



Маршрутизаторы NSG

Программное обеспечение NSG Linux 2.0

Руководство пользователя

Часть 1

Общесистемная конфигурация

Версия программного обеспечения 2.0 build 7

Обновлено 02.06.2016

АННОТАЦИЯ

Данный документ содержит руководство по настройке и применению маршрутизаторов NSG, оснащенных программным обеспечением NSG Linux 2.0. Документ имеет следующую структуру:

- Часть 1. Общесистемная конфигурация.
- Часть 2. Настройка физических интерфейсов, портов и сетевых интерфейсов. Обработка трафика Ethernet.
- Часть 3. Обработка IP-трафика.
- Часть 4. Приложения и службы IP.
- Часть 5. Туннелирование и виртуальные частные сети (VPN).
- Часть 6. Система обеспечения бесперебойных соединений **uITCP**.
- Часть 7. Основные команды и утилиты NSG Linux.

Руководства по применению маршрутизаторов под управлением NSG Linux 1.0 и базового программного обеспечения NSG, а также других продуктов NSG (модемов, мостов и т.п.), содержатся в отдельных документах.

ВНИМАНИЕ

Данное Руководство пользователя предназначено для лучшего понимания процедуры настройки в целом и описывает суть выполняемых действий — а именно, что необходимо настраивать. Рассматриваемые вопросы относятся, как правило, к сути используемой технологии и являются общими для любых её реализаций, независимо от конкретного производителя и устройства. (Исключением являются вопросы, специфичные для оборудования NSG — таких, как организация пользовательского интерфейса или настройка системы бесперебойных соединений **uITCP**.)

Основной документацией по NSG Linux 2.0 является встроенная справка на борту устройства. Она описывает конкретные команды и параметры настройки — т.е. как настраивать функции и возможности, описанные в данном Руководстве. Для просмотра справки по каждому из параметров следует использовать кнопку  в Web-интерфейсе или команду `_manual (_m)` в консольном интерфейсе. Если справка на вашем языке отсутствует, следует установить на устройстве русскую локаль.

ВНИМАНИЕ Продукция компании непрерывно совершенствуется, в связи с чем возможны изменения отдельных аппаратных и программных характеристик по сравнению с настоящим описанием. Сведения о последних изменениях приведены в файлах README.TXT, CHANGES, а также в документации на отдельные устройства.

Замечания и комментарии по документации NSG принимаются по адресу: doc@nsg.net.ru.

© ООО «Эн-Эс-Джи» 2009–2016

ООО «Эн-Эс-Джи»
Россия 105187 Москва
ул. Вольная, д.35
Тел./факс: (+7-495) 727-19-59 (многоканальный)

<http://www.nsg.ru/>
<mailto:info@nsg.net.ru>
<mailto:sales@nsg.net.ru>
<mailto:support@nsg.net.ru>

§ СОДЕРЖАНИЕ §

Часть 1. Общесистемная конфигурация

§1.1. Введение.....	5
§1.1.1. Назначение и общая характеристика системы.....	5
§1.1.2. Структура программного обеспечения NSG Linux 2.0	5
§1.1.3. Конфигурация и средства управления NSG Linux 2.0	6
§1.1.4. Конфигурационные файлы	7
§1.1.5. Пользователи системы и их командные оболочки	7
§1.1.6. Документация по системе.....	8
§1.2. Начало работы с системой.....	9
§1.2.1. Подключение к устройству через порт Ethernet	9
§1.2.2. Подключение к устройству через консольный порт	9
§1.2.3. Просмотр и поиск справки	10
§1.2.4. Командная оболочка ОС Linux	10
§1.2.5. Множественный доступ к системе.....	10
§1.2.6. Формат записи конфигурации	11
§1.2.7. Разовые команды	12
§1.2.8. Типы данных	12
§1.2.9. Уникальные объекты, именованные и нумерованные списки	13
§1.2.10. Проверка, применение и сохранение конфигурации	14
§1.2.11. Зависимые параметры	14
§1.2.12. Явно указанные и расширенные параметры	15
§1.2.13. Отладка системы, системные журналы	15
§1.2.14. Завершение работы с системой	16
§1.3. Командная оболочка NSG Shell	17
§1.3.1. Общие сведения.....	17
§1.3.2. Команды обзора и перемещения по дереву конфигурации	17
§1.3.3. Команды построения дерева конфигурации	18
§1.3.4. Простой ввод параметров и разовых команд	19
§1.3.5. Сложный ввод параметров и разовых команд	20
§1.3.6. Специальные и итоговые команды	22
§1.3.7. Соглашения и сокращения.....	22
§1.3.8. Исполнение nsgsh из командной строки Linux	22
§1.3.9. Экранный режим работы nsgsh	23
§1.4. Web- и SNMP-управление	25
§1.4.1. Web-интерфейс	25
§1.4.2. SNMP-управление	26
§1.5. Общесистемная конфигурация.....	27
§1.5.1. Имя устройства	27
§1.5.2. Установка системного времени.....	27
§1.5.3. Управление локальными пользователями.....	27
§1.5.4. Настройка средств управления.....	29
§1.5.5. Настраиваемая светодиодная индикация	29
§1.5.6. Резервное копирование и восстановление текущей конфигурации.....	29
§1.5.7. Пользовательские конфигурации.....	31
§1.5.8. Очистка конфигурации	31
§1.5.9. Автоматическое копирование конфигурации между устройствами	31
§1.5.10. Восстановление заводской конфигурации и модернизация программного обеспечения.....	32
§1.5.11. Проверка обновлений программного обеспечения	33
§1.5.12. Перегрузка устройства.....	33
§1.5.13. Сторожевой таймер	33
§1.6. Виртуальные маршрутизаторы	34
§1.6.1. Общие сведения о механизме виртуальных маршрутизаторов.....	34
§1.6.2. Создание виртуальных маршрутизаторов.....	34
§1.6.3. Вход в виртуальный маршрутизатор и особенности его настройки.....	35

§1.7. Безопасность устройства	36
§1.7.1. Общие рекомендации	36
§1.7.2. Аутентификация пользователей	37
§1.7.3. Аудит системы	37
§1.8. Сервисный режим	38
§1.8.1. Общие операции в сервисном режиме	38
§1.8.2. Расширенные настройки сервисного режима	40
§1.8.3. Установка NSG Linux 2.0 на устройства NSG-700.....	42
§1.8.4. Обновление ПО сервисного режима	45
Приложение 1–А. Настройка UTF-8 в консольных программах.....	47
§1–А.1. TeraTerm 4.6x.....	47
§1–А.2. PuTTY 0.60	48
Приложение 1–Б. Корректировка команд при переходе на новую версию NSG Linux 2.0.....	49

§1.1. Введение

§1.1.1. Назначение и общая характеристика системы

Программное обеспечение NSG Linux 2.0 предназначено для маршрутизаторов NSG и реализует современные технологии IP-сетей, такие как VPN, VLAN, протоколы динамической маршрутизации, механизмы QoS, различные варианты туннелирования, мультипротокольной инкапсуляции и VPN, коммутацию канального уровня Ethernet (*bridging*). Основное назначение устройств NSG состоит в построении сетей коммутации пакетов на основе протокола IP, в том числе виртуальных частных сетей (VPN), а также территориально-распределенных сетей Ethernet.

Программное обеспечение NSG Linux 2.0 является штатным для следующих продуктов NSG:

- NSG–600, NSG–605 — CPU Infineon 5120P, поддерживается до версии NSG Linux 2.0.x включительно
- NSG–1800, NSG–1820 — CPU семейства Freescale MPC83xx
- NSG–1000e, NSG–1000/GW — CPU с архитектурой x86

Опционально NSG Linux 2.0 может быть использовано также на следующих сериях устройств:

- NSG–700/4AU *h/w ver.5* и выше¹, NSG–700/8A, NSG–700/12A — CPU Atmel AT91RM9200

В состав системы входят как программные модули, свободно распространяемые на принципах Open Source, так и собственные разработки компании NSG, некоторые из которых могут являться коммерческими продуктами. Программное обеспечение, в соответствии с общепринятой практикой, поставляется в виде "как есть" (*as is*).

Функциональные возможности NSG Linux, а также поддерживаемые стандарты и спецификации, перечислены в документе NSG:

Программное обеспечение NSG

§1.1.2. Структура программного обеспечения NSG Linux 2.0

Программное обеспечение NSG Linux представляет собой интегрированный комплекс на основе ОС Linux, предназначенный для выполнения задач маршрутизации пакетов IP на аппаратной платформе устройств NSG. В состав комплекса входят следующие основные программные компоненты:

- Ядро Linux, необходимым образом доработанное и скомпилированное для работы на процессорах используемого типа.
- Драйверы для интерфейсных модулей и других специфических аппаратных компонент NSG, интегрированные в ядро.
- Набор прикладных пакетов и приложений, предназначенных для маршрутизации/коммутации пакетов, фильтрации, приоритизации трафика и решения других задач. Прикладные компоненты доработаны специалистами NSG для корректной работы в составе единого программно-аппаратного комплекса, с общими средствами конфигурирования и мониторинга.
- Фирменная система бесперебойных соединений **uTCP** — VPN четвертого уровня, ориентированная на гарантированную доставку данных в банковских и других приложениях при множественных переключениях между двумя и более альтернативными каналами связи.
- Системные компоненты NSG для управления приложениями.
- Утилиты и средства управления устройством: Telnet, SSH, ash, nsgsh, SNMP agent, Web-сервер (см. §1.1.3).

Помимо основного ПО NSG Linux 2.0, пакет программного обеспечения включает системный загрузчик и отдельную копию Linux для работы в сервисном режиме (Recovery Linux). Эти компоненты хранятся на отдельных разделах энергонезависимой памяти и не используются пользователем в процессе нормальной работы устройства. Сервисный режим обеспечивает обновление основного ПО, а также восстановление заводской конфигурации в случае неработоспособности или неработоспособной конфигурации основного ПО, утраты пароля и т.п. ситуаций

Конфигурация системы хранится вместе с основным ПО. При переустановке основного программного обеспечения штатными средствами на рабочей системе она сохраняется. При переустановке в сервисном режиме раздел с основным ПО перезаписывается полностью и прежняя конфигурация безусловно утрачивается.

¹ Кроме устройств ранних выпусков, не оборудованных программируемой кнопкой Cold Start (CS).

§1.1.3. Конфигурация и средства управления NSG Linux 2.0

Набор настроек в NSG Linux 2.0 имеет древовидную структуру, узлами и листьями которой являются те или иные параметры или команды. Количество промежуточных узлов дерева не ограничено. Для краткости, командное дерево далее по тексту, как правило, называется "меню".

Параметры хранятся в устройстве постоянно и определяют его конфигурацию.

Команды выполняют разовые действия (например, вывод статистики и состояния текущего объекта, генерацию ключей и т.п.) и не сохраняются в устройстве. Параметры, с которыми вызываются команды (например, адрес для *ping*), хранятся индивидуально для каждого пользователя в пределах сеанса его работы в отдельной секции Local Configuration конфигурационного дерева.

Пользовательские настройки системы хранятся в файле `/etc/nsgconfig`. Он является простым текстовым файлом, однако имеет строго определённую структуру и синтаксис. По этой причине прямое редактирование данного файла настоятельно не рекомендуется. Ошибки, весьма вероятные при ручном редактировании, во многих случаях могут привести к тому, что конфигурация окажется неработоспособной и система не сможет стартовать. Для работы с деревом конфигурации в NSG Linux 2.0 предусмотрены два инструмента:

Командная оболочка `nsgsh` работает в интерактивном текстовом режиме и доступна через по Telnet, SSH, или через консольный порт (при наличии такового). Оболочка обеспечивает просмотр и перемещение по дереву конфигурации, добавление и удаление узлов и листьев дерева, ввод и редактирование листьев дерева — т.е. настройку параметров системы и исполнение команд. Кроме того, она содержит исчерпывающее описание команд и параметров на русском языке. Подробно о работе с `nsgsh` см. §1.3.

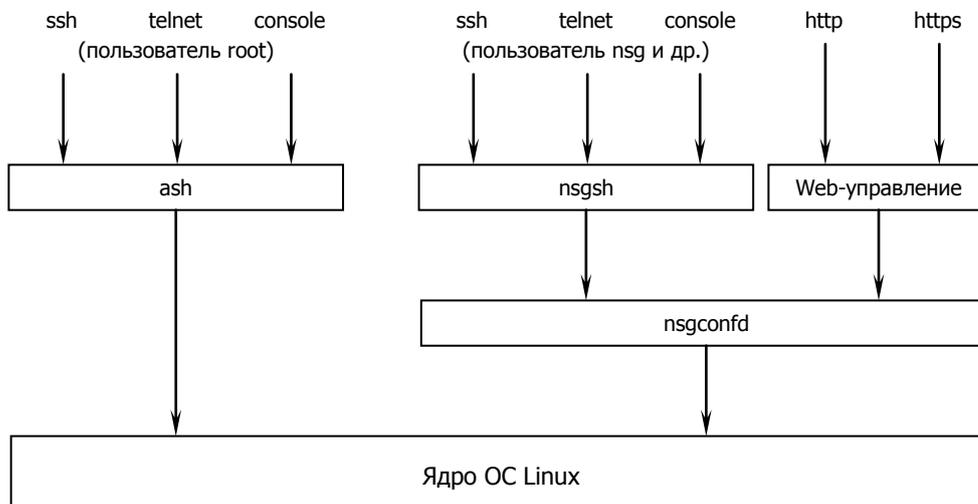
Web-сервер предназначен для управления устройством посредством протоколов HTTP или HTTPS. Сервер обеспечивает все те же операции с деревом конфигурации, а также визуальное представление этого дерева. Это освобождает администратора системы от необходимости держать структуру дерева в уме, что значительно упрощает его работу. Для Web-управления может быть использован любой современный графический браузер с поддержкой Javascript, соответствующий стандартам W3C. Подробно о Web-управлении см. §1.4.

ПРИМЕЧАНИЕ Использование Web-управления не освобождает от необходимости понимания используемых сетевых технологий, сути выполняемых команд и настраиваемых параметров.

В качестве основного инструмента управления предлагается Web-управление через порт Ethernet. Оболочка `nsgsh` предназначена, в основном, для эпизодического управления удалённым устройством по медленным каналам связи, таким как GPRS, или при высокой нагрузке, когда дополнительный расход вычислительных ресурсов для работы Web-сервера нецелесообразен.

