеют собственные загрузчики — например, ASPLinux использует загрузчик ASPLoader.

Кроме обычного GRUB существует и его более современная версия — GRUB-PC (GRUB2). Особенности этой версии: возможность загружать Linux с раздела ext4 и другой, более гибкий, файл конфигурации.

При включении питания компьютера с установленной на нем ОССН Astra Linux после этапа процедуры POST системы BIOS/EFI отображается экран начального загрузчика GRUB2.

Рисунок . Экран загрузчика GNU GRUB OCCH Astra Linux.

По умолчанию ОССН может быть загружена в двух основных режимах, каждый из которых может быть осуществлён в двух вариантах (рис. 1):

1. Режим **РаХ**: загрузка ядра ОССН с включённым модулем **РаХ**, обеспечивающим возможность настройки прав доступа приложений к страницам памяти.

2. Режим Generic: загрузка ядра ОССН без модуля **РаХ**.

В компьютерной безопасности **PaX** (произн. «Пакс») — это патч к ядру Linux, который предоставляет возможность настроить минимальные права доступа приложений к страницам памяти. Таким образом обеспечивается достаточно тонкая настройка, с помощью которой программам разрешается выполнять только те действия, которые необходимы, исходя из предоставляемой ими функциональности, но не более того. PaX был впервые выпущен в 2000 году.

PaX помечает сегмент данных программ в памяти как недоступный для исполнения (так как он по определению не может содержать программных директив, которые необходимо выполнить), а сегмент кода — как не перезаписываемый, и, в придачу, при каждом запросе выделяет память программе из произвольных мест (рандомизация страниц памяти). Эта методика эффективна против применения различных эксплойтов, использующих, например, уязвимость, основанную на переполнении буфера памяти. Такая защита изначально полностью предотвращает прямое выполнение кода из памяти, и одновременно, с прикладной точки зрения, делает, так называемые, return-to-libc (ret2libc) атаки сложными для выполнения (они становятся выполняемыми скорее наудачу, без заранее предсказуемого результата). Однако, вместе с тем, PaX не предотвращает ошибки, приводящие к возможности переопределения переменных и значений указателей.

PaX был написан одноимённой командой разработчиков. Основатель PaX в настоящее время предпочитает оставаться анонимным по неизвестным общественности причинам.

Логотип PaX: пингвин [Такс](https://wiki2.org/ru/Tux) (символ [Linux](https://wiki2.org/ru/Linux)) имеющий [демонический](https://wiki2.org/ru/%D0%94%D0%B5%D0%BC%D0%BE%D0%BD) вид.

В режиме **РаХ** и Generic возможно выбрать вариант загрузки Recovery Mode (режим восстановления), в котором выполняется сценарий загрузки с проверкой ошибок доступа к корневой файловой системе или остановки процесса загрузки из-за ошибок в каком-либо сервисе.

При этом режим Recovery Mode предоставляет администратору ОССН (в данном случае пользователь от имени учётной записи root) интерфейс CLI для запуска утилит диагностики и восстановления (рис. 2).

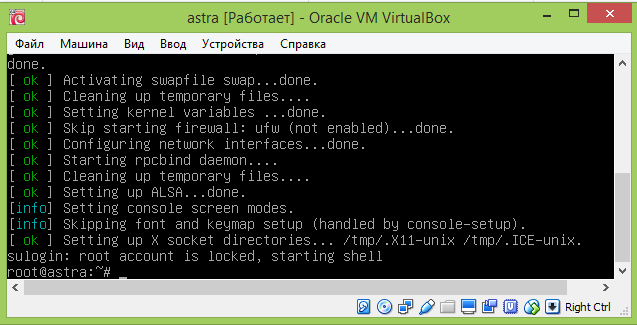


Рисунок 2. CLI-интерфейс OCCH Recovery Mode.

Завершение функционирования в режиме Recovery Mode выполняется командами reboot (перезагрузка) или halt (останов).

После окончания процесса загрузки в режимах РаХ или Generic пользователю предоставляется возможность работы с ОССН в графическом режиме (интерфейс GUI) или в режиме командной строки (интерфейс CLI).

В соответствии со сценарием инициализации системы Debian GNU /Linux, на базе которой разработан дистрибутив ОССН, пользователю по умолчанию предоставляется возможность работы:

* с интерфейсом CLI в шести терминалах (консоли ttyl-tty6);
* с интерфейсом GUI в седьмом терминале (консоль tty7).

Переключение между указанными терминалами осуществляется комбинацией клавиш Ctrl+Alt+FN, где N - номер консоли (устройства tty), в котором пользователь будет регистрировать свою сессию. Например, для перехода в консоль tty1 нужно нажать Ctrl+Alt+F1. Первые шесть виртуальных консолей работают как терминал с приглашением ввода логина и пароля.

Выход из текстового терминала (консоли) осуществляется комбинацией Ctrl + Alt + F7. Консоль tty7 по сути также является виртуальным терминалом, который по умолчанию всегда выделен под графическую среду (Xorg, etc.). Переключение между виртуальными консолями выполняется комбинацией Alt + LeftArrow или Alt + RightArrow.

Вид терминала на примере консоли ttyl представлен на рис. 3.

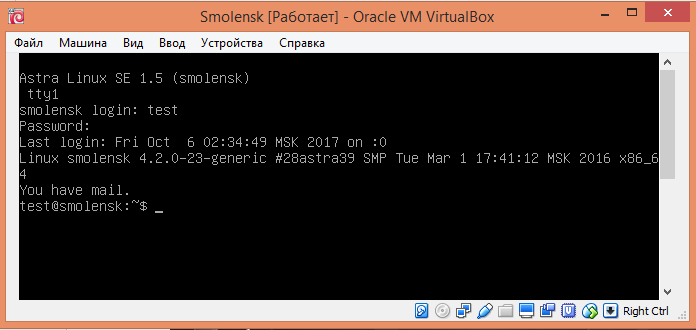


Рисунок . CLI-интерфейс терминала в консоли tty1

Интерфейс командной строки (англ. Command line interface, CLI) — разновидность текстового интерфейса (**CUI**) между человеком и компьютером, в котором инструкции компьютеру даются в основном путём ввода с клавиатуры текстовых строк (команд), в UNIX-системах возможно применение мыши. Также известен под названием консоль.

