Network Working Group
Request for Comments: 4250
Category: Standards Track

S. Lehtinen
SSH Communications Security Corp
C. Lonvick, Ed.
Cisco Systems, Inc.
January 2006

# Номера, выделенные для протокола SSH

The Secure Shell (SSH) Protocol Assigned Numbers

**Статус документа**

В этом документе содержится спецификация протокола, предложенного сообществу Internet. Документ служит приглашением к дискуссии в целях развития и совершенствования протокола. Текущее состояние стандартизации протокола вы можете узнать из документа Internet Official Protocol Standards (STD 1). Документ может распространяться без ограничений.

**Авторские права**

Copyright (C) The Internet Society (2006).

**Аннотация**

В этом документе определены инструкции для IANA и начальные значения выделенные IANA для протокола SSH. Документ предназначен исключительно для инициализации реестров IANA, упомянутых в комплекте документов SSH.

# Оглавление

# *1. Введение*

В этом документе не определяются какие-либо новые протоколы. Документ предназначен только для создания начальных баз данных IANA для протокола SSH и содержит также инструкции по дальнейшему выделению значений. Кроме одного алгоритма, имеющего в основном историческое значение, данный документ не определяет никаких новых протоколов или диапазонов значений, которые не были бы определены в документах [SSH-ARCH], [SSH-TRANS], [SSH-USERAUTH] и [SSH-CONNECT].

# *2. Разработчики*

Основными разработчиками этого комплекта документов являются: Tatu Ylonen, Tero Kivinen, Timo J. Rinne, Sami Lehtinen (все из SSH Communications Security Corp) и Markku-Juhani O. Saarinen (университет Jyvaskyla). Darren Moffat был редактором этого комплекта документов и внёс важный вклад в работу.

За годы подготовки этого документа множество людей внесло свой вклад. В их число входят: Data Andersson, Ben Harris, Bill Sommerfeld, Brent McClure, Niels Moller, Damien Miller, Derek Fawcus, Frank Cusack, Heikki Nousiainen, Jakob Schlyter, Jeff Van Dyke, Jeffrey Altman, Jeffrey Hutzelman, Jon Bright, Joseph Galbraith, Ken Hornstein, Markus Friedl, Martin Forssen, Nicolas Williams, Niels Provos, Perry Metzger, Peter Gutmann, Simon Josefsson, Simon Tatham, Wei Dai, Denis Bider, der Mouse и Tadayoshi Kohno. Указанные в списке люди могли не участвовать в написании данного документа, но они внесли свой вклад в его подготовку.

# *3. Используемые в документе соглашения*

## 3.1. Ключевые слова RFC 2119

Во всех документах, связанных с протоколом SSH, следует использовать ключевые слова: **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не нужно** (SHALL NOT), **следует** (SHOULD), **не следует** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) для описания уровня требования. Интерпретация этих слов описана в [RFC2119].

## 3.2. Ключевые слова RFC 2434

Ключевые слова **приватное использование** (PRIVATE USE), **иерархическое выделение** (HIERARCHICAL ALLOCATION), **выделение в соответствии с порядком запросов** (FIRST COME FIRST SERVED), **экспертное рассмотрение** (EXPERT REVIEW), **требуется спецификация** (SPECIFICATION REQUIRED), **одобрение IESG** (IESG APPROVAL), **согласование с IETF** (IETF CONSENSUS), **стандартизация** (STANDARDS ACTION) в данном документе при их использовании в контексте распределения пространства имён интерпретируются в соответствии с [RFC2434]. Толкование ключевых слов для ясности продублировано в этом документе.

**Приватное использование** (PRIVATE USE) - только для приватного или локального использования с типом и целями, определяемыми локальным сайтом. Не предпринимается попыток предотвратить использование этого имени на других сайтах с иными (и несовместимыми) целями. Для IANA нет необходимости рассматривать такие объекты и в общем случае они бесполезны с точки зрения совместимости.

**Иерархическое выделение** (HIERARCHICAL ALLOCATION) - обладающие полномочиями менеджеры могут выделять контролируемые ими значения, как часть общего пространства имён. IANA контролирует верхние уровни пространства имён в соответствии со своей политикой.

**Выделение в соответствии с порядком запросов** (FIRST COME FIRST SERVED) - выделенные значения доступны каждому, кто предоставит о себе контактную информацию и краткое описание целей использования. Конкретные значения в общем случае выделяются IANA, конкретные имена обычно выдаются по запросам.