Благодаря наличию вышеперечисленных инструментов, администрирование устройств NSG не требует знания операционной системы Linux как таковой. Тем не менее, знакомство с основными командами ОС Linux желательно, как минимум, для получения отладочной информации в сложных ситуациях. Описание основных команд Linux, которые могут потребоваться пользователю, а также фирменных утилит NSG, содержится в [Части 7](#) данного руководства.

Системное приглашение `nsgsh` содержит административное имя устройства и полный путь к текущему узлу конфигурации, например, `hostname: port.eth0.ifAddress>`. Для удобства описания, во всей документации по NSG Linux 2.0 названия узлов приводятся, в аналогичном виде; корень дерева обозначается точкой, например, `.port.eth0.ifAddress`.



Командная оболочка `ash` является стандартной компонентой ОС Linux и предназначена для пользователей, обладающих необходимыми знаниями в этой области. Кроме того, маршрутизаторы NSG можно рассматривать как универсальную Linux-машину со специфическими аппаратными возможностями. На устройстве имеется набор общепринятых утилит и инструментов (`grep`, `gzip` и др.), а также текстовый редактор `nano`. В частности, пользователь может устанавливать на эту машину дополнительные программные компоненты, доступные в исходных кодах или разработанные самостоятельно, писать скрипты Linux и т.п. Все пользовательские компоненты следует устанавливать только в директорию `/etc`, либо на дополнительную энергонезависимую память (`flash`, `HDD`) при наличии таковой.

Встроенный *SNMP-агент* обеспечивает управление устройством с помощью любой стандартной системы или утилит SNMP-управления. Агент поддерживает SNMP v1, v2 и v3. В данной версии NSG Linux 2.0 SNMP-управление реализовано только в рамках стандартных MIB II и не предусматривает использования фирменных MIB. Подробно об SNMP-управлении см. [Часть 4](#).

SMS-управление представляет собой фирменную разработку NSG и позволяет исполнять ограниченный набор команд, предварительно заданный администратором, при помощи мобильного телефона. Приложение имеет дружелюбный интерфейс пользователя и работает на большинстве современных мобильных телефонов GSM и 3G (UMTS) с поддержкой Java. Для SMS-управления устройство NSG должно быть оснащено интерфейсным модулем или встроенным интерфейсом GSM/GPRS/EDGE или 3G. Подробно об SMS-управлении см. [Часть 4](#). Кроме того, возможна отправка текстовых SMS-оповещений из скриптов Linux (подробно см. [Часть 7](#)).

ПРИМЕЧАНИЕ Оболочка `nsgsh` и Web-сервер являются, по существу, пользовательскими интерфейсами к демону `nsgconfd`. Демон работает постоянно и поддерживает заданную конфигурацию устройства. Если изменять её средствами, находящимися вне `nsgconfd` — например, командами оболочки `ash`, запускаемыми вручную или из каких-либо скриптов (`netping`, `SMS-управление` и т.п.), или по SNMP, то `nsgconfd` будет обнаруживать эти изменения и восстанавливать свою конфигурацию. Эту особенность необходимо учитывать при написании скриптов ОС Linux (см. [Часть 7](#)).

§1.1.4. Конфигурационные файлы

Конфигурационное дерево системы хранится в текстовом файле `/etc/nsgconfig`. Это основной набор параметров, определяющий настройку устройства.

Помимо указанного файла, полная конфигурация устройства включает в себя также ряд других файлов, хранящихся отдельно в директории `/etc`, в частности:

- Пароли и хэши для различных целей, в частности, пароли пользователей.
- Сертификаты X.509.
- Пользовательские скрипты, приложения и файлы конфигурации.
- Системные файлы и директории, изменённые пользователем (например, `inittab` или дополнительные стартовые скрипты в директории `/etc/init.d`).

Для сохранения и восстановления полной конфигурации следует использовать команды и утилиты, предусмотренные для этого в командной оболочке и Web-интерфейсе. Они упаковывают директорию `/etc` целиком и позволяют экспортировать/импортировать её в виде единого файла. По этой же причине все пользовательские файлы (сертификаты, скрипты и др.) рекомендуется хранить также внутри директории `/etc`, чтобы они автоматически попадали в резервную копию. Временные файлы (логи и т.п.) следует записывать в директорию `/tmp` или `/var`.

§1.1.5. Пользователи системы и их командные оболочки

В системе всегда существуют два системных пользователя: `root` и `nsg`. Различия между ними заключаются в следующем:

- Только пользователь `root` имеет право работать в командной оболочке `ash`. Пользователь `nsg` может работать только в оболочке `nsgsh` и Web-интерфейсе.
При входе в систему по Telnet, SSH или через консольный порт для пользователя `root` всегда запускается `ash`, а для пользователя `nsg` — оболочка `nsgsh`. В `nsgsh` только пользователю `root` доступна команда `_tty` для запуска вложенной оболочки `ash`.
- Только пользователь `root` имеет право принудительно завершать работу пользователя, который в данный момент является администратором, т.е. осуществляет управление устройством с правами чтения и записи (подробнее см. §1.2.5).

При входе в систему через Web-интерфейс, а также при запуске `nsgsh` из-под `ash`, оба эти пользователя имеют доступ с правами чтения и записи ко всему дереву команд.

Помимо этих двух пользователей, администратор системы может создавать неограниченное число других пользователей с произвольными именами. При консольном входе в систему для этих пользователей также запускается оболочка `nsgsh`. Для каждого пользователя администратором создаётся индивидуальное меню, состоящее из ссылок на узлы полного командного дерева. По умолчанию, это дерево пустое, т.е. вновь созданному пользователю не доступны никакие команды.

Пользователи могут иметь права чтения и записи (`read-write`), или только чтения (`read-only`). Узлы, не входящие в индивидуальное меню пользователя, невидимы для него как в `nsgsh`, так и в Web-интерфейсе. Кроме того, для дополнительных пользователей также недоступна оболочка `ash`.

§1.1.6. Документация по системе

Web-интерфейс NSG Linux 2.0 содержит подробную встроенную справку по каждой команде, параметру и узлу меню на русском языке, а также краткую подсказку на русском и английском языках. Таким образом, система является самодокументированной и не нуждается в описании команд на бумажном носителе. По этой причине, а также с целью сохранения лесного покрова Земли, представление описания команд в печатном виде представляется нецелесообразным.

ВНИМАНИЕ Для просмотра подробной справки в данной версии NSG Linux 2.0 требуется установить русскую локаль, поскольку справка на других языках отсутствует.

Структура исходного кода NSG Linux 2.0 предусматривает возможность перевода справки на другие языки.

Данное Руководство пользователя является дополнением к встроенной справке и уделяет основное внимание следующим аспектам:

- Вход и первичное знакомство с системой.
- Работа в командной оболочке `nsgsh`.
- Основы работы в Web-интерфейсе.
- Управление пользователями и пользовательскими меню.
- Существо операций, выполняемых при настройке тех или иных программных компонент и объектов, смысл настраиваемых при этом параметров. Описание приводится в стиле "дорожной карты" (*roadmap*) для процедуры настройки в целом. Предполагается, что администратор устройства знаком с используемыми сетевыми технологиями, а подробное ознакомление с каждым из параметров по отдельности производится в Web-интерфейсе.
- Выполнение операций по восстановлению системы в сервисном режиме.
- Установка NSG Linux 2.0 на устройства под управлением NSG Linux 1.0.
- Возможные особенности перехода на новые версии NSG Linux 2.0 (отдельные случаи несовместимости с существующими конфигурациями, и т.п.)
- Основы работы в ОС Linux для администраторов устройств NSG.

Документация по продуктам с открытым кодом, использованным в составе NSG Linux, доступна на Web-сайтах их разработчиков, соответственно. Описания команд и основных приложений ОС Linux имеются в общей документации по ОС Linux. Данное руководство не имеет своей целью их дублирование. Наиболее распространенным форматом для их исчерпывающего описания являются так называемые страницы руководств (*man pages*), входящие в состав большинства дистрибутивов Linux и доступные на многих интернет-ресурсах, например:

<http://www.opennet.ru/man.shtml> (на русском и английском языках, но некоторые разделы могут быть устаревшими)

<http://linux.die.net/man/> (на английском языке, более актуальные версии некоторых страниц)

Из прочих информационных материалов, посвящённых отдельным аспектам использования ОС Linux, следует упомянуть:

Bert Hubert et al. Linux Advanced Routing & Traffic Control HOWTO — исчерпывающее руководство по обработке IP-трафика в Linux. Доступно на многих интернет-ресурсах.

<http://tldp.org> — The Linux Documentation Project. Весьма подробные и доступные руководства по отдельным вопросам, включая общетеоретическое введение в предмет.

<http://ru.wikipedia.org/wiki/Iptables> — доступное описание возможностей пакета `IPtables` и механизма `netfilter` на русском языке.

§1.2. Начало работы с системой

§1.2.1. Подключение к устройству через порт Ethernet

Все устройства NSG, работающие под управлением NSG Linux 2.0, оснащены одним или несколькими встроенными портами Fast Ethernet. Для выполнения первичной настройки порту eth0 в заводской конфигурации устройства назначен IP-адрес 192.168.1.1/24. Для устройств, оснащённых встроенным коммутатором Ethernet на данном порту процессора, подключение может производиться к любому порту коммутатора, поскольку в заводской конфигурации он настроен как простой коммутатор без VLAN и логического разделения портов. Требуемый тип кабеля Ethernet для различных устройств:

- для подключения NSG–1000e, NSG–1000/GW к ПК — кроссовый (зелёный NSG)
- для подключения NSG–1000e, NSG–1000/GW к коммутатору ЛС — прямой (синий или серый NSG)
- для остальных моделей, поддерживающих NSG Linux 2.0 — несущественно.

Сетевому адаптеру ПК необходимо назначить IP-адрес, принадлежащий к вышеуказанной сети, т.е. любой другой адрес вида 192.168.1.x с маской 255.255.255.0. Для доступа в консольном режиме следует использовать стандартный номер порта TCP 23 и любой клиент Telnet на ПК. Будет выведено приглашение

```
nsg login:
```

Необходимо ввести имя пользователя `nsg`. Пароль по умолчанию для данного пользователя не установлен и не запрашивается. При правильном вводе имени и пароля пользователь входит в командную оболочку `nsgsh`.

Для Web-управления может быть использован любой современный графический браузер с поддержкой Javascript, cookies и CSS, соответствующий стандартам W3C. Доступ к Web-интерфейсу осуществляется по протоколу HTTP через стандартный порт TCP 80. После подключения к устройству выводится страница аутентификации, содержащая два окна для ввода имени пользователя и пароля. Следует ввести имя `nsg` и оставить поле пароля пустым. При правильном вводе имени и пароля пользователь попадает на страницу управления устройством.

Другие способы доступа к устройству (HTTPS, SSH, SNMP, вход с именем `root` или иным именем пользователя) в заводской конфигурации выключены и становятся доступными после соответствующей настройки пользователем. После настройки доступ к устройству возможен также через любой порт по любому IP-адресу, назначенному данному устройству (если это не запрещено фильтрами), или через порт RS–232, имеющий тип инкапсуляции `login`.

§1.2.2. Подключение к устройству через консольный порт

В данной версии NSG Linux 2.0 не предусмотрено наличие выделенного консольного порта в полном смысле этого термина, т.е. выполняющего две функции:

- вывод отладочных сообщений ядра
- вход в систему в терминальном текстовом режиме

Отладочные сообщения могут быть просмотрены, в случае необходимости, другими средствами (командой `dmesg`). Для терминального входа в систему можно сконфигурировать любой порт RS–232, имеющийся на устройстве; для этой цели ему необходимо назначить инкапсуляцию `login` и, при необходимости, параметры физического уровня (по умолчанию 9600 бит/с, 8n1, Hardware Flow Control; рекомендуется 115200 бит/с). Подробно о настройке асинхронных портов RS–232 см. [Часть 2](#).

В большинстве устройств NSG консольный порт имеет разъём RJ–45 и назначение контактов, совпадающее с таковым в устройствах Cisco Systems. Для подключения к COM-порту ПК или к модему следует использовать кабель NSG CAS–V24/D9/FC/A (голубой), или аналогичные кабели и адаптеры Cisco. Исключением являются устройства серии NSG–1000, оснащённые стандартным разъёмом DB–9f. Для их подключения к ПК используется стандартный нуль-модемный кросс-кабель (DB–9f/DB–9f). Если устройство не имеет вообще никаких интерфейсов RS–232, то консольный порт в нём может быть организован с помощью переходника USB–COM. Распайка всех кабелей приведена в Руководствах пользователя по соответствующим устройствам, а также в документе NSG:

Маршрутизаторы NSG–700, NSG–1800. Интерфейсные модули. Серия UM–xxx. Руководство пользователя.

Для работы с устройством можно использовать любую программу эмуляции терминала. В процессе загрузки устройства на данный порт, в отличие от полнофункционального консольного порта, не выводится ничего; по завершении загрузки будет выведено приглашение

```
nsg login:
```

Дальнейшая работа в консольном интерфейсе идентична работе по Telnet.

В отдельных моделях устройств и версиях ПО предусмотрена настройка консольного порта в меню сервисного режима. Подробнее об этой настройке см. §1.8.2. В этом случае порт используется исключительно в качестве консольного и не может быть перенастроен для передачи пользовательских данных.

ПРИМЕЧАНИЕ Функциональность консольного порта на некоторых устройствах (NSG-700, NSG-1800) изменена описанным выше образом, начиная с версии 2.0 *build 3*. В устройствах, оснащённых устаревшей версией ПО сервисного режима, могут наблюдаться следующие аномалии:

- в процессе загрузки и работы устройства в порт выводятся отладочные сообщения;
- порт не может быть сконфигурован никаким иным образом.

При наличии указанных симптомов следует обновить ПО сервисного режима. Обновление выполняется в рабочем режиме с помощью *service pack*. Подробно о данном обновлении см. §1.8.4.

§1.2.3. Просмотр и поиск справки

NSG Linux 2.0 имеет встроенную краткую подсказку на русском и английском языках, а также подробную справку на русском языке. Для просмотра подсказки в консольной оболочке *nsgsh* введите команду `_help`, в Web-интерфейсе наведите указатель мыши на имя параметра или узла дерева. Для просмотра подробной справки введите команду `_manual` или нажмите кнопку .