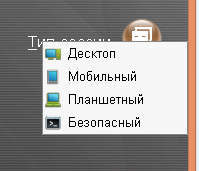
Текстовый пользовательский интерфейс, ТПИ (англ. Text user interface, TUI; также Character User Interface, **CUI**) — разновидность интерфейса пользователя, использующая при вводе-выводе и представлении информации исключительно набор буквенно-цифровых символов и символов псевдографики. Характеризуется малой требовательностью к ресурсам аппаратуры ввода-вывода (в частности, памяти) и высокой скоростью отображения информации. Появился на одном из начальных этапов развития вычислительной техники, при развитии возможностей аппаратуры, нацеленной на реализацию появившегося ранее интерфейса командной строки, который, в свою очередь, является наследником использования телетайпов в качестве интерфейса вычислительной техники. Интерфейс командной строки имеет ряд преимуществ в юзабилити перед графическим интерфейсом, поэтому программы с текстовым интерфейсом создаются и используются по сей день, особенно в специфических сферах и на маломощном оборудовании.

Недостатком подобного типа интерфейса является ограниченность изобразительных средств по причине ограниченности количества символов, включённых в состав шрифта, предоставляемого аппаратурой.

В терминале консоли tty7 по умолчанию загружается GUI-интерфейс, представленный первоначально экраном графического входа в систему. Управляющие элементы этого экрана показаны на рис. 4.

Рассмотрим элементы экрана входа в систему подробнее.

Рисунок . Элементы экрана входа в систему.

**1. Тип сессии**

Этот элемент управления обеспечивает пользователю выбор среды пользовательской сессии ОССН. При этом пользователю предоставляется возможность переключиться между следующими средами пользовательской сессии:

* Десктоп;
* Мобильный;
* Планшетный
* Безопасный

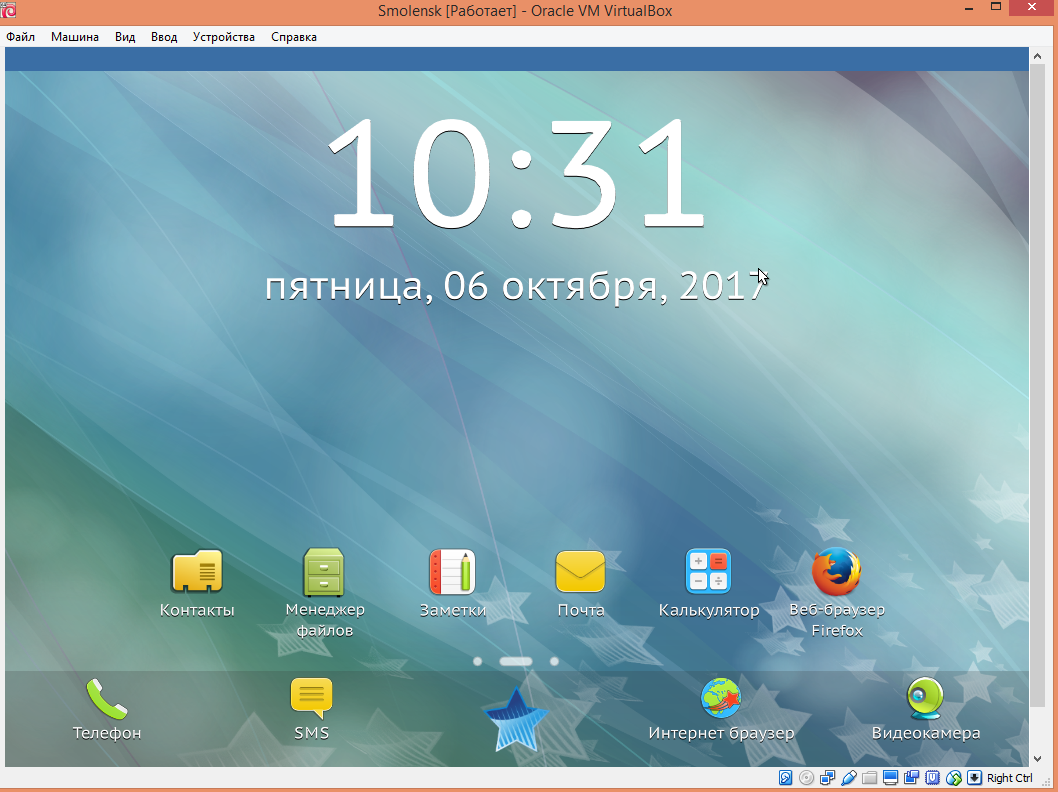


Рисунок . Мобильный графический интерфейс Astra Linux.

Во время самостоятельной подготовки изучите особенности мобильного интерфейса. При отказе работы графической оболочки перейдите в консоль tty1 и выполните команду sudo reboot.