**Экспертное рассмотрение** (EXPERT REVIEW) - требуется одобрение уполномоченного эксперта (Designated Expert).

**Требуется спецификация** (SPECIFICATION REQUIRED) - значения и их трактовки должны быть документированы в RFC или иных легкодоступных документах с достаточным уровнем детализации для обеспечения совместимости между независимыми реализациями.

**Одобрение IESG** (IESG APPROVAL) - выделение должно быть одобрено IESG, но не требуется документирования в RFC (хотя IESG может затребовать документы или иные материалы для поддержки).

**Согласование с IETF** (IETF CONSENSUS) - новые значения выделяются с согласия IETF. В частности, новые значения выделяются в RFC одобренных IESG. Обычно IESG рассматривает предложения уполномоченных лиц (например, участников рабочей группы, если таковая существует).

**Стандартизация** (STANDARDS ACTION) - значения выделяются только для предложенных стандартов (Standards Track RFC), одобренных IESG.

## 3.3. Поля протокола и их значения

В данном наборе документов определяются поля протокола и возможные значения этих полей. Поля будут определяться вместе с протокольными сообщениями. Например, поле SSH_MSG_CHANNEL_DATA определяется следующим образом:

```
byte      SSH_MSG_CHANNEL_DATA
uint32    recipient channel (канал получателя)
string    data (данные)
```

В данном документе поля протокола будут указываться в одинарных кавычках, а значения полей – в двойных. В приведённом выше примере поле 'data' может содержать значения "foo" и "bar".

# *4. Взаимодействие с IANA*

Этот документ целиком представляет собой соображения IANA по поводу протокола SSH, как определено в документах [SSH-ARCH], [SSH-TRANS], [SSH-USERAUTH], [SSH-CONNECT]. Данный раздел включает соглашения, используемые при именовании пространств имён, начальное состояние реестра и рекомендации по распределению.

## 4.1. Номера сообщений

Поле номера сообщения (Message Number) представляет собой один байт, описывающий тип содержимого пакета.

### *4.1.1. Соглашения*

Пакеты SSH используют номера сообщений от 1 до 255. Эти номера распределены между различными компонентами:

**Протокол транспортного уровня**

1 - 19 базовые сообщения транспортного уровня (например, disconnect, ignore, debug и т. п.);
20 - 29 согласование алгоритма;
30 - 49 сообщения, связанные с обменом ключами (допускается совпадение номеров для разных методов аутентификации).

**Протокол аутентификации пользователей**

50 - 59 базовые сообщения протокола аутентификации;

60 - 79 сообщения, связанные с методом аутентификации (допускается совпадение номеров для различных методов).

**Протокол соединений**

80 - 89 базовые сообщения протокола;
90 - 127 сообщения, связанные с каналом.

**Зарезервировано для клиентских протоколов**

128 - 191 резерв

**Локальные расширения**

192 - 255 локальные расширения.

### *4.1.2. Начальное распределение*

В приведённой ниже таблице содержатся выделенные изначально значения идентификаторов сообщений (Message ID).

| *Message ID* | *Значение* | *Документ* |
|---|---|---|
| SSH_MSG_DISCONNECT | 1 | [SSH-TRANS] |
| SSH_MSG_IGNORE | 2 | [SSH-TRANS] |
| SSH_MSG_UNIMPLEMENTED | 3 | [SSH-TRANS] |
| SSH_MSG_DEBUG | 4 | [SSH-TRANS] |
| SSH_MSG_SERVICE_REQUEST | 5 | [SSH-TRANS] |
| SSH_MSG_SERVICE_ACCEPT | 6 | [SSH-TRANS] |
| SSH_MSG_KEXINIT | 20 | [SSH-TRANS] |
| SSH_MSG_NEWKEYS | 21 | [SSH-TRANS] |
| SSH_MSG_USERAUTH_REQUEST | 50 | [SSH-USERAUTH] |
| SSH_MSG_USERAUTH_FAILURE | 51 | [SSH-USERAUTH] |
| SSH_MSG_USERAUTH_SUCCESS | 52 | [SSH-USERAUTH] |
| SSH_MSG_USERAUTH_BANNER | 53 | [SSH-USERAUTH] |
| SSH_MSG_GLOBAL_REQUEST | 80 | [SSH-CONNECT] |
| SSH_MSG_REQUEST_SUCCESS | 81 | [SSH-CONNECT] |
| SSH_MSG_REQUEST_FAILURE | 82 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_OPEN | 90 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_OPEN_CONFIRMATION | 91 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_OPEN_FAILURE | 92 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_WINDOW_ADJUST | 93 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_DATA | 94 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_EXTENDED_DATA | 95 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_EOF | 96 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_CLOSE | 97 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_REQUEST | 98 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_SUCCESS | 99 | [SSH-CONNECT] |
| SSH_MSG_CHANNEL_FAILURE | 100 | [SSH-CONNECT] |

### *4.1.3. Последующее выделение*

Запросы на выделение новых номеров для сообщений из диапазонов 1 - 29, 50 - 59, 80 - 127 **должны** обрабатываться с использованием процедуры **стандартизации** (STANDARDS ACTION), описанной в [RFC2434].