ВНИМАНИЕ Поскольку в текущей версии NSG Linux 2.0 имеется справка только на русском языке, для просмотра справки необходимо установить русскую локаль (на данный сеанс или по умолчанию на все сеансы для данного пользователя). Для просмотра справки в консольном интерфейсе необходимо, помимо этого, использовать эмулятор терминала, клиент Telnet или SSH с поддержкой кодировки UTF-8. Примеры настройки некоторых рекомендуемых программ для работы с UTF-8 приведены в Приложении 1–А.

Для поиска необходимых узлов конфигурационного дерева предназначена команда `search`. С её помощью можно найти все узлы, содержащие заданную подстроку в имени самого узла, подсказке или в тексте справки.

Если узел, найденный с помощью команды `search`, не виден в текущем дереве меню, это значит, что один или несколько промежуточных узлов (начиная с первого отсутствующего) зависит от каких-то других параметров предыдущего уровня и появляется только тогда, когда этот параметр имеет какое-то определённое значение. В этом случае следует изучить параметры предыдущего уровня, отмеченные знаком . Например, набор параметров канального уровня в меню порта зависит от выбранной инкапсуляции.

§1.2.4. Командная оболочка ОС Linux

Для пользователей, нуждающихся в непосредственном доступе к командам ОС Linux (в частности, для отладки системы и получения расширенной диагностики в сложных случаях), на устройстве имеется командная оболочка *ash* — одна из общепринятых для Linux. Чтобы получить доступ в неё, следует войти в систему с именем `root` любым из возможных способов: по Telnet, SSH, через консольный порт или через встроенный эмулятор терминала в Web-интерфейсе (`.system.login`). `root` является суперпользователем и имеет привилегию использовать оболочку *ash*.

Из командной оболочки *ash* можно запустить вложенную оболочку *nsgsh* командой `nsgsh`.

В процессе работы в *nsgsh* можно запустить вложенную оболочку *ash* командой `_tty`.

ПРИМЕЧАНИЕ В заводской конфигурации вход с именем `root` запрещён по соображениям безопасности. Чтобы сделать его возможным, необходимо установить пароль для пользователя `root` (см. §1.5.3).

§1.2.5. Множественный доступ к системе

Система допускает одновременный доступ нескольких пользователей через одну или разные командные оболочки и Web-интерфейс, в том числе — под одним именем, а также нескольких пользователей, имеющих права чтения и записи. Однако при этом следует различать административные права пользователя и административный статус его текущей сессии, который может быть равным или ниже административных прав.

Если к системе подключено одновременно несколько пользователей с правами `read-write`, то только один из них, подключившийся первым, будет иметь статус `admin`. Он сможет изменять значения параметров и выполнять команды, требующие прав записи и изменения конфигурации (`_apply`, `_write`, `_new`, `_insert`, `_remove` и т.п.). Остальным пользователям будет предоставлен доступ только в статусе `user` с текущими правами `read-only`.

Права, требуемые для настройки каждого параметра или выполнения команды, обозначаются цветом как в *nsgsh*, так и в Web-интерфейсе (см. §1.3.2, §1.4.1).

Если пользователь, имеющий статус `admin`, не активен в течение времени `.system.sessions.idle-time`, то при следующем входе какого-либо пользователя, имеющего права `read-write`, он получает статус `admin`, а сессия прежнего администратора завершается принудительно. Однако статус текущих сессий при этом не изменяется, т.е. если один из оставшихся пользователей желает стать администратором, то он должен выйти из системы и войти снова.

Максимальное время неактивности для всех сессий всех пользователей ограничено параметром `max-inactivity-time`. По истечении этого времени сессия принудительно завершается. Это позволяет избежать накопления неактивных сессий, которые могут возникать при некорректном завершении работы пользователя (обрыве связи, выходе из браузера без выхода из системы) или некорректной работе скриптов.

Пользователь `root` имеет специальную привилегию — он, и только он, может принудительно завершать сессии других администраторов при помощи команды `.system.sessions.close=admin`. (В версиях NSG Linux 2.0 *build 4* и ранее — следует узнать состояние текущих сессий командой `.system.sessions.show`, определить номер сессии, имеющей статус `admin`, и завершить её командой `.system.sessions.close`.) После этого необходимо выйти из системы и снова войти в неё, чтобы получить статус администратора.

Особым случаем является исполнение пользовательских скриптов, изменяющих конфигурацию с помощью утилиты `nsgsh`. Скрипты исполняются с правами `root` — но это не гарантирует им действительных прав администратора в данный момент, поскольку может уже исполняться административная сессия или другой скрипт. Поэтому при написании скриптов необходимо обратить внимание на права доступа для используемых в них команд. Подробнее см. описание `nsgsh` в [Части 7](#).

Управление по SNMP осуществляется от имени пользователя `root` и подчиняется общим правилам. В частности, если в системе уже работает пользователь со статусом `admin`, то SNMP-управление ограничивается только мониторингом.

§1.2.6. Формат записи конфигурации

Для вывода и ввода конфигурации в текстовом виде командами `_show`, `_print`, `_get` в оболочке `nsgsh` или кнопкой  в Web-интерфейсе могут использоваться два формата. В обоих форматах может быть записана как вся конфигурация, начиная от корня дерева, так и текущий узел. Выбор формата по умолчанию осуществляется индивидуально для каждого пользователя.

"Сырой" (*raw*) формат близок к оригинальному формату файла `/etc/nsgconfig` и отличается от него добавлением переносов строк и отступов в начале строки (в соответствии с уровнем вложенности каждого узла), например:

```
nsg:port>_print
{
  eth0 = {
    ifAddress = {
      broadcast = "192.168.1.3",
      peer = "192.168.1.2",
      prefix = "192.168.1.1/30"
    },
    link = {
      ["adm-state"] = "up"
    }
  }
}
```

Уровни вложенности узлов обозначены фигурными скобками. При вводе-выводе конфигурации (например, для отправки в службу техподдержки NSG) и при ручном редактировании необходимо обращать особое внимание на парность открывающих-закрывающих скобок и на наличие запятой *между* узлами одного уровня. После последнего узла каждого уровня, т.е. перед закрывающей скобкой этого уровня, запятая *не* ставится.

"Человеко-читаемый" (*user-friendly*) формат модифицирован для компактности и удобства визуального восприятия, например:

```
nsg:port>_print
eth0
: ifAddress
: : prefix      = "192.168.1.1/30"
: : peer       = "192.168.1.2"
: : broadcast  = "192.168.1.3"
: link
: : adm-state  = "up"
```

Двоеточие и пробел означают переход на следующий уровень меню. При вводе-выводе конфигурации (например, для отправки в службу техподдержки NSG) необходимо сохранять их в неизменном виде.

Этот же формат используется для записи конфигурации в данном документе и в другой документации по NSG Linux 2.0. (Ну не скриншоты же вставлять, право...)

§1.2.7. Разовые команды

Разовые команды, не сохраняемые в конфигурации (`show`, `launch` и т.п.), также являются листьями командного дерева. При вводе команды она исполняется немедленно. В Web-интерфейсе для исполнения команды следует нажать кнопку ▶.

Настройки команды, если таковые требуются (например, IP-адрес для `ping`), задаются в этом же узле дерева. В отличие от параметров, определяющих работу устройства, они не сохраняются в конфигурации и действуют только в пределах текущего сеанса работы пользователя.

Часть команд доступна только текущему администратору системы; как правило, это команды, критически влияющие на работу системы: перезагрузка, установка системного времени, генерация ключей и т.п. Другие — в сессиях с любыми правами; в основном, это команды просмотра различной информации, пользование прикладными инструментами (`ping`, `traceroute`, отправка почты, SMS и т.п.), а также рестарт отдельных портов. Права на исполнение каждой команды обозначаются цветом как в `nsgsh`, так и в Web-интерфейсе (см. §1.3.2, §1.4.1).

Кроме того, команды могут быть разовыми по существу (например, рестарт) и интерактивными (например, вход в другую оболочку). Интерактивные команды не могут исполняться в пакетном режиме `nsgsh`.

Для ресурсоёмких или особо ответственных команд, например, для рестарта системы, требуется дополнительное подтверждение от пользователя. При работе в `nsgsh` подтверждение запрашивается в интерактивном режиме; в пакетном режиме после вызова команды необходимо добавить значение `yes`. В Web-интерфейсе необходимо перед исполнением команды набрать `yes` в поле ввода, с соблюдением регистра.

§1.2.8. Типы данных

Параметры конфигурации (листья командного дерева), сохраняемые в энергонезависимой памяти, имеют один из следующих типов:

Логические параметры	Могут иметь только одно из двух булевских значений <code>true</code> или <code>false</code> . Используются для включения/выключения, активации/деактивации и т.п. различных функций.
Числовые параметры	Могут вводиться: <ul style="list-style-type: none"> — В десятичном виде (например, <code>86400</code> или <code>255</code>). — В шестнадцатеричном виде (<code>0xFF</code> или <code>0xff</code>). — В виде арифметического выражения (<code>60*60*24</code>). Тип (целое/нецелое) и диапазон возможных значений указывается во встроенной справке. Используются для указания интервалов времени, числа попыток, номеров портов, количества дочерних объектов и т.п.
Перечислимые параметры	Может принимать только одно значение из фиксированного набора, например <code>policy = "ACCEPT"</code> или <code>"DENY"</code> .
Строковые параметры	Значение — произвольная строка символов. Используются для большинства объектов конфигурации. Строка может включать в себя любые ASCII-символы — в т.ч. и специальные (пробел и др.), однако в этом случае для неё требуется более строгий формат ввода (см. §1.3.4). Для отдельных параметров может быть предусмотрен только более узкий набор символов и/или специфический формат. В частности, IP-адреса, префиксы, маски и шаблоны адресов (инверсные маски) рассматриваются как текстовые строки строго определённого формата. Строка, представляющая собой имя элемента списка (см. §1.2.9), как правило, должна начинаться с буквы. (Исключения оговариваются отдельно.)
<code>nil</code>	Специальное значение. Используется для установки значения параметра по умолчанию. Для параметров, являющихся конечными узлами ("листьями") дерева, равносильно команде <code>_remove</code> или <code>-</code> .

В конфигурации устройства все перечислимые и строковые параметры заключаются в двойные кавычки. При вводе эти кавычки не обязательны, за исключением случаев использования спецсимволов. Логические, числовые параметры и `nil` пишутся всегда без кавычек.

В консольной оболочке `nsgsh` для объёмных строковых параметров, например, скриптов или фильтров динамических маршрутов, допускается режим полноэкранного ввода с помощью редактора `paop`. В редакторе можно использовать перевод строки (`Enter`), а при выходе из редактора полученный текст сохраняется в виде одной строки с разделителями `\n`.

В Web-интерфейсе логические и перечислимые параметры представлены полем выбора с фиксированным набором значений, а числовые и строковые параметры — полем свободного ввода. Если строка содержит разделитель `\n`, то при попытке редактирования она автоматически преобразуется в текстовое окно. Для некоторых параметров, которые типично являются многострочными, режим текстового окна используется по умолчанию.

§1.2.9. Уникальные объекты, именованные и нумерованные списки

Каждый узел конфигурации, создаваемый системой, является уникальным и имеет заранее определённое имя.

В ряде узлов администратор может создавать дочерние объекты, например, новых пользователей, маршруты, туннели, правила фильтрации и т.п., причём их число программно не ограничено. Совокупность таких объектов образует *список*. В зависимости от природы этих объектов, они могут быть *именованными* или *нумерованными*. При создании элемента списка необходимо указать его имя либо номер, соответственно.

Именованный объект имеет в системе алфавитно-цифровое имя, причём начинаться оно должно, как правило, с буквы. В конфигурации элементы именованного списка выглядят так же, как и уникальные объекты, предопределённые в структуре меню.

ВНИМАНИЕ Категорически не рекомендуется использовать имена, содержащие только заглавные буквы и знаки препинания. Имена такого вида зарезервированы для служебных целей.

Элементы нумерованного списка идентифицируются номерами, начиная с 1, например:

```
nsg:port.eth0.ifAddress>_print
1
: prefix= "1.2.3.4/24"
: _keep=true
2
: prefix= "5.6.7.8/24"
: _keep=true
```

Один узел конфигурационного дерева может содержать в себе не более одного списка, а также, наряду со списком, произвольное число дочерних узлов.

Параметры, начинающиеся со знака подчёркивания (_), являются служебными. Они могут создаваться и удаляться системой автоматически и не предназначены для настройки пользователем; в последующих примерах они опускаются. В вышеприведённом примере таковым является параметр `_keep`, сигнализирующий о том, что данный узел создан пользователем.

Ключевое различие между двумя типами списков состоит в том, что нумерованные списки всегда являются плотными, т.е. система может перенумеровывать элементы подряд согласно определённым правилам. Предположим, что нумерованный список содержит N элементов, тогда:

- При создании списка будет автоматически предложен следующий свободный номер $N+1$.
- При попытке создания элемента с номером, большим N , он всегда получает номер $N+1$.
- При создании элемента с номером, меньшим N , новый элемент получает указанный номер, а прежний элемент с этим номером и все последующие сдвигаются вверх на единицу.
- При удалении элемента все последующие сдвигаются вниз на единицу, т.е. список будет содержать номера с элементами от 1 до $N-1$.

Порядковый номер элемента в списке может рассматриваться как некоторый приоритет, а может и не рассматриваться — это зависит от природы конкретного списка и указано во встроенной справке. В частности, он не является приоритетом, если элементы списка имеют некий приоритизирующий параметр внутри себя: приоритет, вес, метрику и т.п.

Особым случаем являются именованные списки, в которых имена являются целыми числами, или могут вводиться как целые числа и автоматически приводятся к некоторому типовому формату, например, `br1`. Внешне они могут быть похожи на нумерованные, но отличаются тем, что не упорядочиваются автоматически. Такие случаи оговариваются в документации особо.

Для построения командного дерева в `nsgsh` используются команды:

<code>_new</code>	Создание нового элемента именованного списка.
<code>_insert</code>	Создание нового элемента нумерованного списка.
<code>_remove</code>	Удаление текущего узла вместе со всеми его дочерними узлами. (Данная команда может применяться также к уникальным узлам, предопределённым в структуре меню; в этом случае все параметры внутри узла устанавливаются в значения по умолчанию.)

