Терминальный клиент. Описание

Терминальный клиент, это frontend часть к системе CLI, выполненная в виде консольного приложения. Данное консольное приложение предназначено для использования в качестве командного интерпретатора. Это означает, что мы можем создать пользователя, назначит ему терминальный клиент в качестве командного интерпретатора, после чего при заходе в систему под этим пользователем (как локально, так и через telnet, ssh) мы будем попадать в терминальный клиент. При этом один экземпляр терминального клиента предназначен для взаимодействия только с одним пользователем, при этом в системе одновременно может выполняться несколько терминальных клиентов.

Функциональность:

* Один экземпляр терминального клиента взаимодействует только с одним пользователем.
* Реализует обязательный вход пользователя в систему (ввод аутентификационных данных пользователя) перед началом работы.
* Принимает ввод пользователя.
* Позволяет редактировать ввод пользователя при помощи той же комбинации клавиш, что и командная оболочка bash.
* Поддерживает историю команд в том же виде (с тем же набором комбинаций клавиш и команд), что и командная оболочка bash.
* Поддерживает автодополнение вводимой строки; за данными об автодополнении обращается на CLI сервис.
* Поддерживает несколько разных режимов работы (состояния) терминального клиента. В зависимости от режима изменяется функциональность и поведение терминального клиента. На данный момент поддерживаются следующие режимы работы: режим ввода команд, режим обработки команд, режим ввода данных аутентификации (режим ввода имени пользователя и пароля).
* Поддерживает возможность определять набор команд, выполнение которых происходит на самом терминальном клиенте. На данный момент этот набор команд пуст.
* Отправляет ввод пользователя, не являющийся одной из команд терминального клиента, на CLI сервис для его обработки.
* Принимает сообщения от CLI сервиса; реакция на эти сообщения зависит как от вида сообщений, так и от текущего режима работы (состояния) терминального клиента.