Толкование сообщений с номерами от 30 до 49 связано с используемым методом обмена ключами и должно быть указано при определении соответствующего метода обмена ключами.

Толкование сообщений с номерами от 60 до 79 связано с используемым методом аутентификации и должно быть указано при определении соответствующего метода.

Запросы на выделение новых номеров в диапазоне от 128 до 191 **должны** обрабатываться путём согласования с IETF (метод IETF CONSENSUS), как описано в [RFC2434].

IANA не контролирует сообщения с номерами от 192 до 255. Этот диапазон будет сохранен для **приватного использования** (PRIVATE USE).

## 4.2. Коды и описания причин разрыва соединения

Код сообщения о разрыве (Disconnection Message 'reason code') представляет собой 32-битовое целое число без знака (uint32). Связанное с сообщением Disconnection Message описание ('description') причины включает понятный человеку текст, поясняющий причину разрыва соединения.

### *4.2.1. Соглашения*

Пакеты протокола, содержащие сообщение SSH_MSG_DISCONNECT, **должны** иметь код причины 'reason code' в диапазоне от 0x00000001 до 0xFFFFFFFF. Конкретные значения описаны в документе [SSH-TRANS].

### *4.2.2. Начальное распределение*

Приведённая таблица содержит выделенные описания (description) и коды причины (reason code) для сообщений SSH_MSG_DISCONNECT.

| *Имя* | *Код причины* |
|---|---|
| SSH_DISCONNECT_HOST_NOT_ALLOWED_TO_CONNECT | 1 |
| SSH_DISCONNECT_PROTOCOL_ERROR | 2 |
| SSH_DISCONNECT_KEY_EXCHANGE_FAILED | 3 |
| SSH_DISCONNECT_RESERVED | 4 |
| SSH_DISCONNECT_MAC_ERROR | 5 |
| SSH_DISCONNECT_COMPRESSION_ERROR | 6 |
| SSH_DISCONNECT_SERVICE_NOT_AVAILABLE | 7 |
| SSH_DISCONNECT_PROTOCOL_VERSION_NOT_SUPPORTED | 8 |
| SSH_DISCONNECT_HOST_KEY_NOT_VERIFIABLE | 9 |
| SSH_DISCONNECT_CONNECTION_LOST | 10 |
| SSH_DISCONNECT_BY_APPLICATION | 11 |
| SSH_DISCONNECT_TOO_MANY_CONNECTIONS | 12 |
| SSH_DISCONNECT_AUTH_CANCELLED_BY_USER | 13 |
| SSH_DISCONNECT_NO_MORE_AUTH_METHODS_AVAILABLE | 14 |
| SSH_DISCONNECT_ILLEGAL_USER_NAME | 15 |

### *4.2.3. Последующее выделение*

Значения кода причины для сообщений о разрыве (Disconnection Message) **должны** выделяться последовательно. Запросы на выделение новых значений кодов и связанных с ними описаний (Disconnection Message description) из диапазона 0x00000010 – 0xFDFFFFFF **должны** обрабатываться методом **согласования с IETF**, как описано в [RFC2434]. IANA не будет распределять значения Disconnection Message reason code из диапазона 0xFE000000 – 0xFFFFFFFF, сохраняемые для **приватного использования**, как описано в [RFC2434].

## 4.3. Причины и описания отказов в канальных соединениях

Коды причин отказа в канальных соединениях (Channel Connection Failure reason code) указываются 32-битовыми целыми числами без знака (uint32). Связанные с кодом описания (Channel Connection Failure description) представляют собой понятный человеку текст, объясняющий причину отказа. Описания приведены в документе [SSH-CONNECT].

### *4.3.1. Соглашения*

Пакеты протокола, содержащие сообщение SSH_MSG_CHANNEL_OPEN_FAILURE, **должны** иметь значение кода причины отказа (Channel Connection Failure reason code) из диапазона 0x00000001 - 0xFFFFFFFF.