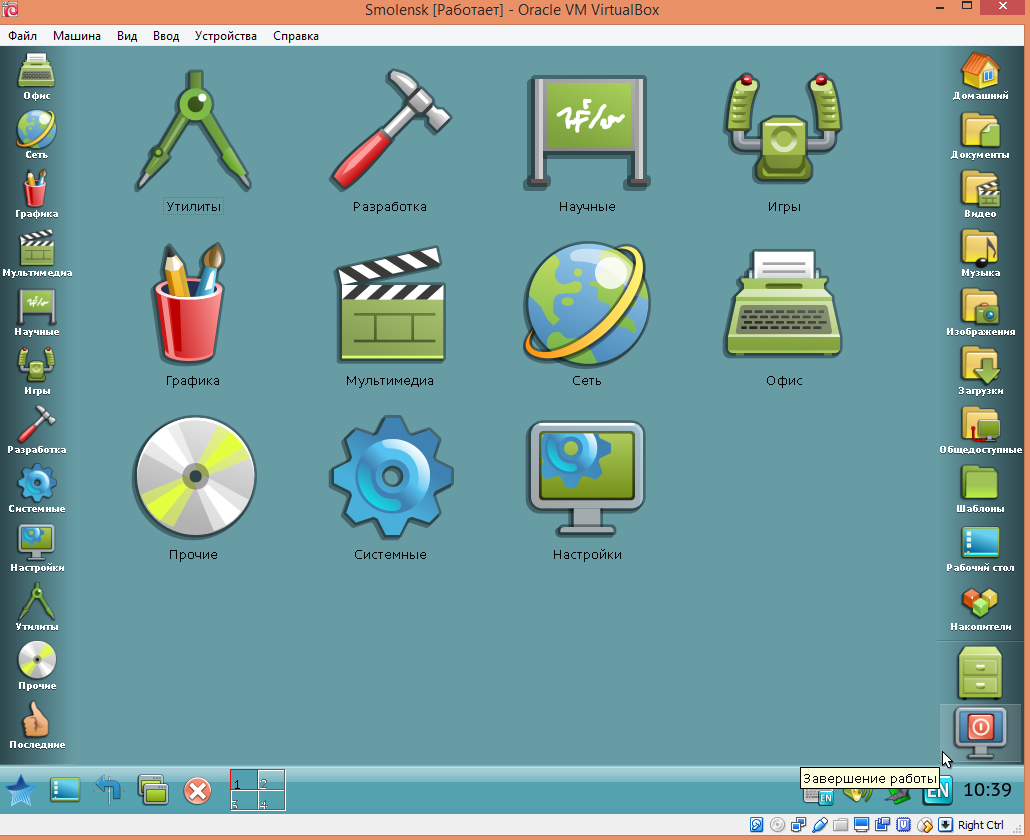
Безопасная сессия предназначена для работы в ОССН с интерфейсом CLI (реализуемым после запуска графического сервера X.Org утилитой fly-term) с использованием консольных команд. Часто используемые консольные команды в утилите fly-term заданы в виде выпадающего списка. Выбранная из него команда переносится в командную строку, после чего достаточно ввести её аргументы и (при необходимости) опции. Также этот список можно изменить с помощью соответствующего элемента меню «Настройка» утилиты fly-term.

Рисунок . Графический интерфейс для планшета.

Выход из безопасной сессии выполняется с помощью элемента «Выход» меню «Файл» утилиты fly-term или командой exit в командной строке. При этом в случае, если открыто две или более вкладки виртуальных терминалов, потребуется подтверждение за­вершения сессии.

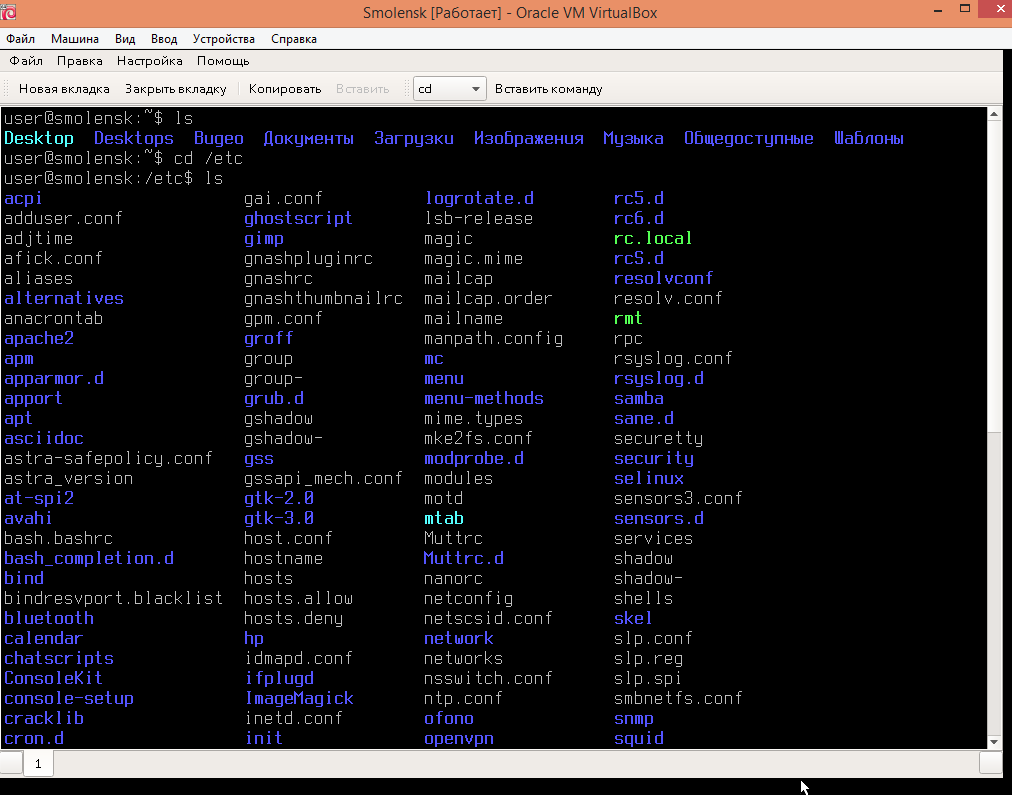
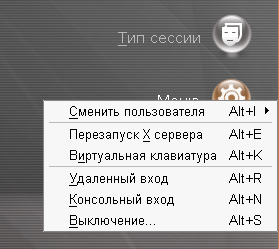


Рисунок . Безопасная сессия с интерфейсом CLI.

**2. Меню**

Меню пользовательского входа в ОССН позволяет реализовать следующие возможности.

Консольный вход (комбинация клавиш Alt+N) — позволяет работать с интерфейсом CLI в терминале консоли ttyl. При выборе этого пункта меню пользователю предварительно отображается окно с сообщением о том, что переключение в консольный режим приведёт к невозможности работы в графическом режиме, который станет возможным снова через 10 секунд после окончания последнего успешного консольного входа или через 40 секунд, если ни один консольный вход не будет осуществлён.

Удаленный вход (комбинация клавиш Alt+R) - позволяет пользователю зарегистрироваться в ОССН другого компьютера (удалённой ОССН) с использованием процесса rlogin. Выбор этого пункта меню отображает окно с перечнем доступных для удалённой регистрации ОССН. При этом имеется возможность добавить новую удалённую ОССН, введя сетевой путь к ней. Также имеется возможность (кнопка «Меню») вызвать меню пользовательского входа для локальной GUI или локальной CLI регистрации в ней.