Начиная с версии NSG Linux 2.0 *build 5*, команды `_new` и `_insert` являются синонимами, нужный тип списка выбирается автоматически. Добавлен также новый формат команд — `+` и `-`.

В Web-интерфейсе используются кнопки `+` и `-`.

Если параметр не содержится в конфигурации, то он имеет специальное значение `nil`.

ВНИМАНИЕ В зависимости от природы конкретного параметра, отсутствие параметра может быть не тождественно ситуации, когда параметр присутствует и имеет некоторое значение по умолчанию, нулевое или пустое значение. Такие ситуации оговариваются особо во встроенной справке.

§1.2.10. Проверка, применение и сохранение конфигурации

Новые значения параметров, задаваемые пользователем, не вступают в силу немедленно. Для этого необходимо выполнить дополнительную операцию — *применить* конфигурацию. Если данная операция не может быть выполнена в текущем узле дерева, по природе этого узла, то она выполняется в вышестоящем узле.

Чтобы убедиться, что созданная конфигурация не содержит формальных ошибок и её применение не приведёт к неработоспособности устройства, рекомендуется выполнить дополнительную операцию — *проверить* конфигурацию.

Изменения текущей конфигурации — как применённые, так и не применённые — не сохраняются автоматически в энергонезависимой памяти. Сохранение производится отдельной командой.

ВНИМАНИЕ Операции *применить конфигурацию* и *сохранить конфигурацию* выполняются независимо друг от друга. Например, конфигурация может быть сохранена, но не применена; в этом случае она вступит в силу после отдельной операции *применить* или рестарта устройства.

Команды `nsgsh` для данных операций и соответствующие им кнопки Web-интерфейса приведены ниже:

Команда	nsgsh	Web
проверить	<code>_validate</code>	
применить	<code>_apply</code>	
сохранить	<code>_write</code>	

Важно помнить, что одновременно в устройстве имеются, в общем случае, три различные конфигурации: сохранённая в энергонезависимой памяти, редактируемая в интерфейсе пользователя и реально действующая в устройстве. Команда `_apply` или кнопка  копируют редактируемую конфигурацию в действующую (целиком или по фрагментам — в зависимости от того, где они использованы). Команда `_write` или кнопка  копируют редактируемую конфигурацию в сохранённую. При завершении сеанса работы пользователя редактируемая им конфигурация уничтожается, и если она не была сохранена и/или применена — то уничтожается безвозвратно.

ВНИМАНИЕ Имеют место исключительные ситуации, когда настройки, выполняемые в одном узле конфигурации, приводят к изменениям в других узлах: автоматическому созданию и удалению правил фильтрации, NAT, туннелей и т.п. Данные изменения не приводятся в действие без ведома пользователя, поскольку это может приводить к непредсказуемым результатам — в частности, к потере доступа к устройству. Их необходимо применить вручную, предварительно убедившись, что пользователь в данный момент готов к их последствиям.

ВНИМАНИЕ Необходимо проявлять особую осторожность при настройке тех элементов конфигурации, через которые производится управление устройством в данный момент: портов, туннелей, коммутаторов и т.п. После применения сделанных изменений пользователь может потерять доступ к устройству; в этом случае на устройстве останется активная сессия в статусе администратора, которая будет блокировать все новые сессии — её придётся специально снимать или дожидаться её отключения по тайм-ауту, и при этом будут безвозвратно утрачены все сделанные в ней изменения.

Настоятельно рекомендуется все изменения в одних сущностях конфигурации выполнять, по возможности, при доступе через другие сущности. Например, настройку встроенного коммутатора Ethernet и порта Ethernet выполнять через консольный порт. Если это невозможно, то следует сначала сохранить полученную конфигурацию (не применяя её!), а затем применить её или рестартовать устройство целиком.

§1.2.11. Зависимые параметры

Ряд узлов конфигурационного дерева зависит от других узлов. Например, если асинхронному порту назначена инкапсуляция `ppp`, то он будет иметь один набор параметров, если `reverse-telnet` — другой, а при инкапсуляции `popo` (т.е. порт отключён) — только минимальный набор, общий для всех протоколов. Параметры, значения которых влияют на наличие других узлов, отмечены в Web-интерфейсе знаком .

Если при смене некоторого параметра обнаруживается, что в конфигурации имеются зависимые параметры, специфичные только для его старого значения, то они сохраняются в виде отдельной ветви, не прикрепленной к общему дереву. Это обнаруживается, например, командой `_validate` (см. выше), но не мешает нормальной работе устройства в новой конфигурации. В этом случае `_validate` предлагает удалить их окончательно. Если не удалять их и когда-либо впоследствии вернуть параметру прежнее значение, то обрубленные ветви, зависимые от него, снова встают на место со своими прежними значениями.

§1.2.12. Явно указанные и расширенные параметры

Крупные и сложные пакеты Linux, такие как PPP, IPsec, IPtables, OpenVPN и др., имеют очень большое число параметров и опций, далеко не все из которых равно актуальны и широко употребимы. Выносить их все подряд в командный интерфейс маршрутизатора не только нецелесообразно, но и в большинстве случаев контрпродуктивно, поскольку избыток параметров мешает пользователю сориентироваться и быстро найти действительно важные параметры для своей утилитарной задачи. По этой причине в интерфейс (Web- и nsgsh) вынесена явным образом, как правило, только небольшая часть параметров: либо обязательные для всех конфигураций и важнейшие по существу для работы данной функции, либо наиболее употребимые в большинстве практических задач.

Для использования любых других параметров в таких пакетах предусмотрено поле `extra` в соответствующей ветви конфигурации. В нём могут быть указаны, в случае необходимости, любые параметры и опции, предусмотренные в командной строке данного пакета. Данная строка при обработке конфигурации копируется целиком, как есть, в строку вызова соответствующего приложения или в его временный файл конфигурации. (Наряду с параметрами и опциями, которые генерируются явно заданными параметрами конфигурации.)

В частности, если пользователю удобнее оперировать опциями и параметрами командной строки Linux, то возможно записать её целиком в поле `extra` и не вводить никакие другие параметры. Если у пользователя имеется готовый конфигурационный файл, отлаженный, например, на ПК с ОС Linux, то можно поместить его на устройство NSG и указать его местонахождение в поле `extra` (с соответствующим ключом в зависимости от синтаксиса конкретного приложения); он будет задействован и исполнен, и приложение будет запущено с точно теми же настройками без повторной конфигурации вручную. Например:

```
: tunnel
:: openvpn
::: ovpn1
:::: extra = "/etc/openvpn/myoldPCconfig.conf"
```

(для демона `openvpn`, если командная строка не содержит других параметров, саму опцию `--config` разрешается опускать).

Предложения о добавлении новых параметров явным образом в интерфейс NSG Linux 2.0, если пользователь считает их важными и актуальными для многих практических задач, следует направлять в службу технической поддержки NSG по электронной почте.

§1.2.13. Отладка системы, системные журналы

Общий системный журнал ОС Linux ведётся службой Syslog, которая позволяет выводить сообщения в локальный файл и/или на удалённый сервер. Настройка службы производится в узле `.system.syslog`. Основным параметром является административное состояние службы и место хранения журнала: выключена, в локальный файл, на удалённый сервер или на оба из них. Уровень важности сообщений, регистрируемых в журнале, указывается целым числом от 1 (система неработоспособна) до 8 (все сообщения, включая отладочные).

ВНИМАНИЕ Для правильной интерпретации событий, зафиксированных в различных журналах, и разбора нештатных ситуаций в сети, необходимо корректно установить системное время, как минимум, на данном устройстве. Настоятельно рекомендуется устанавливать время синхронно на всех устройствах сети от единого сервера NTP/SNTP.

Для локального файла можно указать специфический путь и имя (по умолчанию стандартное `/var/log/messages`), для удалённого сервера необходимо указать IP-адрес (или сетевое имя, разрешаемое через DNS) и возможно указать нестандартный порт. В первую очередь, явное указание локального файла может быть необходимо по той причине, что вся директория `/var` является временной и утрачивается при перезагрузке устройства. Для поиска и устранения неисправностей и ошибок конфигурации, приводящих, так или иначе, к перезагрузке, необходимо перенаправить журнал на такой носитель, который позволит сохранить его после перезагрузки и извлечь для последующего анализа. Локально на устройстве NSG это могут быть:

- Любой из типов сменной памяти, поддерживаемый данным устройством: USB Flash, SD card и т.п. Подробно о работе с внешними носителями см. [Часть 2](#).
- Директория `/etc`, хранящаяся во встроенной энергонезависимой памяти устройства. Данный вариант следует использовать с осторожностью, поскольку файл системного журнала пишется непрерывно, а любая флеш-память допускает только ограниченное число циклов перезаписи. Кроме того, размер ROM также ограничен. На устройствах с ROM и SSD рекомендуется хранить системный журнал в `/etc` только временно, в процессе отладки работы устройства, и вернуть его в исходное положение перед постановкой устройства в эксплуатацию в штатном режиме. Это ограничение не относится к устройствам, оснащённым HDD.
- На устройствах, оснащённых HDD/SSD, можно вручную создать в свободной области диска отдельный раздел и смонтировать в файловую систему средствами ОС Linux.

Ротация системного журнала регулируется двумя параметрами: размером сегмента и числом сегментов. Каждый сегмент представляет собой отдельный файл. Как только размер файла достигает предельного размера (с точностью до длины текущей записи), файл закрывается, переименовывается, и открывается следующий под тем же именем. Закрытые сегменты хранятся в файлах с же именем и суффиксами .0, .1 и т.д. По достижении максимального числа сегментов самый старый файл (с наибольшим номером) удаляется, оставшиеся перенумеровываются со сдвигом на 1, текущий сегмент сохраняется с суффиксом .0, и открывается новый файл. Таким образом, суммарный размер журнала есть произведение этих двух параметров. Увеличивая число сегментов и уменьшая размер единичного сегмента, можно при постоянном суммарном размере сделать ротацию более плавной.

Основным процессом, управляющим работой всех специфических компонент NSG Linux 2.0 (настройкой сети, запуском приложений), является NSG Configuration Daemon — `nsgconfd`. Журнал работы этого процесса сохраняется, по умолчанию, в файле `/var/log/nsgconfd.log`. Иное имя и/или путь к файлу, его размер и уровень детализации можно настроить по мере необходимости. К выбору места для хранения файла и к синхронизации системных часов относятся все те же соображения, что приведены выше для Syslog.

Просмотреть журналы можно командами `.system.log.show`, `.system.syslog.show`, `.system.syslog.filter="подстрока"`, или средствами командной оболочки Linux:

```
cat /var/log/messages      Вывести содержимое того или иного журнала, соответственно, на консоль.
cat /var/log/nsgconfd.log

tail -f /var/log/messages  Вывести конец журнала и продолжать вывод новых записей по мере их
tail -f /var/log/nsgconfd.log поступления. (Рекомендуется запускать в отдельном сеансе Telnet.)
```

При аномальной работе системы эти файлы, с большой вероятностью, будут полезны для самостоятельной отладки или для анализа службой технической поддержки NSG.

Помимо этого, полезная информация о работе системы, особенно в случае аномального функционирования, может быть получена стандартными для Linux командами и утилитами:

```
dmesg      Просмотр системных сообщений.
ps         Просмотр списка процессов.
top        Просмотр расходования памяти, вычислительной мощности, и наиболее
           потребляющих их процессов.
netstat    Просмотр показателей работы сетевой подсистемы IP. Для более подробной
           справки см. netstat --help или man pages.
tcpdump    Трассировка ввода/вывода на указанном сетевом интерфейсе. Для более
           подробной справки см. tcpdump --help или man pages.
```

Эти команды доступны как непосредственно из `ash`, так и из `nsgsh` и Web-интерфейса (узлы `.system.show` и `.tools`). С их помощью можно, в частности, быстро выявить такие проблемы, как утечки памяти, накопление неосвобождённых портов TCP/UDP, аномальное поведение отдельных процессов (постоянный рестарт или его отсутствие, зомби-процессы, аномальный расход ресурсов) и т.п.

§1.2.14. Завершение работы с системой

Для корректного завершения работы в командной оболочке `nsgsh` необходимо выполнить команду `_quit`. Для выхода из Web-интерфейса следует нажать кнопку . В обоих случаях, если имеются не сохранённые или не применённые изменения конфигурации, система выдаст соответствующее предупреждение и предложит сохранить/применить изменения. Если пользователь выбирает ответ No, т.е. "выйти без сохранения/применения", то:

- не применённые и не сохранённые изменения — утрачиваются немедленно
- сохранённые и не применённые — вступают в силу после следующего рестарта устройства
- применённые, но не сохранённые — остаются в силе до следующей перезагрузки устройства, но не видны в новых сессиях и не могут быть изменены или отменены без перезагрузки.

Утраченные изменения не могут быть восстановлены.

Для выхода из командной оболочки `ash` необходимо ввести команду `exit`.

Если командная оболочка является единственной и запущена при входе пользователя в систему, то при выходе из неё пользователь возвращается на этап входа в систему и должен снова ввести имя и пароль. Если командная оболочка была запущена в качестве вложенной (дочерней) из-под другой оболочки, то по завершении работы происходит возврат в родительскую оболочку.

§1.3. Командная оболочка NSG Shell

§1.3.1. Общие сведения

`nsgsh` (NSG Shell) — командная оболочка NSG Linux 2.0, предназначенная для управления в режиме текстового терминала при доступе к устройству по Telnet, SSH или через консольный порт. Оболочка обеспечивает перемещение по дереву конфигурации, создание и удаление ветвей и листьев, просмотр, проверку, применение и сохранение конфигурации.

Запуск `nsgsh` производится автоматически при входе в систему с именем `nsg` или именем пользователя, созданного администратором. При работе в командной оболочке `ash` для запуска вложенной командной оболочки следует ввести команду `nsgsh`; ввода имени и пароля при этом не требуется.

Текущее местонахождение пользователя на дереве конфигурации указывается в системном приглашении, например, `.port.eth0.ifAddress>`. Из этого узла пользователь может перейти в родительский (вышестоящий) узел, в один из дочерних (нижестоящих) узлов, в корень дерева, а также выполнить ряд системных команд, перечисленных ниже. Ввод команды во всех случаях завершается нажатием клавиши `<Enter>`.

ВНИМАНИЕ Большие и маленькие буквы при вводе команд, имён узлов и т.д. различаются, как это принято во всех Unix-подобных системах.