### *4.3.2. Начальное распределение*

Значения, выделенные для кодов причин отказа вместе с описаниями приведены в таблице. Отметим, что значения кодов для удобства даны в десятичном представлении, хотя реально это поле содержит беззнаковые шестнадцатеричные значения типа uint32.

| *Имя* | *Код причины* |
|---|---|
| SSH_OPEN_ADMINISTRATIVELY_PROHIBITED | 1 |
| SSH_OPEN_UNKNOWN_CHANNEL_TYPE | 2 |
| SSH_DISCONNECT_KEY_EXCHANGE_FAILED | 3 |
| SSH_OPEN_RESOURCE_SHORTAGE | 4 |

### *4.3.3. Последующее выделение*

Значения кодов причин отказа Channel Connection Failure reason code **должны** выделяться последовательно. Запросы на выделение новых значений кодов и связанных с ними описаний из диапазона 0x00000005 - 0xFDFFFFFF **должны** обрабатываться методом **согласования с IETF**, как описано в [RFC2434]. IANA не будет выделять значения кодов причины отказа из диапазона 0xFE000000 - 0xFFFFFFFF, оставляя эти значения для **приватного использования**, как описано в [RFC2434].

### *4.3.4. Замечания для диапазона, выделенного для приватных целей*

Хотя IANA не контролирует использование кодов из диапазона 0xFE000000 - 0xFFFFFFFF, этот диапазон разбит на две части и управляется на основе приведённых ниже соглашений.

- Значения кодов от 0xFE000000 до 0xFEFFFFFF используются с локально распределенными каналами. Например, если в предлагаемом канале типа (channel type) example_session@example.com возникает отказ, сервер будет передавать код причины, выделенный IANA (из указанного выше диапазона 0x00000001 – 0xFDFFFFFF) или локально выделенный код из диапазона 0xFE000000 - 0xFEFFFFFF. Естественно, что в тех случаях, когда сервер не понимает предложенное значение channel type, даже если это определённое локально значение типа канала, код причины **должен** иметь значение 0x00000003, как указано выше. Если сервер понимает тип канала, но канал не удаётся открыть, серверу **следует** передавать локально распределенное значение кода причины, согласованное с предложенным локальным значением типа канала. Предполагается, что на практике сначала будет предприниматься попытка использования выделенного IANA значения 'reason code', а потом локально распределенных значений кода причины.
- Для кодов, начинающихся с 0xFF не существует каких-либо ограничений или рекомендаций по использованию. Для этих кодов не предполагается также какой-либо совместимости. Данный диапазон предназначен прежде всего для экспериментальных целей.

## 4.4. data_type_code и data

Значение Extended Channel Data Transfer data_type_code представляет собой 32-битовое целое число без знака (uint32). Связанное с ним текстовое сообщение data предназначено для человека и описывает тип данных, которые могут передаваться в этом канале.

### *4.4.1. Соглашения*

Пакеты протокола, содержащие сообщения SSH_MSG_CHANNEL_EXTENDED_DATA, **должны** иметь значения data_type_code из диапазона 0x00000001 - 0xFFFFFFFF. Описание приведено в документе [SSH-CONNECT].

### *4.4.2. Начальное распределение*

Начальное распределение значений для data_type_code и data приведено в таблице. Отметим, что значение data_type_code для удобства приведено в десятичном представлении, хотя оно имеет тип uint32.

| *Имя* | *data_type_code* |
|---|---|
| SSH_EXTENDED_DATA_STDERR | 1 |

### *4.4.3. Последующее выделение*

Значения Extended Channel Data Transfer data_type_code **должны** выделяться последовательно. Запросы на выделение новых кодов типа данных и связанных с ними описаний (data) из диапазона 0x00000002 – 0xFDFFFFFF **должны** обрабатываться методом **согласования с IETF**, как описано в [RFC2434]. IANA не будет выделять значений data_type_code из диапазона 0xFE000000 – 0xFFFFFFFF, оставляя их для **приватного использования**, как описано в документе [RFC2434].

## 4.5. Терминальные режимы псевдотерминала

Сообщения SSH_MSG_CHANNEL_REQUEST со строкой "pty-req" **должны** содержать encoded terminal modes. Значения терминальных режимов (encoded terminal modes) представляют собой байтовый поток пар «код-аргумент» (opcode-argument).

### *4.5.1. Соглашения*

Пакеты протокола, содержащие сообщение SSH_MSG_CHANNEL_REQUEST со строкой "pty-req", **должны** включать значение encoded terminal modes. Код операции (opcode) представляет собой 1 байт и может принимать значения от 1 до 255. Коды из диапазона 1 - 159 имеют аргумент типа uint32. Коды операций от 160 до 255 пока не определены.