• Перезапуск X сервера (комбинация клавиш Alt+E) — позволяет передать текущему сеансу X.Org сервера сигнал принудительного перезапуска.

• Вызов виртуальной клавиатуры (комбинация клавиш Alt+K) — позволяет вывести на экран регистрации виртуальную клавиатуру. Применяется в случае невозможности использования аппаратной клавиатуры на традиционных рабочих станциях или же при установке ОССН на устройствах с сенсорным экраном.

• Сменить пользователя (комбинация клавиш Alt+I) — позволяет отобразить список открытых пользовательских сессий с указанием номеров виртуальных терминалов, на которых они выполняются и типов пользовательских сессий (fly, default (по умолчанию), twm или console).

• Выключение (комбинация клавиш Alt+S) — выводит на экран окно завершения работы с ОССН. Используя это окно, пользователь может произвести выключение или перезагрузку ОССН без выполнения входа какого-либо пользователя, а также спланировать их по истечении заданного интервала времени (режим планирования доступен только администратору системы и требует ввода его пароля).

## 2. Администрирование параметров графического входа в систему

Рассмотренные в предыдущем параграфе настройки графического входа в систему заданы по умолчанию. Однако администратор ОССН имеет возможность изменить их с использованием графической утилиты fly-admin-dm. Её запуск можно осуществить двумя способами:

• с использованием элемента «Вход в систему» меню «Настройки» главного меню защищённой графической подсистемы Fly (рис. 8);

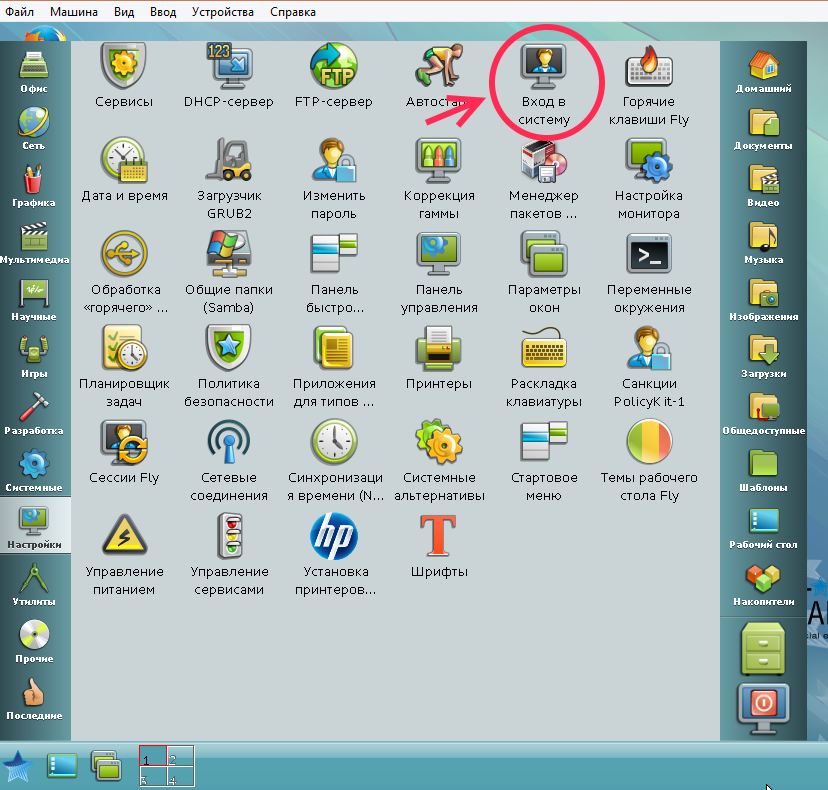


Рисунок . Элемент «Вход в систему» меню «Настройки.

• через ввод команды fly-admin-dm в командной строке утилиты fly-term .

Первые три меню fly-admin-dm («Основное», «Диалог» и «Тема») предназначены для оформления окна графического входа в систему с использованием тем оформления. Вкладки «Выключение», «Пользователи» и «Дополнительно» предназначены для конфигурирования работы окна графического входа в систему.

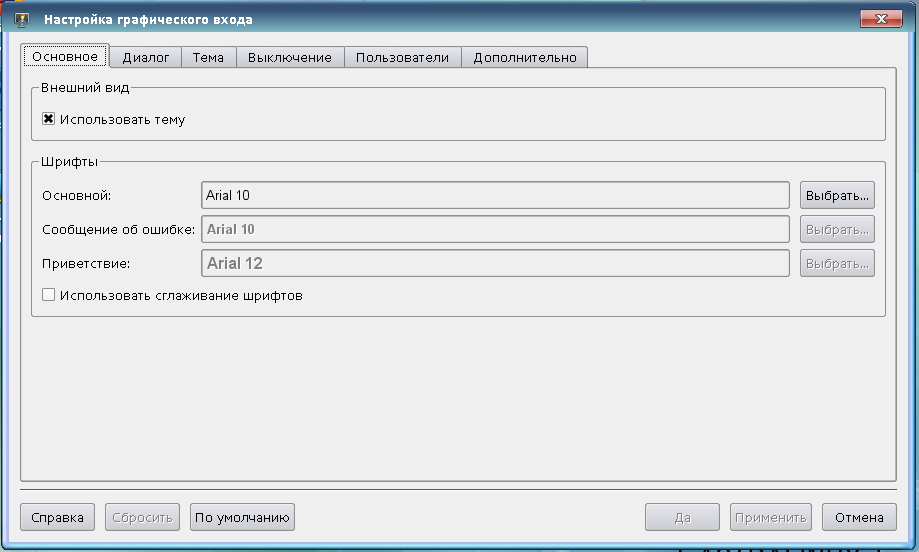


Рисунок . Интерфейс графической утилиты fly-admin-dm.

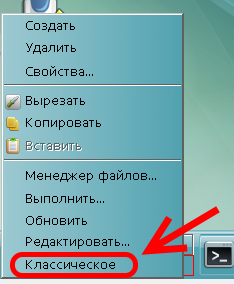
## 3. Основные приёмы работы с защищённой графической подсистемой Fly.