Для административных пользователей `nsg` и `root`, всегда существующих в системе, дерево конфигурации доступно полностью. Для каждого из остальных пользователей, создаваемых администратором для выполнения каких-либо частных задач, создаётся отдельное дерево конфигурации, состоящее из ссылок на узлы основного дерева. Пользователям будут доступны также дочерние узлы и параметры внутри этих узлов. Узлы основного дерева, находящиеся вне этих ссылок, для пользователей не видны и, следовательно, недоступны.

Оболочка `nsgsh` может работать либо в консольном режиме, либо в режиме экранного диалога. Сущность выполняемых действий от этого не меняется. Описание экранного режима см. в §1.3.9.

§1.3.2. Команды обзора и перемещения по дереву конфигурации

- `_` Вывести список системных команд, доступных данному пользователю.
 - `_help` Вывести краткую подсказку по данному узлу или параметру.
 - `_manual` Вывести подробную справку по данному узлу или параметру.
 - `_locale` Выбор системной локали (совокупности языковых и национальных настроек) для вывода подсказок и справок. Локаль указывается стандартными двухбуквенными сочетаниями: `en`, `ru` и т.п. Локаль, используемая по умолчанию, может быть установлена индивидуально в настройках каждого пользователя.
Для передачи национальных алфавитов используется кодировка UTF-8. Чтобы корректно читать русский текст, необходимо установить данную кодировку в свойствах используемого клиента Telnet, SSH или эмулятора терминала (см. Приложение 1–А). В Web-браузере кодировка устанавливается автоматически.
 - `_output-mode` Выбор формата (*raw* или *user-friendly*) для представления конфигурации командами `_show`, `_print` и т.п. Формат, используемый по умолчанию, может быть установлен индивидуально в настройках каждого пользователя.
 - `_show` Показать конфигурацию системы, вместе со всеми значениями по умолчанию, начиная от данного узла. Если выполнить данную команду в корне дерева, то выводится полная конфигурация.
- ВНИМАНИЕ** Необходимо различать системную команду `_show` и команды `show` (без подчеркика), имеющиеся внутри отдельных объектов для просмотра их состояния, статистики, журналов и т.п.
- `_print` Показать конфигурацию системы, начиная от данного узла. В отличие от `_show`, данная команда выводит, для краткости, только параметры, значения которых отличаются от умолчательных. Полученная текстовая конфигурация может быть скопирована из окна терминальной программы в буфер обмена на пользовательской машине или сохранена в простом текстовом файле.
 - `_get` Загрузить фрагмент конфигурации в текстовом виде — команда, парная к `_print`. После ввода данной команды `nsgsh` готова к приёму конфигурации. Следует передать, средствами терминальной программы, текстовый файл с фрагментом конфигурации, или вставить фрагмент конфигурации из буфера обмена. По завершении передачи необходимо нажать последовательно клавиши `<CTRL-D>` и `<Enter>`. Принятый текст проверяется на соответствие структуре текущего узла меню и, в случае успеха, вставляется в конфигурацию целиком. Если при проверке происходит ошибка, то текущая конфигурация устройства остаётся без изменений.

ВНИМАНИЕ При экспорте-импорте конфигурации в текстовом виде при помощи команд `_show`, `_print` и `_get` необходимо строго соблюдать форматирование и сохранять префиксы ":" (двоеточие, пробел) в начале строк (для формата *user-friendly*), либо фигурные скобки и запятые (для формата *raw*). В противном случае сохранённую конфигурацию нельзя будет загрузить обратно на устройство.

ПРИМЕЧАНИЕ Команда `_get` полностью стирает имеющуюся конфигурацию данного узла и полностью перезаписывает её. Поэтому в ней всегда необходимо вводить требуемую конфигурацию текущего узла целиком; если это нецелесообразно, то следует перейти в нужные дочерние узлы и загрузить конфигурацию каждого из них по отдельности.

`_fullscreen` Только в конечных параметрах ("листьях дерева"), имеющих строковый тип: ввод и редактирование данного параметра с помощью редактора `nano`. Редактор позволяет вводить объёмные параметры, например, скрипты, в виде текста из любого числа строк. После сохранения и выхода из редактора полученный текст преобразуется в одну строку с разделителями `\n`. Если текущее значение параметра содержит `\n` или `\r`, то в редакторе оно снова открывается как многострочный текст.

пустая команда (просто нажатие клавиши `<Enter>`)

Вывод перечня дочерних узлов и/или параметров, предусмотренных для данного узла. Перечень включает в себя как параметры, действительно содержащиеся в конфигурационном файле, так и отсутствующие в нём.

Если данный узел может содержать в себе именованный или нумерованный список (см. §1.2.9), определяемый пользователем, то наряду с именами или номерами существующих узлов выводится также системная команда `_new` или `_insert`, соответственно, для создания элементов списка. Если пользовательская терминальная программа поддерживает атрибуты текста, то при выводе используются следующие стили:

Синий	Промежуточные узлы дерева.
Обычный текст	Параметры (листья), имеющие значения по умолчанию.
Полужирный	Параметры (листья дерева), значения которых установлены пользователем.
Зелёный	Разовые команды (листья), не сохраняемые в конфигурации.
Ярко-зелёный	Разовые интерактивные команды (листья), не сохраняемые в конфигурации.

Для сессии с правами `user` параметры и команды, не доступные для изменения/исполнения в данной сессии, обозначаются следующим образом:

Ярко-Жёлтый	Параметры, значения которых установлены пользователем.
Жёлтый	Параметры, имеющие значения по умолчанию.
Красный	Разовые команды.
Ярко-красный	Разовые интерактивные команды.

имя узла/параметра

Переход в указанный дочерний узел, или ввод параметра. Если имя узла или параметра содержит спецсимволы, то оно вводится особым образом (см. §1.3.4).

`<Пробел>` или `_exit`

Переход к вышестоящему узлу командного дерева. Действует также при вводе параметра как `Esc`, т.е. возвращает в предыдущий узел без изменения текущего значения параметра.

`<Пробел><Пробел >` или `_end`

Переход к корню командного дерева.

§1.3.3. Команды построения дерева конфигурации

Для создания, удаления и клонирования узлов конфигурационного дерева используются следующие команды:

`_new` Создать новый элемент именованного списка внутри данного узла, ввести имя для него и перейти в этот узел, например:

```
system.users> <Enter>
  _new      Add new element to node
system.users> _new<Enter>
system.users._new = basile<Enter>
system.users.basile>
```

`_insert` Создать новый элемент нумерованного списка внутри данного узла, ввести номер для него и перейти в этот узел, например:

```
tunnels> <Enter>
  _insert   Insert element in list
tunnels> _insert<Enter>
tunnels._insert = 1<Enter>
tunnels.1>
```

Начиная с версии NSG Linux 2.0 *build 5*, команды `_new` и `_insert` являются синонимами, нужный тип списка выбирается автоматически.

ПРИМЕЧАНИЕ В качестве подсказки и справки для всех элементов списка выводится одно и то же предустановленное значение (или пустое поле), если по сути данных элементов внутри них не предусмотрен специальный параметр `@help`.

Все элементы списков, созданные пользователем, имеют служебный флаг `_keep=true`. (В последующих примерах для краткости он опущен.)

`_remove` Действие команды зависит от того, к чему она применяется:

- Для элементов списков, созданных администратором: удалить текущий элемент со всеми его дочерними узлами и параметрами.
- Для узлов и параметров, жёстко заданных в схеме конфигурации: установить значение по умолчанию (если команда применяется в узле, то для всех дочерних параметров данного узла). Для выполнения операции запрашивается подтверждение. После выполнения операции пользователь переходит в вышестоящий узел.

§1.3.4. Простой ввод параметров и разовых команд

При входе в параметр (конечный лист дерева) системное приглашение принимает вид:

```
hostname:node...node.parameter =
```

после чего следует ввести значение параметра. Чтобы просмотреть справку о возможных значениях данного параметра, следует ввести пустое значение (просто нажать `<Enter>`). Эта же справка выводится при вводе любого значения, недопустимого для данного параметра. Перечень или диапазон допустимых значений для некоторых параметров может зависеть от типа шасси и/или интерфейсной карты, версии программного обеспечения и настроек других параметров.

В именах узлов, именах параметров, логических (`true`, `false`) и перечислимых значениях допускаются любые сокращения, позволяющие однозначно определить их.

При вводе строковых параметров двойные или одинарные кавычки опциональны, за исключением следующих особых случаев:

- Пустая строка вводится как: `""` или `' '` (две кавычки подряд).
- Значение, которое по смыслу данного параметра является строкой, но содержит одни только цифры, необходимо вводить в кавычках. В противном случае она будет интерпретирована как числовое значение. На практике использовать такие строки не рекомендуется, но это приходится делать в некоторых случаях. Например, для различных случаев NAT можно указывать диапазон портов TCP или UDP (через дефис), список портов (через запятую), или единственный порт. Если первые две записи однозначно интерпретируются как строка, то одиночный номер порта выглядит как число, и, чтобы он соответствовал типу параметра, его необходимо взять в кавычки.
- Если строка содержит спецсимволы `() пробел ? ' " ; \ @` то она может вводиться:
 - С записью указанных спецсимволов в виде *esc*-последовательностей `\`, `'` и т.п.
 - В одинарных (`'`) или двойных (`"`) кавычках. В этом случае (внутри открытых кавычек одного типа) кавычки другого типа спецсимволом не являются. Кавычки того же типа необходимо вводить в виде *esc*-последовательностей.

- Если строка содержит спецсимволы `. =` то в интерактивном режиме (после приглашения, завершающегося знаком `=`) она может вводиться как есть. В пакетном режиме, если для присвоения значения параметру используется оператор `.` (синтаксис до NSG Linux 2.0 *build 4* включительно), то значение, содержащие данные спецсимволы, необходимо вводить так же, как и в предыдущем случае.

Во всех случаях (в кавычках или без) строка может содержать также *esc*-последовательности `\r \n \t` и десятичную подстановку `\nnn` (где `nnn` — ASCII-код требуемого символа). Чтобы ввести такие последовательности буквально, следует вводить в них `\` в виде *esc*-последовательности `\\`.

После ввода параметра пользователь остаётся в вышестоящем узле.

Параметры, отсутствующие в конфигурации, имеют специальное значение `nil`. При вводе любого другого значения (в т.ч. совпадающего со значением по умолчанию) параметр и все вышестоящие узлы дерева автоматически включаются в конфигурацию.

При вводе параметров действительно большинство системных команд, перечисленных в остальных параграфах данного раздела, в частности:

<code>_manual</code>	Получение подробной справки о параметре.
<code>_remove</code>	Установка параметра в значение по умолчанию.
<code>_exit</code>	Выход из приглашения ввода значения.

Не имеют смысла только команды `_new`, `_insert` и `_apply` (хотя формально они не запрещены).

Помимо параметров, определяемых пользователем, в системе могут существовать служебные параметры, устанавливаемые и удаляемые автоматически. Имена таких параметров всегда начинаются с подчеркива. Эти параметры недоступны для изменения пользователем и только выводятся командами `_print`, `_show`. Например, для элементов именованных и нумерованных списков устанавливается флаг `_keep=true`. В последующих примерах, для краткости, эти параметры опущены.

Если лист дерева представляет собой разовую команду (`show`, `launch` и т.п.), то при выборе этого листа в `nsgsh` команда запускается на исполнение немедленно. Параметры команды, если они необходимы, настраиваются другими листьями в этом же узле, например:

```
nsg:tools.ping> <Enter>
launch      Запустить на исполнение
count       Число посылок ping (0 – неограниченно)
host        IP-адрес назначения
size        Число байт в пакете ping
source-ip   IP-адрес источника
```

Эти параметры действуют локально в пределах текущего сеанса работы пользователя и не сохраняются в конфигурации. Просмотреть их можно командой `show`, в которой они выводятся отдельным разделом.

Если команда, будучи единожды запущенной, предусматривает работу некоторой программы в течение неограниченного времени (например, `ping`), то для её завершения необходимо ввести в `nsgsh` одну из двух управляющих последовательностей:

```
<CTRL-C><CTRL-C>    Передать в исполняемую программу CTRL-C (нормальное завершение).
<CTRL-C><CTRL-K>    Передать в систему KILL для исполняемой программы (аварийное завершение).
```

§1.3.5. Сложный ввод параметров и разовых команд

Начиная с версии NSG Linux *build 5*, при вводе команд `nsgsh` возможен расширенный формат команд и параметров. В основном, он нужен при пакетном запуске `nsgsh` (см. §1.3.8) но отдельные его элементы могут быть полезны и при интерактивной работе.

Допускается не только пошаговый переход от узла к узлу, но и ввод всего пути к нужному узлу или параметру целиком:

<code>node1.node2...nodeN</code>	Переход в узел N, начиная от текущего положения на командном дереве.
<code>node1.node2...nodeN.parameter</code>	Переход к установке значения параметра (или исполнению разовой команды).
<code>node1.node2...nodeN.parameter.значение</code>	Установка значения с возвратом в исходную точку. При написании команды в данном формате, если значение является строковым параметром и может содержать знаки <code>.</code> и <code>=</code> , то их необходимо вводить в виде <i>esc</i> -последовательностей <code>\.</code> и <code>\=</code> , либо заключить всё значение в кавычки (двойные или одинарные).
<code>node1.node2...nodeN.parameter=значение</code>	Установка значения с возвратом в исходную точку. При написании команды в данном формате, в отличие от предыдущего, знаки <code>.</code> и <code>=</code> не являются спецсимволами и могут вводиться как есть.
<code>..node1.node2...nodeN</code>	Переход в узел N, начиная от корня командного дерева.
<code>..</code>	Переход в родительский узел дерева.
<code>...</code>	Переход в прародительский узел дерева.
и т.п.	
<code>..node1.node2...nodeN</code>	Переход в узел N, начиная от родительского узла.
и т.п.	

Аналогично для ввода параметров.