### *4.5.2. Начальное распределение*

В приведённой ниже таблице дано начальное распределение значений кодированных режимов терминала (encoded terminal modes).

| *Код операции* | *Мнемоника* | *Описание* |
|---|---|---|
| 0 | TTY_OP_END | Указывает на завершение опций. |
| 1 | VINTR | Символ прерывания; 255, если не задан. Аналогично и для других символов. Не все символы поддерживаются каждой системой. |
| 2 | VQUIT | Символ завершения (передаёт сигнал SIGQUIT в POSIX-системах). |
| 3 | VERASE | Удалить символ слева от курсора. |
| 4 | VKILL | Удалить текущую строку ввода. |
| 5 | VEOF | Символ завершения файла (передаёт EOF с терминала). |
| 6 | VEOL | Символ завершения строки в дополнение к возврату каретки и/или переводу строки. |
| 7 | VEOL2 | Дополнительный символ завершения строки. |
| 8 | VSTART | Продолжение вывода после паузы (обычно control-Q). |
| 9 | VSTOP | Пауза при выводе (обычно control-S). |
| 10 | VSUSP | Временная остановка текущей программы. |
| 11 | VDSUSP | Другой символ временной остановки. |
| 12 | VREPRINT | Повторная печать текущей строки ввода. |
| 13 | VWERASE | Удаление слова слева от курсора. |
| 14 | VLNEXT | Ввод следующего символа в виде литерала, даже если этот символ имеет специальное значение. |
| 15 | VFLUSH | Символ сброса (очистки) вывода. |
| 16 | VSWTCH | Переключение на другой shell-уровень. |
| 17 | VSTATUS | Печать строки состояния системы (загрузка, команда, PID и т. д.). |
| 18 | VDISCARD | Переключает состояние очистки терминального вывода. |

| Код операции | Мнемоника | Описание |
|---|---|---|
| 30 | IGNPAR | Игнорировать флаг чётности. Для параметра следует устанавливать значение 0, если этот флаг имеет значение FALSE и 1 для случая TRUE. |
| 31 | PARMRK | Маркировать ошибки чётности. и кадрирования. |
| 32 | INPCK | Включить контроль ошибок чётности. |
| 33 | ISTRIP | Исключать (сбрасывать) 8-й бит символов. |
| 34 | INLCR | Отобразить NL в CR для ввода. |
| 35 | IGNCR | Игнорировать CR на вводе. |
| 36 | ICRNL | Отобразить CR в NL для ввода. |
| 37 | IUCLC | Преобразовать символы верхнего регистра в нижний регистр. |
| 38 | IXON | Включить контроль потока для вывода. |
| 39 | IXANY | Любой символ будет вызывать продолжение (restart) после остановки. |
| 40 | IXOFF | Включить контроль потока для ввода. |
| 41 | IMAXBEL | Звуковой сигнал заполнения входной очереди. |
| 50 | ISIG | Разрешить сигналы INTR, QUIT, [D]SUSP. |
| 51 | ICANON | Привести строки ввода в канонический вид. |
| 52 | XCASE | Разрешить ввод и вывод символов верхнего регистра путём указания их эквивалентов из нижнего регистра с префиксом \. |
| 53 | ECHO | Разрешить эхо-вывод. |
| 54 | ECHOE | Визуально удалять символы. |
| 55 | ECHOK | Символ отбрасывания текущей строки. |
| 56 | ECHONL | Выводить NL даже при отключённом ECHO. |
| 57 | NOFLSH | Не выполнять очистку после прерывания. |
| 58 | TOSTOP | Остановить фоновые задания на выводе. |
| 59 | IEXTEN | Включить расширения. |
| 60 | ECHOCTL | Эхо-символы управления как ^(Char). |
| 61 | ECHOKE | Визуальное удаление строки. |
| 62 | PENDIN | Заново напечатать символы ожидающего ввода. |
| 70 | OPOST | Включить обработку вывода. |
| 71 | OLCUC | Преобразовать символы нижнего регистра в верхний регистр. |
| 72 | ONLCR | Отобразить NL в CR-NL. |
| 73 | OCRNL | Преобразовать возврат каретки в перевод строки (для вывода). |
| 74 | ONOCR | Преобразовать перевод строки в возврат каретки и перевод строки (вывод). |
| 75 | ONLRET | Перевод строки вызывает возврат каретки (вывод). |
| 90 | CS7 | 7-битовый режим. |
| 91 | CS8 | 8-битовый режим. |
| 92 | PARENB | Чётность включена. |
| 93 | PARODD | Считать нечётным даже чётное. |
| 128 | TTY_OP_ISPEED | Задает скорость ввода в бит/сек. |
| 129 | TTY_OP_OSPEED | Задает скорость вывода в бит/сек. |

### *4.5.3. Последующее выделение*

Запросы на выделение новых кодов операций и связанных с ними аргументов **должны** обрабатываться методом **согласования с IETF**, как описано в документе [RFC2434].

