

# Пользователи и роли

## Пользователи

Для пользователя задается Имя, Полное имя и адрес электронной почты. На вкладке Дополнительно указываются роли.

Для заполнения реального профиля этих полей явно будет мало, поэтому используют следующую схему:

- Создается справочник (например Пользователи), в нем указывают необходимые реквизиты. Можно добавить дополнительные объекты, в которых есть связь со справочником Пользователи. Например, для бизнес процессов дополнительно создается справочник РолиИсполнителей (этот справочник в общем случае никак не связан с Ролями в контексте назначенных ролей пользователю) и регистр сведений РегистрАдресации. Подробнее см. в [Общая информация о задачах](#)
- Один из реквизитов (обычно Наименование) в справочнике Пользователи на уровне сравнения строк должен соответствовать Имени пользователя. Затем создается минимум один параметр сеанса, в который при авторизации сохраняется ссылка на справочник Пользователи. После этого через данную ссылку возможно получить всю дополнительную информацию по текущему пользователю. Для удобства и снижения количества запросов к БД стоит для часто используемых данных создать параметры сеанса и сохранять в них данные при авторизации.

## Программное добавление пользователя

```
ПользовательИБ = ПользователиИнформационнойБазы.СоздатьПользователя();
ПользовательИБ.Имя = "ИвановИИ";
ПользовательИБ.ПолноеИмя = "Иванов Иван Иванович";
ПользовательИБ.АутентификацияОС = Ложь;
ПользовательИБ.АутентификацияOpenID = Ложь;
ПользовательИБ.АутентификацияСтандартная = Истина;
ПользовательИБ.ЗапрещеноИзменятьПароль = Ложь;
ПользовательИБ.ПоказыватьВСпискеВыбора = Истина;
ПользовательИБ.Пароль = "";
//ПользовательИБ.ОсновнойИнтерфейс = Метаданные.Интерфейсы[""];
//ПользовательИБ.Язык = Метаданные.Языки[""];
//ПользовательИБ.Роли.Добавить(Метаданные.Роли[""]);
//ПользовательИБ.Роли.Добавить(Метаданные.Роли.АдминистраторСистемы);
//ПользовательИБ.РежимЗапуска = РежимыЗапуска[""];
```

```
ПользовательИБ.Записать();
```

```
НовыйПользователь = Справочники.Пользователи.СоздатьЭлемент();
```

```
НовыйПользователь.Наименование = "ИвановИИ";
```

```
НовыйПользователь.ОбменДанными.Загрузка = Истина;
```

```
НовыйПользователь.ИдентификаторПользователяИБ = ПользовательИБ.УникальныйИдентификатор;
```

```
НовыйПользователь.Недействителен = Ложь;
```

```
НовыйПользователь.Подготовлен = Истина;
```

```
НовыйПользователь.Записать();
```

Т е можно сделать отдельную обработку для создания пользователя и настройки ролей (ролей в контексте системы и в контексте расширенных параметров за счет справочников/...), из полей которой брать нужные параметры. Можно также добавить создание записей в дополнительных справочниках/... Можно усовершенствовать до создания шаблонов прав доступа, и создавать пользователей без необходимости создания множества записей в разных таблицах, без повторного ввода данных и вероятности где-то ошибиться при ручном вводе.

Скорее всего, есть готовые обработки (если нет, то через Метаданные и пару недель кодирования это реализуется), которые создадут базовые матрицы доступа по всем пользователям для упрощения контроля предоставления прав.

Контроль итоговых прав и выяснения, почему конкретный Петя не видит нужное - на самом деле не тривиальная задача, поскольку классический термин "права доступа" в 1С объединяется с классическим термином "правила отображения". Правила отображения могут быть простыми условиями в рамках запроса в одной из форм, могут размещаться в разных частях конфигурации. А механизм RLS - просто способ неявного автоматического добавления текста условий к исполняемому запросу + небольшие ограничения (неясно практически для чего они нужны). Понять и простить.

## Роли

После добавления роли, нужно обновить БД клавишей F7 чтобы роль отобразилась в списке ролей пользователей.