Названия параметров и названия узлов дерева, как и значения параметров, могут содержать спецсимволы. В этом случае для них действуют те же правила ввода, что и для строковых параметров (см. §1.3.4) — т.е. проблемный узел вводится в кавычках или с использованием *esc*-последовательностей. Без особой необходимости такие названия употреблять не следует, но если есть такая потребность, то это не запрещается, например:

```
.tunnel.ipsec.connections."Kolxoz \"Krasnyi Konoplevod\"".leftid=Pravlenie\ Kolxoza
```

В одной строке допускается вводить несколько команд, разделяя их пробелами и/или точкой с запятой (в любом сочетании), например:

```
nsg:port.m1> type="3g" ppp.main.chat.APN=access.point.name ; ppp.main.chat.mode=GSM<Enter>
```

В этом случае команды исполняются последовательно, как если бы между ними нажималась клавиша <Enter>. Иначе говоря, последующие команды могут быть зависимыми от предыдущих (а также от установок, вытекающих из них по умолчанию). В приведённом примере существенно то, что первым устанавливается тип порта 3g и следующая из него инкапсуляция ppp; только после этого в дереве появляются специфические для PPP узлы и параметры, которые обрабатываются второй и третьей командами. При вводе команд в обратной последовательности они не выполняются.

Переход в некоторый узел и исполнение нескольких команд в нём. Например, приведённая выше последовательность команд может быть записана так:

```
nsg:port.m1> type="3g" ppp.main.chat(APN=access.point.name mode=GSM)<Enter>
```

Следует обратить внимание, что открывающая скобка одновременно заменяет собой точку. Перед и после скобок допускаются пробелы.

Допускается многоуровневая вложенность:

```
nsg:port > m1(type="3g" ppp.main(chat(APN=access.point.name; mode=GSM) set-defaultroute=true))<Enter>
```

Последней командой в строке или в подстроке (в скобках) может быть как установка параметра, так и переход в узел. Если же переход в узел стоит не последним, то следующая команда будет исполнена от исходного узла, и сделанный ранее переход будет забыт.

Аналогичным образом могут исполняться системные команды в заданном узле. Кроме того, для ряда команд возможен ещё один формат:

node1.node2...nodeN._print =node1.node2...nodeN	Вывести пользовательскую конфигурацию узла nodeN.
=	Вывести пользовательскую конфигурацию текущего узла.
node1.node2...nodeN._show ==node1.node2...nodeN	Вывести полную конфигурацию (включая все значения по умолчанию) узла nodeN.
==	Вывести полную конфигурацию текущего узла.
node1.node2...nodeN._manual ?node1.node2...nodeN	Вывести справку для узла nodeN (аналогично — для параметра).
?	Вывести справку для текущего узла.
node1.node2...nodeN._new=имя node1.node2...nodeN._insert=имя +node1.node2...nodeN.имя +имя	Добавить новый элемент списка (тип списка определяется автоматически) с заданным именем или номером.
+node1.node2...nodeN._new +node1.node2...nodeN._insert +	Добавить новый элемент списка в указанный (или текущий) узел без явного указания. Будет выведено интерактивное приглашение ко вводу имени/номера элемента.
node1.node2...nodeN.имя._remove -node1.node2...nodeN -node1.node2...nodeN.имя	Удалить заданный узел или параметр. Для узлов и параметров, существующих всегда (т.е. не являющихся элементами списков) — установить для узла и всех его дочерних элементов по умолчанию значения.
-	Удалить или очистить текущий узел/параметр.

§1.3.6. Специальные и итоговые команды

<code>_tty</code>	Запустить дочернюю командную оболочку <code>ash</code> для выполнения операций в ОС Linux, недоступных обычному пользователю. Команда доступна только для пользователя <code>root</code> .
<code>_tty="команда"</code>	Запустить дочернюю оболочку <code>ash</code> , выполнить в ней заданную команду и вернуться в <code>nsgsh</code> .
<code>_validate</code>	Проверить корректность текущего узла.
<code>_apply</code>	Применить фрагмент конфигурации, начиная от текущего узла. Выводится вновь созданная конфигурация и запрашивается подтверждение. Если команда <code>_apply</code> в текущем узле недоступна, это означает, что применение сделанных изменений затрагивает также и другие узлы меню. В этом случае команду <code>_apply</code> следует использовать в одном из вышестоящих узлов.
<code>_write</code>	Сохранить конфигурацию в энергонезависимой памяти.
<code>_quit</code>	Выход из <code>nsgsh</code> . Если при этом в конфигурации имеются не применённые или не сохранённые фрагменты, будет выдано соответствующее предупреждение.

§1.3.7. Соглашения и сокращения

Для большего удобства пользования в `nsgsh` поддерживаются следующие соглашения и сокращения, принятые в большинстве консольных командных оболочек:

- При вводе команды (в том числе — имени дочернего узла для перехода в него) достаточно ввести только начальную последовательность символов, определяющую её однозначно. В частности, для системных команд достаточно вводить `_a`, `_w` и т.п. После этого команда может быть автоматически дополнена по нажатию клавиши `<TAB>`.
- Если введённое начало команды неоднозначно, то выводится список команд, имеющих такое начало.
- При вводе логического или перечислимого параметра (например, `up` либо `down`) также достаточно ввести его однозначное начало.
- Разрешается произвольное редактирование ввода при помощи клавиш `←`, `→`, `` и `<Backspace>`. Однако следует иметь в виду, что функция данных клавиш, особенно ``, может зависеть от конкретной программы эмуляции терминала или клиента Telnet.
- Клавиши `↑` и `↓` позволяют листать историю команд и повторять ранее введённые команды.

§1.3.8. Исполнение `nsgsh` из командной строки Linux

Командная оболочка `nsgsh` может исполняться из командной строки оболочки `ash` в пакетном режиме. Операндами при её вызове являются внутренние команды `nsgsh`, которые следует выполнить. После выполнения этих команд `nsgsh`, по умолчанию, завершает работу и возвращает управление `ash`. Формат вызова `nsgsh`:

```
nsgsh опции путь.параметр=значение [[путь.параметр=значение] ... ]
```

Возможные форматы операндов (переход в узел, установка параметра, от текущего узла, от корневого узла, от родительского узла и т.п.) описаны в §1.3.4, §1.3.5. В отличие от интерактивной работы в `nsgsh`, при пакетном вызове к перечню спецсимволов, требующих указания особым образом, добавляются спецсимволы `ash` (`$` `>` `<` `|` `&` ``` `(` `)` и т.п.) — в путях, параметрах и значениях. Символ `/` в данном случае специальным не является.

ВНИМАНИЕ В отличие от версий NSG Linux 2.0 *build 4* и ранее, точка вводится так же, как и другие спецсимволы — т.е. в кавычках или в виде *esc*-последовательности `\`. Прежний формат (две точки подряд преобразуются в одну, вводимую буквально) более не работает.

Подробное описание утилиты `nsgsh` приведено в [Части 7](#). Пакетный запуск может производиться как вручную для выполнения априори известных действий, так и автоматически из различных скриптов, предусмотренных в конфигурации. При этом особого внимания требует вопрос о правах доступа для каждой конкретной команды при каждом конкретном запуске `nsgsh`. Требуемые права можно определить по цветовой индикации в `nsgsh` и Web-интерфейсе.

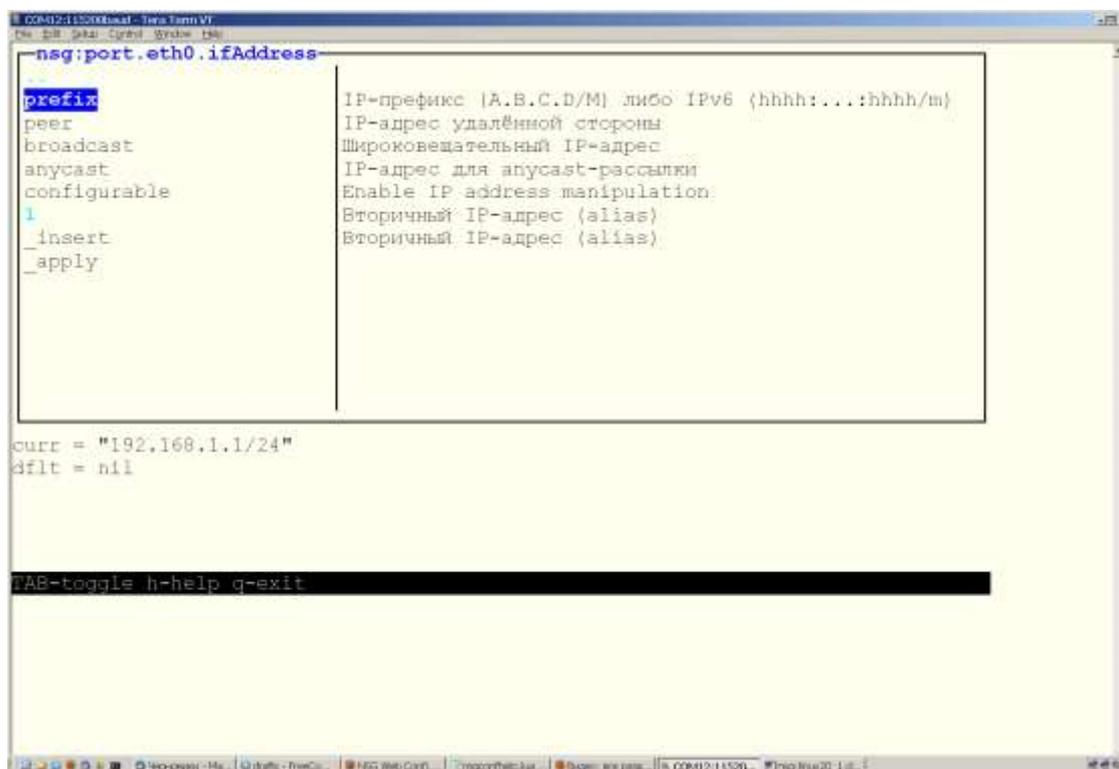
Подробно об опциях и параметрах `nsgsh` см. [Часть 7](#).

§1.3.9. Экранный режим работы nsgsh

Помимо обычного консольного режима, при котором команды пользователя и ответы системы последовательно выводятся на экран и прокручиваются снизу вверх, в *nsgsh* реализован режим работы в виде экранного диалога. Этот режим несколько более удобен для интерактивной работы, но требует, чтобы используемая программа Telnet, SSH или эмуляции консоли работала в режиме какого-либо интеллектуального терминала (VT52, VT100 и т.п.), а не простейшего TTY.

Предпочтительный режим для каждого пользователя устанавливается в узле настройки данного пользователя (*.system.users.имя.curses*). Если предпочтительный режим не установлен, то при каждом входе в систему предлагается выбрать режим на данную сессию; по умолчанию предлагается использовать консервативный консольный интерфейс. Чтобы избавиться от этого предложения, следует явно выбрать тот или иной предпочтительный режим.

В экранном режиме окно программы разбивается на несколько функциональных областей. В верхней строке выводится имя устройства и путь к текущему узлу конфигурации. Ниже выводится список команд и параметров данного узла и подсказки к ним; двумя точками (..) традиционно обозначается переход на вышестоящий уровень дерева. Перемещение по списку команд производится клавишами ↑ и ↓. Ниже таблицы выводятся текущее значение параметра, на котором стоит указатель, и его значение по умолчанию. С помощью клавиш ← и → можно перемещаться вперёд и назад по истории команд. Разбиение окна подбирается динамически в зависимости от его размеров; если подсказка не умещается в отведённую для неё область, она выводится "бегущей строкой".



Выбор элемента меню производится клавишей Enter. Если выбранный элемент также представляет собой узел конфигурационного дерева, то выполняется переход в этот узел; если это конечный элемент (лист дерева) — то ниже появляется поле для ввода нового значения.

ПРИМЕЧАНИЕ. В отличие от консольного режима, внутри конечного узла (листа) дерева, т.е. непосредственно в процессе ввода его значения, системные команды не действуют; для отказа от ввода следует использовать клавишу Esc.

В меню также добавлены некоторые некоторые из системных команд *nsgsh* (см. предыдущие параграфы), наиболее актуальные в данном узле. Полный список служебных команд можно просмотреть по нажатию клавиши h, что эквивалентно *_help*. Все системные команды можно выполнять нажатием клавиши, соответствующей первой букве команды: m, a, i, n, r и т.д.

Если вывод некоторых команд (в частности, *manual*, *print*, *show*) не помещается целиком в нижней части окна, то можно скрыть/открыть верхнюю часть окна клавишей TAB, чтобы освободить больше места на экране, а если этого мало — использовать прокрутку клавишами ↑ и ↓.

В процессе работы в консольном режиме можно переключиться в экранный режим с помощью команды `_curses-mode`; для возвращения в режим ТТУ достаточно нажать `q`. Это особенность работы команды `quit` в экранном режиме. Если же `nsgsh` была запущена в консольном режиме изначально (согласно настройкам для данного пользователя или его выбору в момент входа в систему), то `q`, как обычно, завершает её работу.

При запуске `nsgsh` из командной строки `bash` можно использовать опцию `-C`.

ПРИМЕЧАНИЕ Все записи справки отформатированы в предположении традиционного текстового терминала шириной 80 символов. Динамическое форматирование записей не предусмотрено ввиду необходимости поддерживать совместимость с простейшими ТТУ-терминалами. При выводе на терминал с меньшей шириной окна возможны искажения, снижающие удобство восприятия текста.

§1.4. Web- и SNMP-управление

§1.4.1. Web-интерфейс

Web-интерфейс NSG Linux 2.0 представляет собой графический инструмент для манипулирования файлом конфигурации и выполнения разовых команд. По умолчанию, Web-интерфейс работает по протоколу HTTP на стандартном порту TCP 80.

Web-интерфейс представляет дерево конфигурации в графическом виде и позволяет изменять его узлы и листья. Структура командного дерева в основном окне Web-интерфейса копирует структуру команд консольной оболочки `nsgsh`. (За исключением отдельных команд, которые являются специфичными для Web-интерфейса и принципиально не имеют аналогов в `nsgsh` — например, загрузки-выгрузки файлов непосредственно из пользовательского Web-браузера).

После входа в систему дерево изображается развёрнутым на один уровень. Следует нажать мышью символ  слева от одного из узлов, чтобы развернуть эту ветвь на следующий уровень, и далее идти таким же образом до нужного параметра. Чтобы свернуть ненужную ветвь дерева, следует нажать символ .

ВНИМАНИЕ Если Web-интерфейс ведёт себя неадекватно (не разворачиваются ветви, не показывается справка и т.п.), то следует, в первую очередь, очистить кэш браузера. Данная проблема, как правило, связана с изменениями компонент Javascript при смене версии NSG Linux: старые скрипты, сохранённые в кэше, оказываются не соответствующими новой версии ПО. Для некоторых браузеров (в частности, Mozilla Firefox) доступны соответствующие дополнения для очистки кэша с помощью специальной кнопки или горячей клавиши. Рекомендуется использовать их при администрировании большого числа устройств и/или частой смене ПО.