## 4.6. Имена

В следующих параграфах значения для пространства имён являются текстовыми. В этом разделе приводятся соглашения и инструкции для IANA по выделению имён в будущем. В соответствующих параграфах приводится начальное распределение имён.

### *4.6.1. Соглашения для имён*

Все имена, регистрируемые IANA в соответствии со следующими параграфами, **должны** состоять только из печатаемых символов набора US-ASCII; **недопустимо** включение в имена символов @, запятых (,), пробелов, управляющих символов (коды ASCII до 32 включительно) и символа ASCII с кодом 127 (DEL). Регистр символов в именах принимается во внимание, **недопустимы** имена длиной более 64 символов.

Резервируются возможности использования локальных имён. IANA не будет регистрировать и контролировать имена, содержащие символ @.

Имена с символом @ будут использовать формат name@domainname; собственно именем является часть, расположенная слева от символа @. Формат этой части не задаётся спецификацией, однако она **должна** содержать только печатаемые символы набора US-ASCII; **недопустимо** включение запятых (,), пробелов, символов управления (коды ASCII до 32 включительно) и символов с кодом ASCII 127 (DEL). Символ @ **должен** быть единственным в имени. Часть имени после символа @ **должна** быть корректным полным доменным именем [RFC1034], контролируемым персоной или организацией, определившей имя. Регистр символов в именах принимается во внимание; **недопустимо** создание имён, размер которых превышает 64 символа. Каждый домен волен самостоятельно управлять своим локальным пространством имён. Следует отметить, что такие имена похожи на определяемые в соответствии с STD 11 [RFC0822] адреса электронной почты. Однако это сходство совершенно случайно и не имеет ничего общего с STD 11 [RFC0822]. Примером локально определённого имени может служить ourcipher-cbc@example.com.

### *4.6.2. Последующее выделение имён*

Запросы на выделение новых имён **должны** обрабатываться путём **согласования с IETF**, как описано в [RFC2434].

## 4.7. Имена служб

Имя службы (service name) используется для описания протокольного уровня. Определённые в настоящее время имена указаны в таблице.

| *Имя службы* | *Документ* |
|---|---|
| ssh-userauth | [SSH-USERAUTH] |
| ssh-connection | [SSH-CONNECT] |

## 4.8. Имена методов аутентификации

Название метода аутентификации (Authentication Method Name) используется для описания метода применяемого службой ssh-userauth [SSH-USERAUTH]. В таблице приводятся определённые в настоящее время имена методов аутентификации.

| *Метод* | *Документ* |
|---|---|
| publickey | [SSH-USERAUTH, параграф 7] |
| password | [SSH-USERAUTH, параграф 8] |
| hostbased | [SSH-USERAUTH, параграф 9] |
| none | [SSH-USERAUTH, параграф 5.2] |

## 4.9. Имена, выделенные для протоколов соединений

В приведённых ниже таблицах указаны имена, выделенные для типов каналов протокола соединений (Connection Protocol Type) и запросов этого протокола.

### *4.9.1. Типы каналов протоколов соединений*

В таблице приведены имена, выделенные для типа каналов протокола соединений (Connection Protocol Channel Type).

| *Тип канала* | *Документ* |
|---|---|
| session | [SSH-CONNECT, параграф 6.1] |
| x11 | [SSH-CONNECT, параграф 6.3.2] |
| forwarded-tcpip | [SSH-CONNECT, параграф 7.2] |
| direct-tcpip | [SSH-CONNECT, параграф 7.2] |

### *4.9.2. Названия запросов протокола соединений*

В таблице приведены имена запросов протокола соединений (Connection Protocol Global Request Name).

| *Тип запроса* | *Документ* |
|---|---|
| tcpip-forward | [SSH-CONNECT, параграф 7.1] |
| cancel-tcpip-forward | [SSH-CONNECT, параграф 7.1] |

### *4.9.3. Названия запросов канала протокола соединений*

В таблице приводятся выделенные значения имён запросов канала протокола соединений (Connection Protocol Channel Request Name).