В графической сессии по умолчанию используется защищённая графическая подсистема Fly, при функционировании которой используются:

• сервер X. Org — реализации сервера X Window System с открытым исходным кодом;

• рабочий стол Fly, который, в свою очередь, состоит из менеджера окон Fly Window Manager (утилита flу-wm) и набора fly-утилит для пользователей и администраторов с графическим интерфейсом GUI.

Менеджер окон Fly Window Manager организует работу графической оконной среды ОССН и загружает рабочий стол Fly и его окружение (заданный набор fly-утилит). Интегрированный в среду рабочего стола менеджер рабочих столов обеспечивает поддержку одновременной работы с несколькими рабочими столами (по умолчанию, с четырьмя).

Рабочий стол Fly загружается после регистрации пользователя в графической сессии. Он содержит пространство рабочего стола с фоновым изображением, панель задач и элементы интерфейса пользователя (рис. 10).

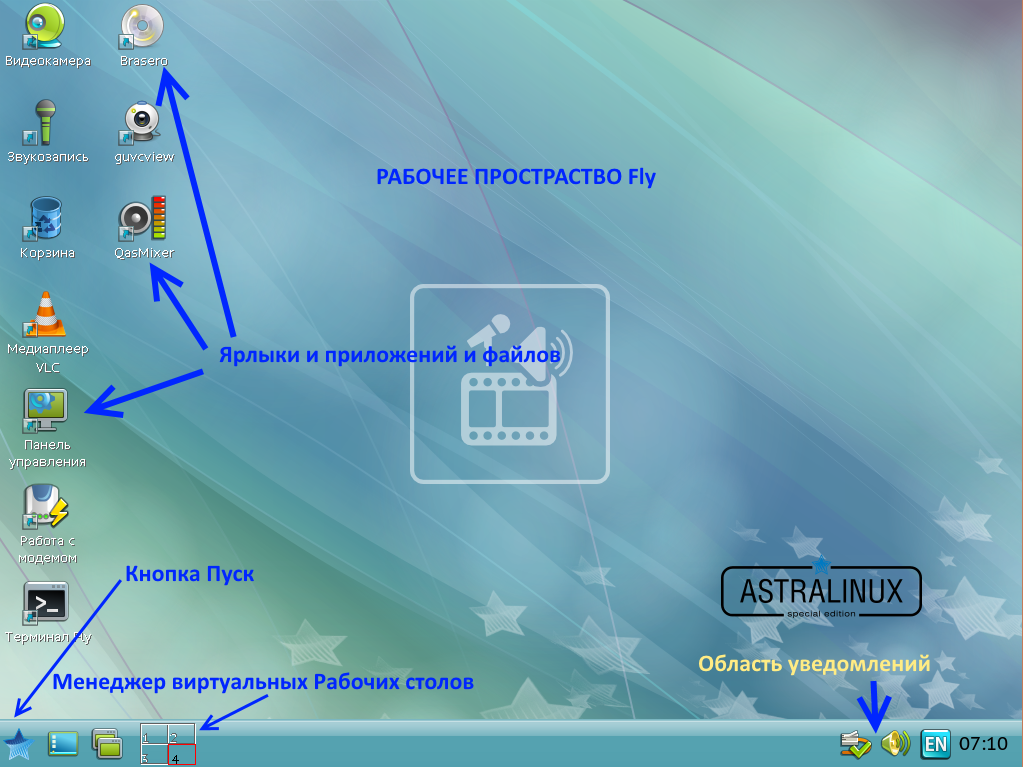
Главное пользовательское меню (вызывается при нажатии на экране кнопки «Пуск») оптимизировано для работы как на традиционных компьютерах, так и на устройствах (планшетах) с сенсорным экраном (используя настройки рабочего стола Fly, его можно преобразовать к классическому виду), и состоит из следующих элементов меню (рис. 11):

• доступные программы и утилиты (левая верхняя область);

• пользовательские каталоги (правая верхняя область);

• режимы работы (левая нижняя область);

• управление файлами (правая нижняя область);

• пользовательские программы и настройки (центральная область).

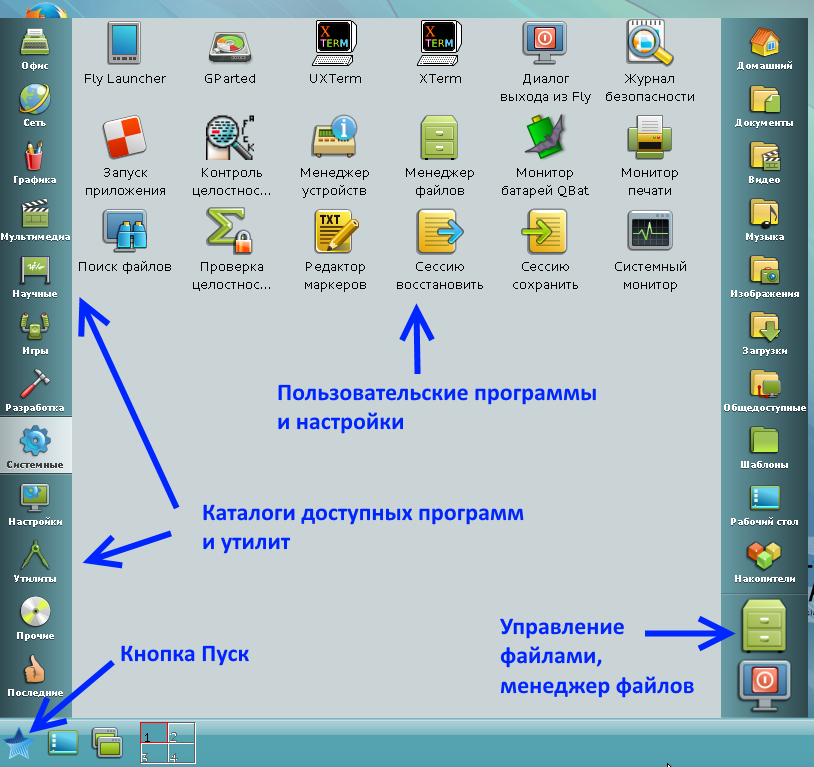
Рисунок 0. Элементы интерфейса рабочего стола Fly.