Поведение

* При запуске терминальный клиент требует от пользователя ввода аутентификационных данных (требует осуществить вход в систему); при этом терминальный клиент находится в режиме ввода данных аутентификации (см. ниже). После ввода аутентификационных данных терминальный клиент отправляет синхронный запрос на вход на CLI сервис. Если аутентификационные данные корректны, то запрос на вход будет успешным; после этого терминальный клиент переходит в режим ввода команд. В случае неуспешного запроса на вход в систему, терминальный клиент остается в режиме ввода данных аутентификации и предлагает пользователю ввести эти данные снова. Если количество попыток входа в систему (ввода аутентификационных данных) превысит некоторый лимит (задаваемый в конфигурационном файле), то терминальный клиент завершает свою работу с ненулевым кодом возврата (если данный лимит установлен в 0, то пользователь может пытаться войти в систему бесконечно долго).
* После входа пользователя в систему происходит обращение на CLI сервис для получения начального состояния (строки подсказки или prompt) при помощи синхронного обмена сообщениями.
* В режимах работы терминального клиента, поддерживающих автодополнение, за данными автодополнения терминальный клиент обращается на CLI сервис при помощи синхронного обмена сообщениями.
* Ввод пользователя считается завершенным при нажатии на клавишу «Enter». После этого ввод пользователя обрабатывается как команда (либо на терминальном клиенте, либо на CLI сервисе).
* После завершения ввода пользователем, этот ввод сначала проверяется на соответствие одной из команд терминального клиента. Если ввод пользователя соответствует одной из команд терминального клиента, то выполняется эта команда. При этом сама команда терминального клиента управляет переключением режима работы. Если ввод пользователя не соответствует какой-либо команде терминального клиента, то ввод пользователя считается командой CLI сервиса и посылается на CLI сервис при помощи асинхронного сообщения. При этом терминальный клиент переходит в режим обработки команды.
* При переходе в режим обработки команд (после отправки ввода пользователя на CLI сервис) терминальный клиент ожидает одно или несколько сообщений следующих типов: **command\_out**, **command\_err** и **end**. При получении сообщения типа **command\_out**, терминальный клиент выводит содержимое этого сообщения в стандартный поток вывода. При получении сообщения типа **command\_err**, терминальный клиент выводит содержимое этого сообщения в стандартный поток вывода ошибок. При получении сообщения типа **end**, терминальный клиент переходит в режим ввода команд. При этом содержимое сообщения типа **end** трактуется как строка подсказки (prompt).
* Результат прерывания работы по нажатию на комбинацию клавиш Ctrl+C зависит от текущего режима работы терминального клиента: для режима ввода команд прерывание приведет к сбросу ввода пользователя, для режима обработки команд — к отправке на CLI сервис сообщения **interrupt**, для режима ввода данных аутентификации — к сбросу введенных данных аутентификации (переход к началу ввода данных аутентификации).
* Прерывание работы по нажатию на комбинацию клавиш Ctrl+C для режима обработки команд приведет к отправке на CLI сервис асинхронного сообщения **interrupt**. В ответ на это сообщение CLI сервис может как сразу прислать сообщение типа **end**, так и после отсылки ряда других сообщений (сообщений типа **command\_out** и **command\_err**).
* Прерывание работы по нажатию на комбинацию клавиш Ctrl+C для режима ввода данных аутентификации приведет к сбросу введенных данных аутентификации, т.е. данные аутентификации будут вводиться с самого начала. При этом сброс введенных данных аутентификации не считается попыткой входа (не увеличивает счетчик попыток входа в систему).
* Прерывание работы по нажатию на комбинации клавиш Ctrl+Z или Ctrl+\ приводит к завершению работы терминального клиента. Перед завершением работы на CLI сервис посылается асинхронное сообщение **exit**.
* При нажатии на комбинации клавиш Ctrl+D происходит выход из текущего режима работы (в случае фундаментального режима работы CLI выход из текущего режима означает выход пользователя из системы и завершение работы терминального клиента). При этом на сервер шлется синхронный запрос. Дальнейшее поведение терминального клиента зависит от ответа. В случае ответа о выходе из текущего режима, терминальный клиент обновляет строку подсказки (prompt) данными из ответа. В случае ответа о завершении работы терминальный клиент завершает свою работу.
* При нажатии на клавишу ? происходи запрос помощи или списка доступных команд. При этом происходит синхронное взаимодействие с сервером. Выбор того, что на самом деле запрашивается на сервере зависит от последнего введенного символа (перед нажатием на клавишу ?). Если этот символ пробельный (пробел, табуляция), то происходит запрос помощи. В противном случае происходит запрос списка доступных команд. Если в результате запроса помощи или списка доступных команд ничего не будет найдено, то на экран ничего не выводится; в том числе и символ '?'. Если в результате запроса помощи или списка доступных команд будет найдена некоторая информация, то на экран выводится символ '?', символ перевода строки и найденная информация (уже на новой строке).
* Если пользователь бездействует в течении некоторого времени, то терминальный клиент получит уведомление о завершении клиентской сессии данного пользователя на CLI сервисе. Данное сообщение выводится в поток стандартного вывода, после чего терминальный клиент завершит свою работу.
* CLI сервис может слать уведомления на терминальный клиент в любой момент времени (на данный момент — только уведомление о прекращении работы из-за превышения времени простоя).

Режимы работы терминального клиента

Терминальный клиент во время своей работы находится в том или ином режиме работы (состоянии). В зависимости от режима работы изменяется функциональность и поведение терминального клиента. На данный момент определены следующие режимы работы терминального клиента:

* Режим ввода данных аутентификации (режим ввода имени пользователя и пароля). В этом режиме терминальный клиент ожидает от пользователь ввода имени пользователя и пароля, при этом имя пользователя маскируется (каким-либо способом). В этом режиме терминальный клиент позволяет пользователю редактировать свой ввод (по большей части при вводе имени пользователя), но не позволяет использовать историю команд и автодополнение. После ввода данных аутентификации, эти данные передаются на CLI сервис с помощью синхронного запроса о входе в систему (о вводе аутентификационных данных пользователя). В случае успешного входа в систему, терминальный клиент переходит в режим обработки команд. В противном случае терминальный клиент остается в режиме ввода данных аутентификации и предлагает пользователю ввести имя и пароль снова. Если же количество попыток ввести данные пользователя превысит лимит, задаваемый в конфигурационном файле (см. ниже), то терминальный клиент завершает свою работу (с ненулевым кодом завершения). Прерывание работы по нажатию на комбинацию клавиш Ctrl+C сбрасывает введенные данные аутентификации и переходит к началу ввода данных аутентификации (при этом терминальный клиент остается в режиме ввода данных аутентификации).
* Режим ввода команд. В этом режиме терминальный клиент ожидает ввода команд от пользователя, позволяя пользователю редактировать свой ввод, пользоваться историей команд и автодополнением. При завершении ввода, введенная команда выполняется либо локально (на терминальном клиенте), либо передается на CLI сервис. Если введенная команда является командой CLI сервиса (т.е. передается на CLI сервис), то терминальный клиент переходит в режим обработки команд. В данном режиме работы терминальный клиент ожидает появление уведомлений со стороны CLI сервиса (например, о завершении работы обработчика клиентских запросов от данного терминального клиента). Прерывание работы по нажатию на комбинацию клавиш Ctrl+C приводит к сбросу ввода пользователя.
* Режим обработки команд. В этом режиме терминальный клиент ожидает окончания обработки команды на CLI сервисе. Ввода от пользователя не ожидается; поэтому возможность редактирования ввода, использования истории команд и автодополнения отсутствует. В режиме обработки команд терминальный клиент ожидает сообщения следующего типа от CLI сервиса: **command\_out**, **command\_err** и **end** (см. протокол взаимодействия между терминальным клиентом и cli сервисом). При получении сообщения типа **command\_out**, терминальный клиент выводит содержимое этого сообщения в стандартный поток вывода. При получении сообщения типа **command\_err**, терминальный клиент выводит содержимое этого сообщения в стандартный поток вывода ошибок. При получении сообщения типа **end**, терминальный клиент переходит в режим ввода команд. Также, в данном режиме работы терминальный клиент ожидает появление уведомлений со стороны CLI сервиса (например, о завершении работы обработчика клиентских запросов от данного терминального клиента). Прерывание работы по нажатию на комбинацию клавиш Ctrl+C приводит к отсылке сообщения типа **interrupt** с терминального клиента на CLI сервис.

Вполне возможно, что в будущем появится еще несколько режимов работы терминального клиента; например, режим текстового редактора, режим взаимодействия с интерактивной командой и т.п.

Команды терминального клиента

Терминальный клиент поддерживает ряд команд, выполнение которых происходит на самом терминальном клиенте, а не передается для выполнения на CLI сервис. При завершении ввода пользователем, его ввод проверяется на соответствие команда терминального клиента. Если ввод пользователя распознается, как некоторая команда терминального клиента, то будет выполняться эта команда. Если ввод пользователя не распознается, как некоторая команда терминального клиента, то этот ввод будет послан на CLI сервис. Выполнение команд на терминальном клиенте происходит синхронно; при этом переключением режим управляет сама команда.

На данный момент набор команд терминального клиента пуст.