| *Тип запроса* | *Документ* |
|---|---|
| pty-req | [SSH-CONNECT, параграф 6.2] |
| x11-req | [SSH-CONNECT, параграф 6.3.1] |
| env | [SSH-CONNECT, параграф 6.4] |
| shell | [SSH-CONNECT, параграф 6.5] |
| exec | [SSH-CONNECT, параграф 6.5] |
| subsystem | [SSH-CONNECT, параграф 6.5] |
| window-change | [SSH-CONNECT, параграф 6.7] |
| xon-xoff | [SSH-CONNECT, параграф 6.8] |
| signal | [SSH-CONNECT, параграф 6.9] |
| exit-status | [SSH-CONNECT, параграф 6.10] |
| exit-signal | [SSH-CONNECT, параграф 6.10] |

### *4.9.4. Начальное распределение имён для сигналов*

В таблице приведены имена, выделенные для сигналов (Signal Name).

| *Сигнал* | *Документ* |
|---|---|
| ABRT | [SSH-CONNECT] |
| ALRM | [SSH-CONNECT] |
| FPE | [SSH-CONNECT] |
| HUP | [SSH-CONNECT] |
| ILL | [SSH-CONNECT] |
| INT | [SSH-CONNECT] |
| KILL | [SSH-CONNECT] |
| PIPE | [SSH-CONNECT] |
| QUIT | [SSH-CONNECT] |
| SEGV | [SSH-CONNECT] |
| TERM | [SSH-CONNECT] |
| USR1 | [SSH-CONNECT] |
| USR2 | [SSH-CONNECT] |

### *4.9.5. Имена подсистем протокола соединений*

Для имён подсистем протокола соединений (Connection Protocol Subsystem Name) начальные значения не выделены.

## 4.10. Имена методов обмена ключами

Имя diffie-hellman-group1-sha1 служит для метода обмена ключами, использованного Oakley в соответствии с [RFC2409]. SSH поддерживает своё пространство идентификаторов групп, которое логически отличается от Oakley [RFC2412] и IKE. Однако для одной дополнительной группы было адаптировано значение номера, выделенного в [RFC3526], с использованием diffie-hellman-group14-sha1 в качестве имени второй определённой группы. Реализациям

следует трактовать эти имена, как простые идентификаторы и не следует предполагать наличие каких-либо связей между группами, используемыми SSH, и группами, определёнными для IKE.

В таблице приведены имена, выделенные для методов обмена ключами.

| *Метод* | *Документ* |
|---|---|
| diffie-hellman-group1-sha1 | [SSH-TRANS, параграф 8.1] |
| diffie-hellman-group14-sha1 | [SSH-TRANS, параграф 8.2] |

## 4.11. Выделенные имена алгоритмов

### *4.11.1. Имена алгоритмов шифрования*

В таблице приведены имена, выделенные для алгоритмов шифрования (Encryption Algorithm Name).

| *Алгоритм* | *Документ* |
|---|---|
| 3des-cbc | [SSH-TRANS, параграф 6.3] |
| blowfish-cbc | [SSH-TRANS, параграф 6.3] |
| twofish256-cbc | [SSH-TRANS, параграф 6.3] |
| twofish-cbc | [SSH-TRANS, параграф 6.3] |
| twofish192-cbc | [SSH-TRANS, параграф 6.3] |
| twofish128-cbc | [SSH-TRANS, параграф 6.3] |
| aes256-cbc | [SSH-TRANS, параграф 6.3] |
| aes192-cbc | [SSH-TRANS, параграф 6.3] |
| aes128-cbc | [SSH-TRANS, параграф 6.3] |
| serpent256-cbc | [SSH-TRANS, параграф 6.3] |
| serpent192-cbc | [SSH-TRANS, параграф 6.3] |
| serpent128-cbc | [SSH-TRANS, параграф 6.3] |
| arcfour | [SSH-TRANS, параграф 6.3] |
| idea-cbc | [SSH-TRANS, параграф 6.3] |
| cast128-cbc | [SSH-TRANS, параграф 6.3] |
| none | [SSH-TRANS, параграф 6.3] |
| des-cbc | [FIPS-46-3] HISTORIC; см. стр. 4 документа [FIPS-46-3] |

### *4.11.2. Имена алгоритмов MAC*

В таблице указаны имена, выделенные для алгоритмов MAC (MAC Algorithm Name).

| *Алгоритм* | *Документ* |
|---|---|
| hmac-sha1 | [SSH-TRANS, параграф 6.4] |
| hmac-sha1-96 | [SSH-TRANS, параграф 6.4] |
| hmac-md5 | [SSH-TRANS, параграф 6.4] |
| hmac-md5-96 | [SSH-TRANS, параграф 6.4] |
| none | [SSH-TRANS, параграф 6.4] |