Рисунок 1. Элементы главного пользовательского меню (кнопка «Пуск»).

Каждый пользователь ОССН имеет возможность индивидуально настроить рабочий стол Fly (внешний вид, расположение элементов, особенности работы с клавиатурой и мышью).

Часть таких настроек жёстко определяется администратором и недоступна непривилегированному пользователю. Доступные пользователю настройки могут быть заданы с использованием утилиты «Панель управления» (fly-admin-center), вызываемой последовательно из основного меню **«Пуск» — «Настройка» — «Панель управления» — «Рабочий стол».** Эта утилита позволяет централизовано использовать некоторые административные и пользовательские утилиты рабочего стола Fly, которые для удобства разделены на несколько категорий. Например, категория «Рабочий стол» объединяет fly-утилиты, большинство из которых может быть применено пользователем для настройки своего индивидуального рабочего стола.

Настройки рабочего стола Fly также могут быть сконфигурированы в режиме, оптимизированном для работы на устройствах с сенсорными экранами. Общим названием таких настроек является режим «Планшет». Для перехода в него используется утилита «Переключатель планшетного режима Fly» (fly-admin-tablet-switch), находящаяся в элементе Настройки» меню «Пуск». Основные визуальные отличия режима «Планшет»:

• приложения запускаются в полноэкранном режиме;

• окна приложений не имеют стандартных для режима «Настольный компьютер» элементов интерфейса (свернуть, развернуть, закрыть окно);

• на панели задач добавлены новые кнопки (например, закрытия активного окна, поворота изображения на 90 градусов, вызова экранной клавиатуры).

Дополнительной настройкой является увеличение иконок приложений, файлов и каталогов, а также областей прокрутки открытого окна. Вид рабочего стола в режиме «Планшет» представлен на рис. 6.

Важной особенностью защищённой графической подсистемы Fly является интегрированная поддержка средств защиты информации (СЗИ) ОССН. По умолчанию пользователю доступны её следующие функции:

• изменение пароля своей учётной записи с помощью консольной команды passwd, утилиты flу-passwd или графической утилиты «Центр обеспечения безопасности»;

• изменение владельца или группы, владеющей объектом файловой системы, созданного пользователем, с помощью консольных команд chown и chgrp, файловых менеджеров Midnight Commander и fly-fm;

• изменение прав доступа в рамках модели minimal ACL (дискреционное управление доступом) к сущности файловой системы, созданной пользователем, с помощью консольной команды chmod, файловых менеджеров Midnight Commander и fly-fm;

• установка уровней доступа (включая неиерархические категории) и целостности при создании новой пользовательской сессии (субъект-сессии);

• получение для текущей пользовательской сессии информации о её уровнях доступа и целостности с помощью консольной команды pdp-id или визуально в области уведомлений на панели задач.

Таким образом, пользователю, для которого администратором ОССН установлены несколько допустимых мандатных уровней доступа или целостности, на экране графического входа в систему (рис. 12) отобразится меню установки их конкретных значений. В дальнейшем выбранные уровни отображаются в виде индикатора в области уведомлений на панели задач.

Для большей наглядности при обработке информации конкретными оконными приложениями (субъект-сессиями) с GUI-интерфейсом значение их уровня доступа дублируется цветовым кодированием:

* «Уровень 0» — голубой;
* «Уровень 1» — желтый;
* «Уровень 2» — оранжевый;
* «Уровень 3» — темно-розовый;
* «Уровень 4» — красный;
* «Уровень 5» — коричневый;
* «Уровень 6» — пурпурный;
* «Уровень 7» — темно-фиолетовый.

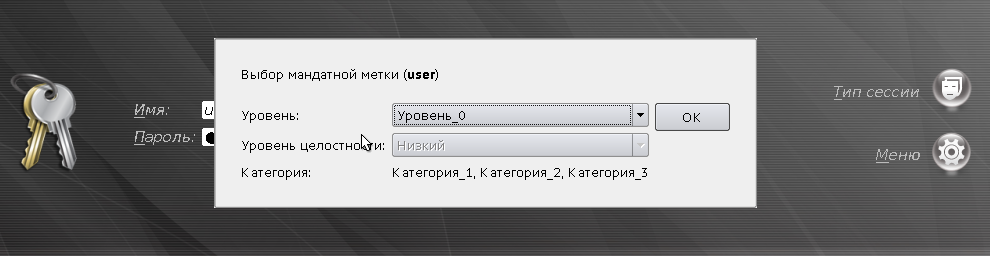


Рисунок . Меню установки уровней доступа и целостности пользовательской.

***Переключимся с основной темы на рассмотрение моделей управления доступа.***

**Базовые модели управления доступом.**

Общим подходом для всех моделей управления доступом является разделение множества сущностей, составляющих систему, на множества объектов и субъектов.

При этом определения понятий «объект» и «субъект» могут существенно различаться. Мы будем подразумевать, что объекты являются некоторыми контейнерами с информацией, а субъекты – пользователи, которые выполняют различные операции над этими объектами.

Безопасность обработки информации обеспечивается путем решения задачи управления доступом субъектов к объектам в соответствии с заданным набором правил и ограничений, которые образуют политику безопасности.

Можно выделить три основные модели управления доступом к объектам: мандатную, дискреционную и ролевую.

**1. Мандатная модель**

Классической мандатной моделью считается модель Белла-ЛаПадулы . Она базируется на правилах секретного документооборота, использующегося в правительственных учреждениях. В этой модели каждому объекту и субъекту (пользователю) системы назначается свой уровень допуска. Все возможные уровни допуска системы четко определены и упорядочены по возрастанию секретности. Действуют два основных правила:

1. Пользователь может читать только объекты с уровнем допуска не выше его собственного.

2. Пользователь может изменять только те объекты, уровень допуска которых не ниже его собственного.

Цель первого правила очевидна каждому, второе может вызвать недоумение. Смысл же его в том, чтобы воспрепятствовать пользователю с высоким уровнем доступа, даже случайно, раскрыть какие-то известные ему тайны.