Для административных пользователей `nsg` и `root`, всегда существующих в системе, дерево содержит все узлы и параметры, предусмотренные для данного устройства. Для других пользователей поддерево доступных параметров конструируется индивидуально.

В отличие от `nsgsh`, в Web-интерфейсе все параметры, доступные пользователю, выводятся постоянно (аналогично команде `_show`). При этом значения по умолчанию показаны серым цветом.

При наведении мыши (без щелчка) на имя параметра выводится всплывающая краткая подсказка по этому параметру. Чтобы просмотреть подробную справку по данному параметру, следует нажать кнопку . Изменение локали (языка) для справок и подсказок производится с помощью меню в верхней части экрана и действует до конца текущего сеанса работы; в новом сеансе оно устанавливается снова в соответствии с настройками данного пользователя.

Для ввода или выбора значения параметра необходимо дважды щёлкнуть мышью в этом поле. Поля ввода для логических и перечислимых параметров представлены списком с фиксированным набором значений. Для числовых и строковых параметров даётся окно свободного ввода; если введено значение, недопустимое для данного параметра, оно игнорируется, выводится подсказка, и параметр сохраняет прежнее значение.

Строковый параметр может содержать разделители `\n` или `\r`. При редактировании такой строки она преобразуется в многострочное текстовое окно, а при выходе из этого окна — снова в строку с разделителями. Для отдельных параметров, например, для скриптов, фильтров динамических маршрутов, многострочный режим установлен по умолчанию.

Если природа некоторого узла конфигурации предполагает, что внутри него создается именованный или нумерованный список, то для создания новых элементов списка следует нажать кнопку . Для удаления элемента списка следует нажать кнопку  рядом с ним.

Аналогом команд `_print` и `_get` для экспорта-импорта конфигурации в текстовом виде является режим текстового редактирования. Режим доступен как в корне дерева (при этом пользовательская конфигурация выводится полностью), так и в отдельных узлах и управляется кнопками:

-  Перейти в режим текстового редактирования.
-  Вернуть исходное состояние данного фрагмента конфигурации.
-  Проверить данный фрагмент конфигурации.
-  Проверить данный фрагмент конфигурации и выйти из режима текстового редактирования.

ВНИМАНИЕ Режим текстового редактирования предназначен, в основном, для вывода конфигурации (например, для отправки в службу технической поддержки NSG) и для клонирования и загрузки готовых фрагментов конфигурации. Ручное редактирование выполняется пользователем на свой страх и риск и требует хорошего понимания своих действий. При редактировании критически важно соблюдать уровни вложенности команд, указываемые сочетаниями "двоеточие и пробел".

Другие операции с объектами выполняются при помощи кнопок:

-  Справа от каждого параметра: перейти в режим редактирования данного параметра. Аналогично двойному щелчку мышью, но актуально при работе с планшетами и других устройств, где двойной щелчок перехватывается системой и используется для других целей, или не предусмотрен вовсе.
-  Установить для данного узла и всех его дочерних узлов и параметров значение по умолчанию. В частности, удалить все элементы дочерних списков.
-  Применить фрагмент конфигурации, начиная от данного узла, или выполнить разовую команду.
-  Завершить исполнение команды.
-  Сохранить конфигурацию.
-  Обновить конфигурацию на экране.
-  Включить/выключить режим показа пользовательской конфигурации каждого узла непосредственно в строку с ним (насколько хватит места). При наведении мыши конфигурация узла показывается полностью в плавающем окне.
-  Открыть дополнительную Web-страницу для выполнения операций, принципиально не имеющих аналога в консольном интерфейсе (например, мониторинга системы **uiTCP**).
-  Завершить работу в Web-интерфейсе.

В зависимости от прав, требуемых для редактирования параметров и исполнения команд, используются следующие цветовые обозначения кнопок:

- Синий** Действие требует прав администратора и доступно в данной сессии.
- Серый** Действие требует прав администратора и не доступно в данной сессии.
- Зелёный** Разовая команда не требует прав администратора и доступна в любой сессии.

Различия между разовыми и интерактивными командами в Web-интерфейсе не предусмотрено, поскольку он, по определению, интерактивен.

Для доступа к консольным командным оболочкам используется команда `.system.login`. При этом на устройстве запускается клиент Telnet на локальный адрес (127.0.0.1) и порт TCP 23. Ввод/вывод этого клиента транслируется в Web-интерфейс.

При работе в любых консольных программах можно использовать escape-последовательности из двух клавиш `^A...^Z`, `^[`, `^]`, `^_` для ввода спецсимволов CTRL-A и т.д., и `^^` для ввода одиночного символа `^`.

ВНИМАНИЕ Web-управление является одним из наиболее ресурсоёмких компонентов системы. На устройствах серии NSG-600 с их ограниченными аппаратными ресурсами (в особенности, оперативной памятью) рекомендуется использовать его только для первоначальной настройки и отключить его перед постановкой устройства в эксплуатацию. Дальнейшее управление устройством, если таковое потребуется, в этом случае производится с помощью SSH или Telnet.

§1.4.2. SNMP-управление

В программное обеспечение устройств NSG Linux 2.0 входит встроенный агент SNMP с поддержкой версий SNMP v1, 2 и 3. Для мониторинга и управления данными устройствами могут использоваться любые стандартные платформы сетевого управления на основе SNMP. Собственно агент SNMP обеспечивает доступ к устройству NSG как к объекту управления, а также определяет субъекты этого управления (SNMP communities). Настройка SNMP-агента осуществляется в узле `.services.snmp`.

Подробнее о настройке агента SNMP см. [Часть 4](#).

§1.5. Общесистемная конфигурация

§1.5.1. Имя устройства

Системное имя устройства устанавливается в пункте меню `.system.hostname`. По умолчанию используется имя `nsg`. Текущее имя устройства выводится в подсказке `nsgsh` и указывается вверху Web-страницы:

```
nsg:system>hostname
nsg:system.hostname = myrouter
= "myrouter"
nsg:system>_a
Config to apply
.....
Are you sure?[yes/no] yes
OK
myrouter:system>
```

§1.5.2. Установка системного времени

Системное время на устройстве может быть установлено вручную командой `.system.clock-set`. Данная команда, однако, имеет практический смысл только в устройствах, оснащённых энергонезависимыми часами реального времени. В устройствах младшей серии NSG-600 такие часы отсутствуют, поэтому при выключении питания устройства системное время в них не сохраняется.

Автоматическая установка или коррекция времени может производиться с помощью встроенного клиента NTP. Настройка производится в узле `.system.ntp`. Обязательными параметрами являются IP-адрес или доменное имя сервера NTP (доменное имя — только при включённом клиенте DNS на устройстве) и периодичность опроса. Первая попытка соединения с сервером производится сразу после старта системы. По умолчанию, клиент NTP выключен.

Для корректной установки местного времени и сдвига зимнего/летнего времени следует выбрать соответствующий часовой пояс в узле `.system.timezone`. Если нужный часовой пояс отсутствует в списке, то следует выбрать значение `other`, при этом появится отдельное окно для ввода переменной окружения TZ явным образом. Подробно о формате данной переменной (хранится в файле `/etc/TZ`) см. документацию по ОС Linux.

ПРИМЕЧАНИЕ В случае изменения правил исчисления поясного и зимнего/летнего времени необходимо обновить NSG Linux до версии, выпущенной не ранее принятия закона о данных изменениях.

Текущее системное время можно посмотреть командой `.system.show.date`.

§1.5.3. Управление локальными пользователями

По умолчанию, в системе имеются два пользователя — `root` и `nsg`. О различиях между этими пользователями см. §1.1.5. Эти два пользователя существуют всегда и имеют доступ к полному дереву команд. Для каждого из этих пользователей возможно:

- Установить или изменить пароль для входа в систему.
- Заблокировать (запретить) или разблокировать (разрешить) вход в систему.
- Установить локаль (язык) по умолчанию. Эта локаль будет использоваться для вывода встроенных подсказок и справки, как в Web-интерфейсе, так и в консольном интерфейсе. Для вывода текста на национальных языках в консольном интерфейсе необходимо использовать клиент Telnet/SSH или эмулятор терминала с поддержкой UTF-8. Указания по настройке UTF-8 в рекомендуемых клиентах приведены в Приложении I-A.
- Установить формат вывода конфигурации в текстовом виде по умолчанию: `raw` или `user-friendly`.

ВНИМАНИЕ В заводской конфигурации устройства пользователь `nsg` имеет пустой пароль. **ПЕРЕД УСТАНОВКОЙ УСТРОЙСТВА В РЕАЛЬНУЮ СЕТЬ НЕОБХОДИМО УСТАНОВИТЬ УНИКАЛЬНЫЙ ПАРОЛЬ** для пользователя `nsg`.

Заводской пароль пользователя `root` умышленно установлен некорректным образом. Таким образом, вход в систему с этим именем невозможен. Это сделано намеренно в целях безопасности, чтобы избавить администратора от необходимости назначать и хранить два пароля во всех случаях. Если же администратор предполагает, что для управления устройством ему понадобится входить в систему как `root`, то необходимо установить пароль также и для этого пользователя.

Помимо этих двух пользователей, администратор может создавать неограниченное количество других пользователей с ограниченными полномочиями, предназначенных для выполнения частных задач. Например, это может быть оператор дежурной смены, имеющий доступ только к мониторингу состояния устройства и рестарту отдельных портов. Или же это может быть владелец некоторого оборудования, удалённо управляемого через консольный порт и устройство NSG в режиме Reverse Telnet; дополнительно, для аппаратного рестарта этого оборудования, ему открывается доступ к цепи дискретного вывода, управляющей его электропитанием.

Для каждого из этих пользователей создаётся индивидуальное дерево команд, в которое включается лишь минимальный набор команд, необходимый им для выполнения своих обязанностей. Это меню, как и полное меню конфигурации, может иметь древовидную структуру. Построение индивидуального дерева производится в узле `scheme`. Каждый узел пользовательского меню описывается в нём двумя компонентами:

- `@link` Указатель на узел полного командного дерева, начиная от корня, например:
 `.port.1-Wire.device.....circuit.5`
 — управление цепью дискретного вывода №5 на контроллере, подключённом по шине 1–Wire.
- `@help` Краткая подсказка, которая будет видна пользователю при выборе этого пункта меню, например:
 Power control for server BESH

ВНИМАНИЕ Если в качестве `@link` указана точка (`.`), то она означает корень дерева конфигурации. Пользователь с такой настройкой получает полный доступ к устройству, наравне с `nsg`. Создавать дополнительных пользователей с такими правами категорически не рекомендуется.

ПРИМЕЧАНИЕ В корне пользовательского дерева (`scheme.@link`) допускается указывать только промежуточный узел дерева. В дополнительных узлах (созданных через `_new` или `"+"`) можно указывать как промежуточные узлы, так и конечные параметры и команды.

ПРИМЕЧАНИЕ Символ `@` является специальным, поэтому при работе в оболочке `nsgsh` названия параметров `"@link"`, `"@help"` необходимо вводить в кавычках.

Внутри указанного узла пользователю будут доступны все содержащиеся в нём команды основного дерева. Чтобы исключить из доступной пользователю ветви некоторый ещё более узкий набор узлов, следует использовать узел `exclude`. В нём строится список исключаемых ветвей.

ВНИМАНИЕ Исключаемые узлы в списке `exclude` указываются по их положению в индивидуальном командном дереве пользователя. Например, в следующем примере:

```

scheme
: power
: : bir
: : : @link           = ".port.1-Wire.device.....circuit.1"
: : : @help          = "Power control for server No.1"
exclude
: 1                  = ".power.bir.circuit.1.off"
: 2                  = ".power.bir.circuit.1.toggle"

```

два исключения не позволяют пользователю выключить питание сервера полностью. С помощью оставшихся доступных команд (`on`, `drop` и `pulse`) он может только включать и рестартовать сервер.

Индивидуальное дерево конфигурации доступно пользователю при входе как через Web-интерфейс, так и через консольный интерфейс.

Особым случаем является вход на вспомогательные Web-страницы посредством кнопки . Для этого в `@link` следует указать соответствующий пункт меню и затем `_html`. Например, для пользователя, которому полагается видеть только страницу Web-мониторинга `uiTCP` и ничего больше, дерево меню будет выглядеть следующим образом:

```

scheme
: @link = ".tunnel.uitcp.show._html"

```

ПРИМЕЧАНИЕ Подсказки в пользовательских меню не используют механизм локалей. Для их корректного вывода необходимо, чтобы локаль администратора, в которой они создаются, совпадала с локалью, в которой будет работать пользователь. Если это невозможно, то не следует использовать в подсказке символы национальных алфавитов.

Каждому из создаваемых пользователей могут быть назначены права `ro` (*read-only*) или `rw` (*read-write*). Пользователь с правами *read-only* во всех случаях может только просматривать конфигурацию, статистику и состояние в доступных ему пунктах меню, а также исполнять разовые команды и изменять параметры этих команд, не сохраняемые в конфигурации (например, адрес и число посылок *ping*). Пользователь с правами *read-write* может становиться текущим администратором устройства (см. §1.2.5). В этом случае он может изменять настройки, в пределах своего дерева конфигурации, а также применять и сохранять изменения.

§1.5.4. Настройка средств управления

Все инструменты сетевого управления, с помощью которых осуществляется доступ к дереву конфигурации устройства NSG, являются прикладными службами IP. Настройка этих служб производится в меню `.services`. В их число входят серверы HTTP, HTTPS, Telnet, SSH, а также сервер SMS-управления и агент SNMP. (В текущей версии могут быть реализованы не все из перечисленных инструментов). Для каждого из них устанавливается, как минимум, состояние (включён/выключен) и номер порта TCP/UDP. Остальные настройки специфичны для каждого сервера. Подробно о настройке средств управления см. [Часть 4](#).

В заводской конфигурации устройства включены сервисы Telnet и HTTP, остальные выключены.

ПРИМЕЧАНИЕ Для работы сервера SSH необходимо сгенерировать ключи. Ключи хранятся в директории `/etc` наравне с другими элементами конфигурации. При переустановке программного обеспечения из сервисного режима (см. §1.8) ключи удаляются, их необходимо сгенерировать заново.