Детали реализации

* Для поддержки такой функциональности в терминальном клиенте, как редактирование ввода пользователя, автодополнение, история команд мы используем библиотеку readline.
* Для поддержки запроса помощи и списка доступных команд при нажатии на клавишу ?, мы задаем обработчик нажатия этой клавиши при помощи функции **rl\_bind\_key**.
* При обмене сообщениями с CLI сервисом в качестве сообщений мы используем объекты среды выполнения Erlang. Для их сериализации в массив байт (для передачи по сети) мы используем функцию **erl\_encode**, а для их десериализации из массива байт (для приема по сети) — функцию **erl\_decode** из библиотеки **erl\_marshal**. Эти объекты на самом деле являются кортежами, состоящими минимум из одного элемента (на данный момент — кортежами из одного или двух элементов)., причем первый элемент всегда является описателем сообщения (тип этого первого элемента всегда атом).
* Со стороны CLI сервиса асинхронно на терминальный клиент приходят следующие сообщения: в режиме обработки команды — сообщения **command\_out**, **command\_err** и **end**, а также уведомление **exit** (см. протокол взаимодействия между терминальным клиентом и CLI сервисом). Все эти сообщения имеют следующий вид: кортеж из двух элементов, первым элементом кортежа идет описатель сообщения, вторым элементом кортежа - строка с данными. Поэтому, в терминальном клиенте все эти сообщения будут десериализоваться в объекты одного типа данных, состоящего из описателя и данных. Различать мы эти объекты будем по описателю.
* Такие операции, как обработка завершения ввода от пользователя, обработка автодополнения и обмен сообщений с CLI сервисом не прерываются сигналами (SIGINT, SIGQUIT, SIGTSTP); это означает, что перед выполнение этих операций мы блокируем получение сигналов, а после выполнения этих операций мы разблокируем получение сигналов. При этом при блокировке сигналов мы используем пессимистический подход: блокируем получение сигналов для как можно более крупных блоков (например, блокируем полностью обработчик завершения пользовательского ввода).
* Терминальный клиент и CLI сервис всегда работают на одном компьютере. Поэтому все взаимодействия вида запрос-ответ мы выполняем синхронно. К такому типу взаимодействия относятся вход в систему (ввод аутентификационных данных пользователя), выход из текущего режима работы, запрос текущего состояния, запрос списка расширений, запрос помощи по команде и запрос списка доступных команд по префиксу. Весь остальной обмен сообщениями между терминальным клиентом и CLI сервисом выполняется асинхронно.
* Терминальный клиент и CLI сервис обмениваются сообщениями через сокеты TCP. В связи с тем, что терминальный клиент и CLI сервис всегда располагаются на одном компьютере, более логично было бы использовать сокеты Unix. Однако, библиотека языка Erlang не содержит модулей, позволяющих использовать сокеты Unix, но содержит модуль **gen\_tcp**, позволяющий использовать сокеты TCP.
* Комбинации клавиш Ctrl+D соответствует символ EOF (end of file). Поэтому, когда обработчик данных, вызываемый библиотекой readline, в качестве данных от пользователя получает NULL, мы считаем, что пользователь нажал на комбинацию клавиш Ctrl+D (которую библиотека readline преобразовала в EOF). При таком подходе, очевидно, будут проблемы при работе с перенаправлением стандартного потока ввода для чтения из файла (данную проблему мы решим в будущем).

Конфигурирование

Терминальный клиент может быть сконфигурирован двумя возможными (не пересекающимися) способами: с использованием конфигурационного файла и с помощью опций командной строки. Наличие конфигурационного файла является обязательным; использование опций командной строки обязательным не является. Конфигурационный файл обрабатывается построчно; существуют следующие правила по его обработке (разбору или парсингу):

* Строки, начинающиеся с символа **'#'** считаются комментариями и игнорируются.
* Строки вида **KEY**=**VALUE** разбираются в конфигурационный параметр **KEY** имеющий строковое значение **VALUE**. Если конфигурационный параметр с именем **KEY** встречается в конфигурационном файле несколько раз, то поведение (на данный момент) не определено.
* Все остальные строки считаются некорректными. При появлении такой строки поведение (на данный момент) не определено (если быть более точным, то на данный момент мы пропускаем некорректные строки).

Следующие параметры конфигурационного файла определены на данный момент:

* Параметр, определяющий номер listen порта CLI сервиса: **port\_number=число**. Здесь **число** должно иметь значение из диапазона 0-65535. Параметр **port\_number** является обязательным.
* Параметр, определяющий максимальное количество попыток входа пользователя в систему (максимальное количество попыток ввода аутентификационных данных пользователя): **login\_attempt\_count=число**. Здесь **число** любое положительное целое число, либо специальное значение 0, означающее, что количество попыток входа пользователя в систему не ограничено. Параметр **login\_attempt\_count** является обязательным.

По умолчанию, конфигурационный файл должен иметь следующее имя: */etc/cli\_terminal/cli\_terminal.conf*.

Опции командный строки являются не обязательными; следующие опции командной строки определены на данный момент:

* Параметр, определяющий местоположение конфигурационного файла: **--config=config-file-location**. Здесь **config-file-location** — путь (относительный или абсолютный) до конфигурационного файла терминального клиента. Данный параметр позволяет использовать конфигурационный файл с произвольным именем, распологающийся в произвольном месте файловой системы.