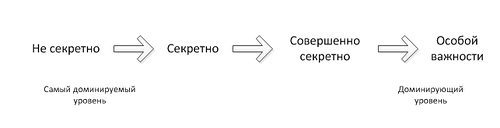
Одной из проблем этой модели считается беспрепятственность обмена информацией между пользователями одного уровня, так как эти пользователи могут выполнять в организации разные функции, и то, что имеет право делать пользователь А, может быть запрещено для Б. Поэтому в практике мандатную модель обычно используют совместно с какой-нибудь другой .

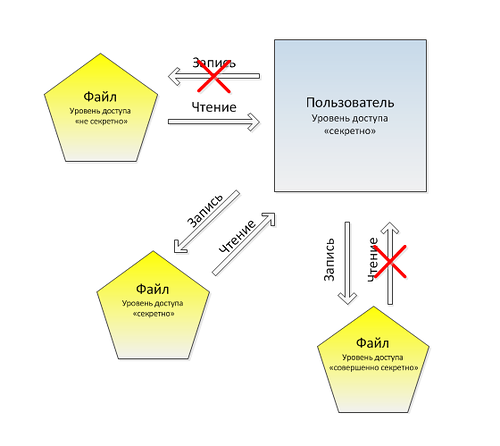
Из этих двух правил можно вынести несколько интересных наблюдений, указывающих на проблемы, которые могут проявиться в процессе адаптации модели к реальному приложению:

• Пользователи «снизу» могут попытаться передать информацию наверх, выложив ее на своем уровне. При этом они никогда не узнают, читал ли ее кто-либо «сверху» или нет, так как документ будет защищен от редактирования вышестоящими лицами.

• Еще пользователи могут попробовать «закинуть» данные на уровень выше. В этом случае верха будут иметь возможность вставить в полученный документ свои комментарии, но отправитель об этом также не узнает. Вообще, о существовании верхних уровней он может узнать только из документации к системе.

• У пользователей с высоким уровнем допуска нет никаких возможностей коммуникации с нижними уровнями. Возможно, наверху сидят очень умные люди, советы которых были бы просто бесценны, но мы об этом никогда не узнаем.

В основе мандатного управления доступом лежит мандатная модель безопасности Белла–ЛаПадулы. Эта модель основывается на правилах секретного документооборота применяемых во многих странах. В отличие от дискреционного управления, в котором пользователям непосредственно прописываются права на чтение, запись и выполнение, в мандатной модели управление доступом происходит неявным образом. Всем пользователям (субъектам) и файлам (объектам) назначаются уровни доступа. Например, «секретно», «совершенно секретно». Пример иерархии уровней доступа: 

Уровни доступа упорядочиваются по доминированию одного уровня над другим. Затем доступ к защищённым файлам осуществляется по двум простым правилам: 1. Пользователь имеет право читать только те документы, уровень безопасности которых не превышает его собственный уровень безопасности. Это правило обеспечивает защиту информации, обрабатываемой более высокоуровневыми пользователями, от доступа со стороны низкоуровневых пользователей. 2. Пользователь имеет право заносить информацию только в те документы, уровень безопасности которых не ниже его собственного уровня безопасности. Это правило предотвращает нарушение режима доступа со стороны высокоуровневых участников процесса обработки информации к низкоуровневым пользователям. Проиллюстрируем оба правила рисунком: 

На рисунке изображены отношения пользователя с уровнем «секретно» с субъектами в трехуровневой мандатной модели. Причем, «совершенно секретно» > «секретно» > «не секретно»

**2. Дискреционная модель**

В дискреционной модели безопасности управление доступом осуществляется путем явной выдачи полномочий на проведение действий с каждым из объектов системы. Например, в модели Харрисона-Руззо-Ульмана для этого служит матрица доступа, в которой определены права доступа субъектов системы к объектам. Строки матрицы соответствуют субъектам, а столбцы –объектам. Каждая ячейка матрицы содержит набор прав, которые соответствующий субъект имеет по отношению к соответствующему объекту.

Как правило, создатель объекта обладает на него полными правами и может делегировать часть прав другим субъектам. Дискреционный подход позволяет создать гораздо более гибкую схему безопасности, чем мандатный, но при этом он и гораздо более сложен в администрировании. С программной точки зрения его реализация очень проста, но при достаточно большом количестве объектов и субъектов система становится практически неуправляемой.

Для решения этой проблемы применяется, например, группировка пользователей. В этом случае права раздаются группам пользователей, а не каждому пользователю в отдельности. Для того чтобы пользователь получил соответствующие разрешения, нужно просто добавить его в одну или несколько групп. Также можно использовать типизацию объектов. Каждому объекту назначается тип, а для каждого типа определяется свой набор прав (схема доступа). В этом случае столбцы матрицы доступа соответствуют не объектам, а типам объектов. Комбинирование этого подхода с группировкой пользователей позволяют существенно уменьшить матрицу доступа, а значит, и упростить ее администрирование.

В сущности, набор прав – это не что иное, как список известных системе операций, снабженных разрешением или запретом на выполнение данной операции. В крупном приложении количество известных операций может быть весьма большим. При этом большая часть операций имеет смысл только для определенных типов объектов, а многие типовые процессы, осуществляемые пользователем в приложении, включают в себя выполнение нескольких элементарных операций над различными объектами. Поэтому, даже с уменьшенной матрицей доступа, продумать политику безопасности приложения, т.е. грамотно разделить полномочия между различными пользователями системы, достаточно сложно.

**3. Ролевая модель**

В ролевой модели операции, которые необходимо выполнять в рамках какой-либо служебной обязанности пользователя системы, группируются в набор, называемый «ролью».

Например, операции по регистрации документов могут быть сгруппированы в роль «регистратор». Для того чтобы множества операций, связанных с различными ролями, не пересекались, вводится иерархическая зависимость между ролями. К примеру, роль «секретарь» может включать в себя роль «регистратор» и, плюс к тому, еще несколько дополнительных операций.

Каждый пользователь системы играет в ней одну или несколько ролей. Выполнение пользователем определенного действия разрешено, если в наборе его ролей есть нужная, и запрещено, если есть нежелательная.