Ролевая модель доступа. Роли независимые, один уровень. Иерархии ролей нет. У одного пользователя может быть несколько ролей. При нескольких ролях у пользователя приоритет разрешения. Если хотя бы в одной роли есть разрешение, то доступ будет разрешен, если во всех ролях есть запрещение, то доступ будет запрещен.

Роли можно копировать для упрощения настроек.

```
&НаСервере
```

```
Процедура ПросмотретьРолиПользователяСервер()
```

```
СписокРолей = Новый Массив;  
Роли = ПользователиИнформационнойБазы.ТекущийПользователь().Роли;  
Для каждого роль из роли цикл  
Сообщить(роль.Имя);  
КонецЦикла  
КонецПроцедуры
```

Права доступа могут настраиваются с точностью до записи базы данных (например у одного пользователя может быть доступ к одним записям справочника Контрагенты, у другого - к другим),

Все права делятся на основные и интерактивные. Основные права (чтение, добавление, изменение, удаление) это разрешения на действия, выполняемые над элементами данных системы или над всей системой в целом, и проверяются независимо от способа обращения к данным. Интерактивные права это разрешения на действия, выполняющиеся пользователем интерактивно. Проверяются при выполнении интерактивных операций стандартными способами. В клиент-серверном варианте проверка основных прав выполняется на сервере, интерактивных на клиенте.

Проверку интерактивных прав доступа можно обойти, если создать в конфигураторе собственную форму объекта и заменить стандартные команды собственными. Проверку основных прав обойти нельзя. Допускается установка основного права и сброс интерактивного, но не наоборот.

Основные и интерактивные права взаимосвязаны. Целостность взаимных связей отслеживается системой автоматически, при установке подчиненного права устанавливаются родительские, при снятии родительского удаляются все подчиненные.

Иерархия прав

**Права корневого элемента конфигурации.** Устанавливается набор прав в контексте способов доступа, административных функций. Деление на группы субъективно, для моего удобства.

Право	Действия
Способы взаимодействия	
Тонкий клиент	право запуска тонкого клиента
Веб клиент	право запуска веб-клиента
Мобильный клиент	право запуска мобильного клиента
Толстый клиент	право запуска толстого клиента

Право	Действия
Внешнее Соединение	право взаимодействия между базами 1С по COM технологии в качестве сервера (про деление на тип сервер / клиент предположение, не проверял)
Automation	возможность взаимодействия между базами 1С по COM технологии или использование внешнего OLE соединения в качестве клиента. Например вызов сервера автоматизации OLE MS Office. (про деление на тип сервер / клиент предположение, не проверял)
<b>Управление функциями и данными</b>	
Администрирование данных	<p>право административных действий над данными. В право входит:</p> <ul style="list-style-type: none"> <li>• Просмотр записей журнала регистрации и получение значений отбора без ограничений.</li> <li>• Установка часового пояса информационной базы (области данных) – вызов метода УстановитьЧасовойПоясИнформационнойБазы().</li> <li>• Создание начального образа подчиненного узла распределенной информационной базы (вызов метода СоздатьНачальныйОбраз()).</li> <li>• Обновление нумерации объектов (вызов метода ОбновитьНумерациюОбъектов()).</li> <li>• Выполнение методов объекта СтандартноеХранилищеНастроекМенеджер в тех случаях, если выполняется работа с настройками пользователя, отличными от текущего.</li> <li>• Выполнение методов ПолучитьДанныеРегистрацииИнформационнойБазы()/ УстановитьДанныеРегистрацииИнформационнойБазы().</li> </ul> <p>Не позволяет открыть конфигурацию в конфигураторе, т е для манипулирования данными через созданные процедуры в режиме Предприятие.</p>