### *4.11.3. Имена алгоритмов открытых ключей*

В таблице приведены имена, выделенные для алгоритмов открытых ключей (Public Key Algorithm).

| *Алгоритм* | *Документ* |
|---|---|
| ssh-dss | [SSH-TRANS, параграф 6.6] |
| ssh-rsa | [SSH-TRANS, параграф 6.6] |
| pgp-sign-rsa | [SSH-TRANS, параграф 6.6] |
| pgp-sign-dss | [SSH-TRANS, параграф 6.6] |

### *4.11.4. Имена алгоритмов компрессии*

В таблице перечислены имена, выделенные для алгоритмов сжатия данных (Compression Algorithm).

| *Алгоритм* | *Документ* |
|---|---|
| none | [SSH-TRANS, параграф 6.2] |
| zlib | [SSH-TRANS, параграф 6.2] |

# *5. Вопросы безопасности*

Данный протокол обеспечивает поддержку защищённых шифрованных соединений через сети, не обеспечивающие защиты. Полное рассмотрение вопросов безопасности для этого протокола содержится в документе [SSH-ARCH].

# *6. Литература*

## 6.1. Нормативные документы


[SSH-ARCH] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Protocol Architecture", RFC 4251, January 2006.

[SSH-TRANS] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Transport Layer Protocol", RFC 4253, January 2006.

[SSH-USERAUTH] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Authentication Protocol", RFC 4252, January 2006.

[SSH-CONNECT] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Connection Protocol", RFC 4254, January 2006.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[RFC2434] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[RFC3526] Kivinen, T. and M. Kojo, "More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)", RFC 3526, May 2003.

## 6.2. Дополнительная литература

[RFC0822] Crocker, D., "Standard for the format of ARPA Internet text messages", STD 11, RFC 822, August 1982.

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, November 1987.

[RFC2412] Orman, H., "The OAKLEY Key Determination Protocol", RFC 2412, November 1998.

[FIPS-46-3] US National Institute of Standards and Technology, "Data Encryption Standard (DES)", Federal Information Processing Standards Publication 46-3, October 1999.

# *Адреса авторов*

**Sami Lehtinen**
SSH Communications Security Corp
Valimotie 17
00380 Helsinki
Finland
EMail: sjl@ssh.com

**Chris Lonvick** (редактор)
Cisco Systems, Inc.
12515 Research Blvd.
Austin 78759
USA
EMail: clonvick@cisco.com

**Перевод на русский язык**

Николай Малых

nmalykh@protokols.ru

# *Торговые марки*

ssh – торговый знак, зарегистрированный в США и/или других странах.

**Полное заявление авторских прав**

**Copyright (C) The Internet Society (2006).**

К этому документу применимы права, лицензии и ограничения, указанные в BCP 78, и, за исключением указанного там, авторы сохраняют свои права.

Этот документ и содержащаяся в нем информация представлены "как есть" и автор, организация, которую он/она представляет или которая выступает спонсором (если таковой имеется), Internet Society и IETF отказываются от каких-либо гарантий (явных или подразумеваемых), включая (но не ограничиваясь) любые гарантии того, что использование представленной здесь информации не будет нарушать чьих-либо прав, и любые предполагаемые гарантии коммерческого использования или применимости для тех или иных задач.

**Интеллектуальная собственность**

IETF не принимает какой-либо позиции в отношении действительности или объема каких-либо прав интеллектуальной собственности (Intellectual Property Rights или IPR) или иных прав, которые, как может быть заявлено, относятся к реализации или использованию описанной в этом документе технологии, или степени, в которой любая лицензия, по которой права могут или не могут быть доступны, не заявляется также применение каких-либо усилий для определения таких прав. Сведения о процедурах IETF в отношении прав в документах RFC можно найти в BCP 78 и BCP 79.

Копии раскрытия IPR, предоставленные секретариату IETF, и любые гарантии доступности лицензий, а также результаты попыток получить общую лицензию или право на использование таких прав собственности разработчиками или пользователями этой спецификации, можно получить из сетевого репозитория IETF IPR по ссылке http://www.ietf.org/ipr.

IETF предлагает любой заинтересованной стороне обратить внимание на авторские права, патенты или использование патентов, а также иные права собственности, которые могут потребоваться для реализации этого стандарта. Информацию следует направлять в IETF по адресу ietf-ipr@ietf.org.

**Подтверждение**

Финансирование функций RFC Editor обеспечено IETF Administrative Support Activity (IASA).