В этой модели у объектов нет определенных хозяев. Вся информация расценивается как принадлежащая организации, владеющей системой. Соответственно, и роли пользователя внутри системы – это роли, которые он играет в данной организации. Как следствие, пользователю невозможно делегировать права на какой-то определенный объект. Либо у него есть доступ ко всем подобным объектам системы, либо нет. Таким образом, преимуществом ролевой модели перед дискреционной является простота администрирования: назначение пользователей на роли и создание новых ролей не составляют никаких трудностей. В то же время она не позволяет управлять разными частями системы по отдельности, и тем более – делегировать какому-либо пользователю такие полномочия.

***Возвращаемся к основной теме лекции.***

Цветовое кодирование применяется как в качестве индикатора значений уровня доступа в области уведомлений, так и к рамке, обрамляющей окно приложения. На рис. 1.17 показан пример цветового кодирования окна файлового менеджера fly-fm для пользовательской сессии со значением уровня доступа равным 1 (как правило, в реальных ОССН этому значению соответствует уровень доступа «Для служебного пользования»).

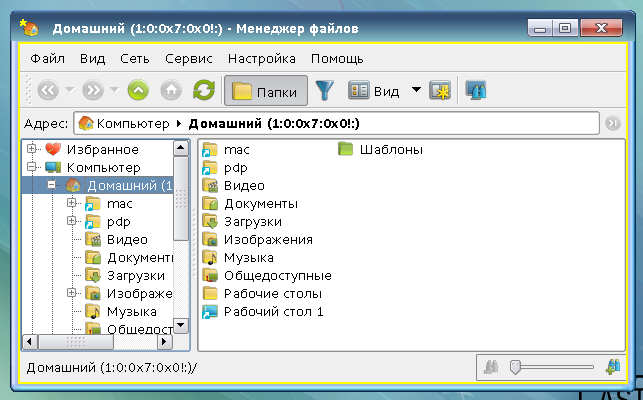
Детальное описание теоретических основ реализуемого в ОССН на основе мандатной сущностно-ролевой ДП-модели (МРОСЛ ДП-модели) мандатных управления доступом и контроля. целостности будет приведено в следующих лекциях.

Рисунок . Пример цветового кодирования значения уровня доступа субъект-сессии, соответствующей оконному приложению.

## 4. Завершение пользовательского сеанса и завершение работы

Для выхода пользователя из сеанса работы и/или завершения работы ОССН используется утилита «Диалог выхода из Fly» (утилита fly-shutdown- dialog), вызываемая из элемента «Системные» главного пользовательского меню (кнопка «Пуск»), интерфейс которой приведён на рис. .

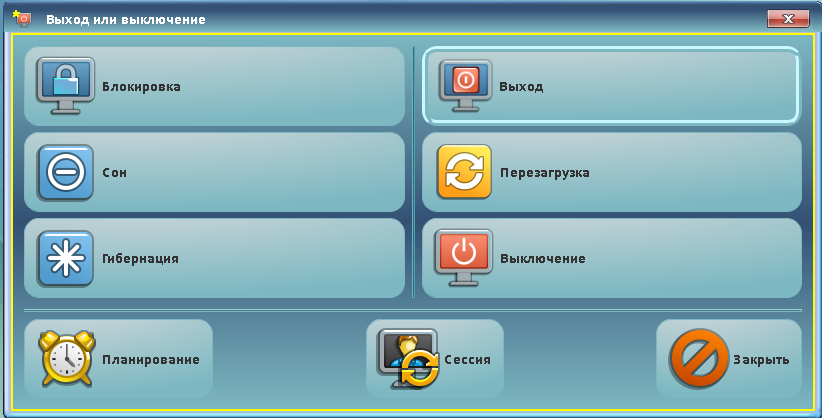


Рисунок . Интерфейс утилиты выхода из сессии и/или завершения работы.

Управляющие кнопки интерфейса сгруппированы по трём позициям:

• прерыванием работы системы — кнопки «Блокировка», «Сон», «Гибернация»;

• выход из пользовательской сессии или выключение (перезагрузка) ОССН — кнопки «Выход», «Перезагрузка», «Выключение»;

• планирование выполнения перечисленных функций или смена типа сессии пользователя — кнопки «Планирование», «Сессия», «Закрыть».

Интерфейс диалога смены типа сессии включает следующие элементы:

1. **Отдельная** - позволяет запустить пользовательскую сессию в новом виртуальном терминале. При этом сессия текущего пользователя останется открытой и в дальнейшем к ней можно будет вернуться в любой момент. Запуск новой пользовательской сессии возвращает пользователя к экрану регистрации в системе. При вводе данных учётной записи нового пользователя она будет открыта в новом виртуальном терминале. При этом в элементе «Меню» экрана регистрации в системе в подпункте «Сменить пользователя» появятся записи открытых пользовательских сессий с указанием номеров виртуальных терминалов, на которых они выполняются и типов пользовательских сессий (fly, default или console). В случае смены пользовательской сессии в защищённой графической подсистеме Fly при уже выполняющейся другой (других) пользовательской сессии в элементе меню «Тип новой графической сессии» появится дополнительный подпункт, указывающий, какой пользователь, на каком терминале и какого типа открыл эту сессию.

2. **Вложенная** — позволяет запустить новую пользовательскую сессию в текущей пользовательской сессии Fly в отдельном окне менеджера XDM (X Display Manager), который является составной частью сервера X.Org позволяет запускать графические сессии локально или удалённо. Для взаимодействия с программами Х-клиентами менеджер XDM использует протокол XDMCP (XDM Control Protocol). При этом пользовательская сессия запускается локально, и сервер X.Org взаимодействует с Х-клиентами через IP-адрес 127.0.0.1 узла localhost. Результатом этого является появление новой пользовательской сессии в окне менеджера XDM. Таким образом, в рамках одной пользовательской сессии Fly может функционировать другая пользовательская сессия Fly.

3. **Удалённая** — позволяет запустить новую пользовательскую сессию на удалённом компьютере. Здесь также используется менеджер XDM, только вместо IP-адреса 127.0.0.1 узла localhost используется IP-адрес удалённого компьютера под управлением ОССН. В остальном режим работы в удалённой пользовательской сессией ничем не отличается от локальной, за исключением того, что её пользователь получает доступ к своему домашнему каталогу на удалённом компьютере.