Право	Действия
Администрирование	<p>предоставляет полный доступ к любым данным и функциям. Подчинено праву Администрирование данных.</p> <ul style="list-style-type: none"> <li>• Редактирование конфигурации</li> <li>• Административные действия над всей информационной базой и редактирование списка пользователей (право для конфигурации).</li> <li>• Возможность настройки параметров соединения (для внешнего источника данных).</li> <li>• Отображение списка лицензий, использовавшихся при работе с конфигурацией и информационной базой, в окне О программе.</li> <li>• Удаление областей данных (включая удаление всех областей сразу).</li> <li>• Возможность выполнения фонового обновления конфигурации базы данных на стороне клиента.</li> <li>• Возможность устанавливать имя профиля безопасности в диалогах управления расширениями.</li> </ul>
Администрирование Расширений Конфигурации	управляет доступом к интерфейсу управления расширениями (как интерактивному, так и программному) в режиме «1С:Предприятие». В безопасном режиме администрирование расширений невозможно.
Обновление Конфигурации Базы Данных	позволяет выполнять обновление конфигурации базы данных в неинтерактивном (не через конфигуратор) режиме, т е через обработку. Не позволяет в конфигураторе открыть конфигурацию.
Режим технического специалиста	Похоже из практического позволяет получить ссылку на текущий объект или перейти по существующей ссылке.
Регистрация в системе взаимодействия	Мутное право. Вроде требует полного доступа к базе, но при установке этого права, административные права не устанавливаются. Хз.
<b>Монопольный режим и статус пользователей</b>	
Монопольный Режим	право на использование <a href="#">монопольного режима</a>
Завершение монопольного режима при начале сеанса	Сеанс другого пользователя с монопольным режимом может быть завершен при начале нового сеанса пользователя с правом ЗавершениеМонопольногоРежимаПриНачалеСеанса.
Активные пользователи	Просто просмотр списка активных пользователей.
Журнал регистрации	Просмотр журнала авторизации пользователей.

Право	Действия
Доступные типы интерфейсов в режиме Предприятие	
5 штук, говорят сами за себя	
Возможность запуска внешних обработок/отчетов	
Интерактивное открытие ... (2 штуки, также ясно).	
Вывод	Возможность копирования в буфер, печати, сохранения в файл
Другое	

Право	Действия
Сохранение данных пользователя	<p>Если право выключено, то:</p> <ul style="list-style-type: none"> <li>• Недоступна кнопка История, обращение к истории из встроенного языка вызывает исключение, интерактивные операции записи не фиксируются в истории.</li> <li>• В панели системных команд недоступна команда вызова списка избранного. В формах Получение ссылки и Переход по ссылке недоступна кнопка Добавить в избранное. При обращении к избранному из встроенного языка вызывается исключение.</li> <li>• Недоступна настройка форм (отсутствует команда Изменить форму).</li> <li>• Недоступны команды настройки панели разделов, панели навигации и начальной страницы.</li> <li>• В отчетах недоступны команды сохранения пользовательских настроек и вариантов отчета. Не выдается предложение запомнить изменения, выполненные пользователем в текущем варианте отчета. Изменения не сохраняются. Команда выбора настроек отчета доступна только в том случае, если у отчета или конфигурации установлено хранилище пользовательских настроек отчета. При вызове метода УстановитьТекущиеПользовательскиеНастройки() выдается исключение.</li> <li>• Недоступны команды сохранения и восстановления данных форм (команды Сохранить параметры и Восстановить параметры). Не выполняется автоматическое сохранение данных форм.</li> <li>• Настройки окон (размер и положение) сохраняются только на время сеанса.</li> <li>• Настройки размера панелей главного окна сохраняются только на время сеанса.</li> <li>• При программном сохранении настроек выдается исключение.</li> <li>• Настройки клиентского приложения сохраняются только на время сеанса. Не отображается флажок Устанавливать режим разрешения отладки при запуске (для тонкого и толстого клиента). Не отображается флажок Устанавливать режим низкой скорости соединения при запуске (для веб-клиента).</li> <li>• Не сохраняется настройка периода в списках. В диалоге настройки периода недоступен флажок Использовать эту настройку периода при открытии.</li> </ul>

**Дочерние классы конфигурации.** Основные и интерактивные. Основные (неинтерактивные) рассматривались ранее. Интерактивные делятся на операции с обычными элементами, с предопределенными, и историей данных. Есть специфичные права для конкретных классов, например для бизнес процессов - Старт, интерактивный старт и интерактивная активация. Сложность в том, что при жесткой политике доступа, некоторые функции (из-за связанного использования) могут требовать права к другим объектам. Поэтому желательно автоматическое тестирование при настройке прав (следующий раздел).

### **Неявное использование прав доступа и ускорение работы**

Если необходимо часть действий выполнять без учета установленных ограничений, а полные права на эти объекты давать из соображений безопасности не стоит, следует вынести эти действия в привилегированные модули или явно включать и выключать привилегированный режим в соответствующих местах программного кода.

Объекты конфигурации могут выполнять обращения к некоторым полям базы данных неявно. При этом наложение ограничений доступа выполняется способом «все», что может приводить к неожиданным сообщениям о нарушении прав доступа. Поэтому ограничивать доступ к таким полям нельзя.

Включение автонумерации или контроля уникальности номеров объектов приводит к неявному чтению поля Код (для документов, бизнес процессов и задач – Номер) при создании нового объекта и при его записи. Если в справочниках в качестве серии кодов выбрано «в пределах подчинения», то происходит неявное чтение полей Код, Родитель. Если используется иерархия групп и элементов, то кроме этого неявно считывается поле ЭтоГруппа. При выборе в качестве серии кодов «в пределах подчинения владельцу» неявно будет выполняться чтение полей Код и Владелец.

Если при записи документа, бизнес-процесса или задачи установлен режим автоматического определения времени, то при записи будет неявно выполняться чтение полей Дата и Ссылка. Поэтому чтение полей Дата и Ссылка должно быть разрешено.

Также рекомендуется проиндексировать реквизиты, которые используются в ограничениях доступа. Ускорение работы с использованием индексов достигается за счет того, что индекс имеет структуру, оптимизированную под поиск, например, сбалансированного дерева. Но индексы занимают дополнительный объем памяти, поэтому перед созданием индекса следует убедиться, что планируемый выигрыш в производительности запросов превысит дополнительную затрату ресурсов компьютера на поддержание индекса.

Использование нескольких таблиц и соединений в ограничениях доступа приводят к усложнению запроса. Поэтому рекомендуется реквизиты, на которые опирается определение доступности записей, включать в состав самого объекта конфигурации, а не обращаться к ним через точку. Это приведет к хранению избыточной информации, но позволит увеличить скорость выполнения запроса.



## Ограничения доступа на уровне записей базы данных

Использовать в случае крайней необходимости!

Механизм RLS (Record Level Security). Нагружает систему дополнительными запросами.

Настраивается только для основных прав. Для регистров накопления, бухгалтерского учета и расчета условия позволяют разграничить доступ по значениям измерений (для регистров бухгалтерского учета по балансовым измерениям), а для объектных данных и регистров сведений условия позволяют разграничивать доступ к данным по любым полям.

Для операций изменения, добавления и удаления можно задать только одно условие, а для операции чтения можно задать несколько ограничений доступа на уровне записей.

Способы настройки ограничений:

	<Прочие поля>	Конкретное имя поля
Способ установки ограничения	На всю запись	На отдельное поле записи
Использование при запросе	Будет накладываться для всех полей объекта, кроме полей, для которых ограничения заданы явным образом	Условие будет накладываться только в том случае, если в запросе присутствует поле, для которого задано ограничение

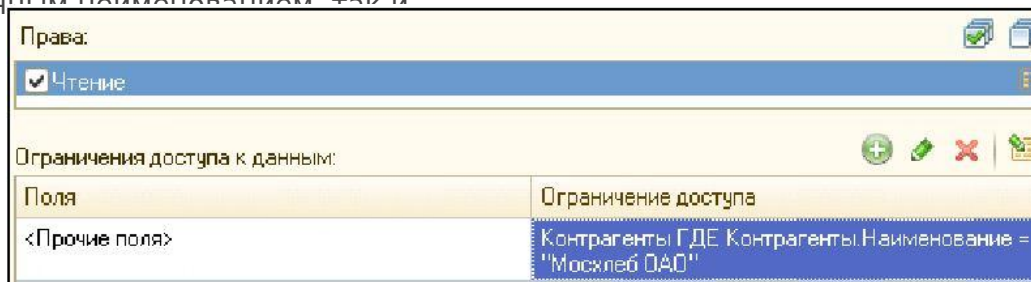
Данные могут быть выбраны из базы запросом или при помощи объектной техники. При использовании объектного чтения объект всегда будет считан из базы целиком. При использовании запроса есть возможность явно указать только необходимые поля, поэтому если в запрос не попадут поля с ограничениями, то данные записей будут предоставлены. Есть дополнительная деталь. Существует два способа функционирования ограничений доступа:

- Все. Операция должна быть выполнена над всеми подразумеваемыми данной операцией объектами базы данных. Если при выполнении такой операции должны быть прочитаны или изменены объекты базы данных, для которых не выполняются соответствующие ограничения доступа, то операция завершается аварийно из-за нарушения прав доступа
- Разрешенные. При выполнении операции над данными должны быть прочитаны только те объекты базы данных, которые удовлетворяют соответствующим ограничениям доступа. Объекты базы данных, не удовлетворяющие ограничениям доступа, при выполнении такой операции считаются отсутствующими и на результат операции не влияют.

При отображении динамических списков используется способ «Разрешенные», а при получении объектов средствами встроенного языка и при записи объектов в базу данных применяется способ «Все». В запросах способом функционирования ограничений можно управлять. Если в тексте запроса используется ключевое слово РАЗРЕШЕННЫЕ, то работа

ограничений выполняется в соответствии с одноименным способом, в противном случае используется способ «Все».

Пример отличия в исполнении запроса в режиме ВСЕ и в режиме РАЗРЕШЕННЫЕ. В таблице есть поле Контрагенты, на него установлено ограничение на наименование. Есть и как записи с данным наименованием, так и



Текст первого запроса:

ВЫБРАТЬ

Контрагенты.Наименование КАК Наименование

ИЗ

Справочник.Контрагенты КАК Контрагенты

Текст второго запроса

ВЫБРАТЬ РАЗРЕШЕННЫЕ

Контрагенты.Наименование КАК Наименование

ИЗ

Справочник.Контрагенты КАК Контрагенты

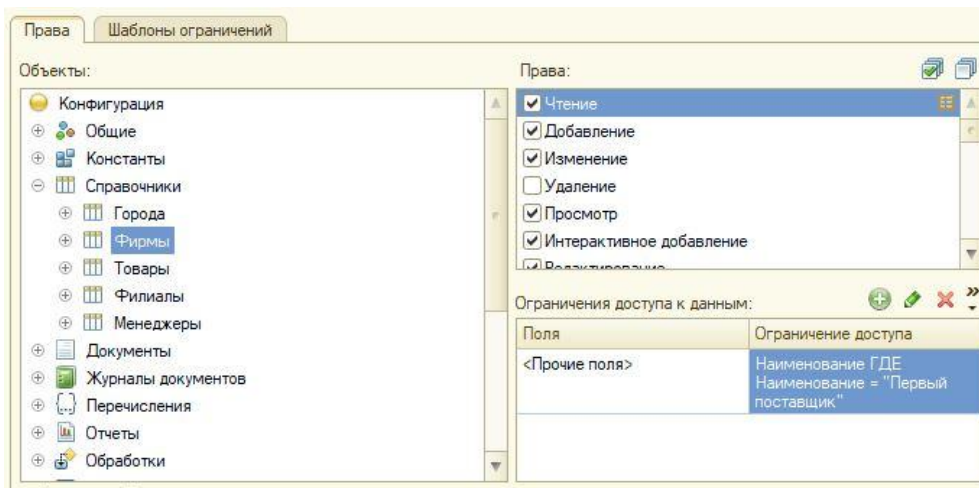
В первом случае будет ошибка "... недостаточно прав ...". Второй запрос вернет список, проходящий по установленному ограничению.

### Язык ограничения доступа к данным.

Подмножество языка запросов. В нем необходимо описать условие (секцию ГДЕ запроса). Такая секция будет добавляться к любым запросам при обращении к этому объекту. Если условие для объекта принимает значение Истина, значит, операция выполняется. Особенности языка:

- В запросе всегда присутствует одна таблица в качестве источника данных – это таблица объекта, на который накладывается ограничение
- В запросе доступны только секции ИЗ и ГДЕ языка запросов
- В условиях можно указывать параметры сеанса и функциональные опции в качестве параметров запроса
- Не допускается применение оператора В ИЕРАРХИИ и предложения ИТОГИ

- Нельзя использовать виртуальные таблицы регистров (например, СрезПоследних или ОстаткиИОбороты)
- В запросе можно использовать шаблоны, упрощающие написание ограничений.



зочника Фирмы:

Можно обращаться через

точку к полям реквизитов основной таблицы

Контрагенты ГДЕ Контрагенты.Регион.Наименование = "Иркутск"

Можно использовать соединения нескольких таблиц. Например, необходимо иметь доступ только к тем контрагентам, которые указаны как основной поставщик в каком-либо товаре:

Контрагенты ИЗ Справочник.Контрагенты КАК Контрагенты  
ВНУТРЕННЕЕ СОЕДИНЕНИЕ  
Справочник.Товары КАК Товары  
ПО  
Контрагенты.Ссылка = Товары.Поставщик

Здесь явно нет секции ГДЕ, однако такое ограничение работает. Применяется следующий алгоритм:

- Запись считается доступной если в результате работы условия для одной записи таблицы основного объекта ограничения получена непустая таблица (т.е. таблица, содержащая не менее одной записи).
- Если в результате работы условия получается пустая таблица, то запись считается недоступной.

В качестве параметров в тексте запроса допустимо использовать параметры сеансов и не зависящие от параметров функциональные опции.

ГДЕ Автор = &ТекущийПользователь

Ограничения, полученные из одной роли, объединяются операцией И. Ограничения, полученные из разных ролей, объединяются операцией ИЛИ.

### Инструкции препроцессора.

Дополнительная "нарезка" запроса.

```
#Если <Выражение> #Тогда
#ИначеЕсли <Выражение> #Тогда
#Иначе
#КонецЕсли
```

Выражения должны иметь тип Булево. В них можно использовать параметры сеанса. В зависимости от истинности, в текст запроса будет включено то или иное выражение. Если текст ограничения доступа содержит инструкции препроцессора, то его нельзя редактировать при помощи конструктора.

### Шаблоны ограничений доступа.

Шаблоны ограничений актуальны в пределах одной роли. Текст шаблона содержит фрагмент ограничения доступа. В нем можно использовать параметры. Параметры в шаблоне выделяются символом «#». После этого символа далее можно использовать:

- Ключевое слово Параметр, после которого в скобках указывается номер параметра в шаблоне
- Ключевое слово ТекущаяТаблица – обозначает вставку в текст полного имени таблицы, для которой строится ограничение
- Ключевое слово ИмяТекущейТаблицы – обозначает вставку в текст полного имени таблицы (как строковое значение, в кавычках), к которой применяется инструкция, на текущем варианте встроенного языка
- Ключевое слово ИмяТекущегоПраваДоступа – содержит имя права, для которого выполняется текущее ограничение: ЧТЕНИЕ, ДОБАВЛЕНИЕ, ИЗМЕНЕНИЕ, УДАЛЕНИЕ
- Имя параметра шаблона – означает вставку в текст ограничения соответствующего параметра шаблона
- Символ "#" – обозначает вставку в текст одного символа "#".

В тексте ограничения используется формат #ИмяШаблона(...Параметры...). Параметры передаются в виде строки, затем добавляются в текст в виде строки (второй пример). Те шаблоны играют роль простого шаблонизатора текста. Но это упрощение может очень сильно запутать код.

### Пример шаблона ограничения по автору:

Текст шаблона:

ГДЕ #Параметр(1) = &ТекущийПользователь

Текст в ограничении

#ОграничениеПоАвтору("Автор")

Итог:

ГДЕ Автор = &ТекущийПользователь

Если в другом документе вместо реквизита Автор используется реквизит с именем Ответственный, то ограничение доступа:

#ОграничениеПоАвтору("Ответственный")

### Пример шаблона ограничения с передачей "сложного" выражения

Текст шаблона:

ГДЕ #Параметр(1) = &ТекущийПользователь #Параметр(2)

Текст в ограничении

#ОграничениеПоАвтору("Автор", "ИЛИ ЭтоГруппа")

Итог:

ГДЕ Автор = &ТекущийПользователь ИЛИ ЭтоГруппа