§1.5.5. Настраиваемая светодиодная индикация

В моделях, оснащённых настраиваемыми светодиодными индикаторами, эти индикаторы могут отображать состояние заданных программных объектов или тестовые сигналы. Настройка индикаторов выполняется в узле `.tools.led`.

Двухцветным индикаторам соответствуют два пункта в этом узле, `l1.green` и `l1.red`. Одновременное свечение обеих компонент создаёт жёлто-оранжевый цвет, однако использовать его следует с осторожностью, поскольку в зависимости от освещения, угла зрения и т.п. он может быть визуально близок к красному.

§1.5.6. Резервное копирование и восстановление текущей конфигурации

а) Резервирование дерева конфигурации в текстовом виде

Текущая конфигурация устройства может быть выведена в текстовом виде в оболочке `nsgsh` с помощью команды `_print`. Вывод может быть сохранён в текстовом файле и загружен обратно (или в другое устройство для клонирования типовой конфигурации) с помощью команды `_get`. Эти команды могут применяться как в корне конфигурационного дерева, так и в нижележащих узлах. Если применять их в корне дерева, то сохраняется и восстанавливается полная конфигурация устройства, введённая пользователем.

В Web-интерфейсе эквивалентом команд `_print` и `_get` является текстовый режим, включение и выключение которого производится кнопками  и . В текстовом режиме пользователь может копировать фрагмент конфигурации из окна Web-интерфейса, вставлять сохранённый ранее фрагмент, или редактировать его непосредственно в этом окне. (Ручное редактирование, во избежание ошибок, рекомендуется выполнять в минимальном объёме — установка уникального IP-адреса, имени устройства и т.п. отличий от типовой конфигурации.)

б) Резервирование конфигурации по сети (клиент)

В общем случае, конфигурация устройства представляет собой набор файлов, хранящийся в директории `/etc`. В их число могут входить, помимо `/etc/nsgconfig`, другие файлы, пользовательские скрипты, изменения в `/etc/inittab` для запуска опциональных компонент непосредственно из ОС Linux и т.п. В частности, это файл с хэшами паролей и, если используются, файлы сертификатов X.509. Таким образом, для манипуляций с конфигурациями возможны два варианта:

- Экспорт-импорт файла `/etc/nsgconfig` в текстовом виде. При этом сохраняется дерево конфигурации, но не вспомогательные файлы: пароли, сертификаты, пользовательские скрипты и т.п.
- Экспорт-импорт директории `/etc` целиком в виде архива. Всё содержимое директории упаковывается в формате `.tar.gz` и выгружается на указанный сервер или в локальный файл.

ПРИМЕЧАНИЕ Архив может быть прочтён любыми стандартными утилитами для архивирования. В общем случае, некоторые файлы можно даже изменить на ПК и запаковать обратно, но следует проявлять осторожность: некоторые утилиты для ОС Windows записывают архив в формате, несовместимым с версией упаковщика на устройстве NSG, поэтому файл не читается.

Управление конфигурациями производится в узле `.system.configurations` и может производиться как из консольного, так и из Web-интерфейса с помощью разовых команд:

```
etc-load
etc-store
nsgconfig-load
nsgconfig-store
```

Аргументом вышеперечисленных команд является место для сохранения конфигурации, указанное в формате URL. URL может включать в себя все компоненты, предусмотренные RFC 1738, а также нестандартный протокол `data://` (необязательные — отмечены квадратными скобками):

протокол://[имя[:пароль]@]хост[:порт]/[путь]/файл

где

протокол Один из поддерживаемых протоколов для передачи или сохранения файла:
 `tftp` } для экспорта/импорта на удалённый сервер
 `ftp` }
 `http` }
 `file` в указанный файл на устройстве NSG
 Параметр обязательный. Протокол HTTP допускается только для импорта конфигурации.

имя Имя пользователя (для FTP, HTTP), если необходимо.

пароль Пароль пользователя (для FTP, HTTP), если необходимо.

хост IP-адрес или доменное имя сервера. Параметр обязательный.

порт Номер порта TCP или UDP на сервере. Необходим, если используется нестандартный порт.

ПРИМЕЧАНИЕ Все пути указываются от корня сервера, в т.ч. от корня локальной файловой системы. В частности, для протокола `file` следует писать, например:

`file:///mnt/usb/mytest.conf`

Здесь `://` является стандартным разделителем, а третий `/` обозначает корень файловой системы на устройстве NSG. Для хранения конфигурации, естественно, необходимо использовать энергонезависимую память. В большинстве устройств (кроме серии NSG-1000) для записи доступна только директория `/etc` (в случае архивирования из неё исключаются файлы с другими конфигурациями). При наличии сменных носителей (USB Flash, SD card) резервные конфигурации могут сохраняться на них.

путь Путь к файлу с конфигурацией на сервере или на локальном устройстве. Может содержать произвольное число вложенных директорий.

файл Имя файла (кроме `data://`).

Например:

`ftp://basile:qwerty@gadukino.ru:21021/xlam/nsg/my_nsg.conf`

При загрузке файла `nsgconfig` следует учитывать, что это равносильно ручному внесению изменений в конфигурацию; чтобы сделанные изменения вступили в силу, необходимо выполнить команду `_apply`, а чтобы сохранить их — команду `_write`.

Аналогично, при импорте конфигурации целиком перезаписываются файлы в директории `/etc`; чтобы применить сделанные изменения, необходимо перезагрузить устройство. Дополнительно сохранять изменения вручную при этом *не следует*, поскольку при этом в `/etc/nsgconfig` будет заново записана действующая конфигурация вместо импортированной.

ВНИМАНИЕ Фактическая распаковка архива и перезапись `/etc` производятся в момент рестарта устройства. Если архивная конфигурация загружена на устройство, то у пользователя есть ещё один шанс отменить эти изменения — удалить её командой `remove-loaded-etc`.

в) Резервирование директории `/etc` по HTTP/HTTPS (сервер)

При использовании Web-интерфейса резервирование директории `/etc` полностью или одного только файла основной конфигурации `/etc/nsgconfig` возможно непосредственно средствами пользовательского Web-браузера. При этом устройство NSG выступает в качестве сервера HTTP или HTTPS. Используются следующие команды в `.system.configurations`:

```
etc-load-from-browser
etc-store-to-browser
nsgconfig-load-from-browser
nsgconfig-store-to-browser
```

В качестве аргумента для команд `load` необходимо выбрать файл, сохранённый на машине пользователя.

ПРИМЕЧАНИЕ Для приложений на основе SSL, использующих хэши сертификатов X.509 (в данной версии это `uitcp`), после импорта конфигурации следует выполнить операцию `rehash`.

§1.5.7. Пользовательские конфигурации

В NSG Linux 2.0 предусмотрена возможность позволяет создавать и хранить множественные конфигурации, например, для разных случаев применения устройства или для отладки сетевого решения. Помимо заводской (по умолчанию) и текущей конфигурации, пользователь может создавать альтернативные конфигурации, экспортировать и импортировать их. Для каждой пользовательской конфигурации создаётся отдельный узел в ветви `.system.configurations`. По умолчанию, она содержит единственный узел `CURRENT`, который относится к текущей конфигурации устройства и существует всегда.

В меню конфигурации содержатся два взаимосвязанных параметра: тип данных (файл `nsgconfig`, т.е. одно лишь конфигурационное дерево, либо заархивированная директория `/etc` целиком) и URL для её хранения. Разовые команды в данном меню позволяют сохранить конфигурацию в указанный URL, просмотреть её (без загрузки на устройство), загрузить, или удалить (точнее, записать в указанный URL пустой файл). В частности, команды, приведённые в предыдущем параграфе:

```
configurations.etc-store
configurations.etc-load
configurations.nsgconfig-store
configurations.nsgconfig-load
```

являются, по существу, синонимами для `configurations.CURRENT.store` и `configurations.CURRENT.load`.

В качестве URL допускается, помимо стандартных протоколов, нестандартный протокол `data` — запись конфигурации непосредственно в строку URL. Для сохранения новой конфигурации в таком виде следует указать просто `data://`. Данная возможность имеется только в конфигурациях, созданных пользователем, и только для хранения конфигурационного дерева.

§1.5.8. Очистка конфигурации

Удалить все настройки конфигурационного дерева можно любым из следующих способов:

- Нажать на кнопку  (или выполнить команду `_remove`) в корне дерева.
- Нажать на кнопку  в корне дерева, стереть всю конфигурацию и нажать кнопку .

ВНИМАНИЕ Вышеперечисленные действия очищают конфигурацию полностью, включая все IP-адреса, установленные по умолчанию. После применения такой пустой конфигурации устройство немедленно станет недоступным для управления. Поэтому очищать конфигурацию таким образом можно только для того, чтобы немедленно записать хотя бы минимальную работоспособную конфигурацию (например, адрес `192.168.1.1/24` для порта `eth0`).

Как и в остальных случаях, после изменения конфигурации необходимо использовать команду `_apply` или кнопку , чтобы изменения вступили в силу.

Все вышеприведённые действия очищают только основное дерево конфигурации и не затрагивают паролей, пользовательских скриптов, резервных конфигураций (см. следующий параграф) и т.п. Чтобы полностью восстановить заводские настройки, т.е. привести устройство в исходное состояние, необходимо выполнить перепрошивку программного обеспечения в сервисном режиме (см. §1.8).

§1.5.9. Автоматическое копирование конфигурации между устройствами

Для упрощения синхронизации между несколькими устройствами в NSG Linux 2.0 имеется механизм автоматического переноса заданных узлов конфигурации с основного устройства на резервное. Данная функция особенно полезна, если конфигурация громоздкая и имеет свойство модифицироваться со временем; наиболее типичный пример — центральный сервер VPN любого типа, который обслуживает большое число клиентов, причём список клиентов постепенно изменяется.

Автоматическое копирование производится с помощью клиента SSH и аутентификации на удалённом устройстве по ключу (чтобы избежать ручного ввода пароля). Работа с клиентом SSH, управление его ключами и настройка копирования конфигурации описаны в [Части 4](#). Собственно копирование производится вручную с помощью разовой команды, но эта команда может запускаться и автоматически по срабатыванию таймера или планировщика задач. Подробно о настройке обработчика событий в целом и, в частности, таймера и планировщика, также см. [Часть 4](#).

§1.5.10. Восстановление заводской конфигурации и модернизация программного обеспечения

а) Обновление программного обеспечения по сети (клиент)

Загрузка нового программного обеспечения на работающую систему выполняется в узле `.system.software`. При этом файл с новым программным обеспечением загружается с указанного сервера TFTP, FTP или HTTP и устанавливается на устройство. Обновление возможно двумя способами:

- Полное обновление (*full-update*). Процедура требует достаточно большой объём свободной оперативной памяти, в которой временно размещается принимаемый файл. Если устройство имеет ограниченные аппаратные ресурсы (серия NSG–600) или исполняет приложения с большим потреблением памяти (напр. BGP), то полное обновление может быть невозможно, процедура завершается с ошибкой на этапе загрузки файла. Существующая версия программного обеспечения при этом остаётся в неизменном виде.
- Частичное обновление от некоторой опорной версии. При этом загружается только разностный файл между опорной и новой версиями. Эта процедура предназначена для случаев, когда полное обновление невозможно. Более новый *service pack* устанавливается поверх всех предыдущих *service packs* от той же опорной версии. Смена самой опорной версии возможна только при полном обновлении.

Для выполнения процедуры необходимо указать полный URL файла (при необходимости он может включать имя и пароль пользователя, номер порта). Формат URL описан в §1.5.6, за исключением того, что поддерживаются только стандартные протоколы TFTP, FTP, HTTP. Команда `launch` или `install`, соответственно, загружает файл, проверяет его и записывает в энергонезависимую память; для выполнения операции требуется подтверждение.

Текущая конфигурация устройства сохраняется. Говоря более точно, при обновлении ПО данным способом заменяется содержимое всех директорий, кроме `/etc`, а также отдельные файлы и поддиректории в `/etc`. Перечень этих файлов и директорий может варьироваться в различных версиях NSG Linux 2.0. В частности, могут перезаписываться файлы `inittab`, `fstab` и директории `init.d`, `nsd`. Гарантированно сохраняется файл `nsconfig` и содержимое следующих директорий:

- `/etc/private/` — выделенная директория для хранения пользовательских файлов, скриптов (в т.ч. стартовых скриптов в `/etc/private/init.d/`) и т.п.
- директории, назначенные по умолчанию для хранения сертификатов X.509 в различных приложениях (`/etc/uitcp/certs/` и т.п.)

Все дополнительные файлы, скрипты, сертификаты и т.п., самостоятельно созданные в системе пользователем, следует хранить именно в этих директориях.

По завершении обновления требуется перезагрузка устройства.

ВНИМАНИЕ При нехватке оперативной памяти (в особенности, на устройствах серии NSG–600) рекомендуется перед обновлением программного обеспечения:

- убедиться, что устройство доступно по Telnet или SSH
- отключить Web-управление (`.services.http.enable = false`)
- сохранить конфигурацию
- рестартовать устройство.

Процедура обновления в этом случае выполняется средствами консольного интерфейса.

б) Обновление программного обеспечения в сервисном режиме и полная очистка конфигурации

Специальный сервисный режим предназначен для загрузки нового программного обеспечения и полного восстановления заводской конфигурации устройства. Этот режим требует физического доступа к устройству и остановки его работы. Загрузка и установка нового программного обеспечения производится средствами пользовательского Web-браузера, устройство NSG выступает в качестве сервера HTTP. Текущая конфигурация устройства удаляется полностью, включая пароли, сертификаты, пользовательские скрипты и т.п. После завершения обновления производится перезагрузка устройства. Подробно о сервисном режиме см. §1.8.

ВНИМАНИЕ Откат устройства на более низкую версию ПО (*downgrade*) настоятельно рекомендуется выполнять только из сервисного режима. Возможность отката непосредственно из рабочего режима не гарантируется, особенно при значительном расхождении в номерах сборок.

ПРИМЕЧАНИЕ Имя и формат файла с программным обеспечением (`*.bin`, `*.tar.gz` и др.) являются специфическими для конкретной серии устройств. В любом случае, действия пользователя не зависят от них и являются одинаковыми для всех устройств.

§1.5.11. Проверка обновлений программного обеспечения

Установка нового программного обеспечения — как полного, так и всех возможных частичных обновлений — производится администратором только вручную, после пробной эксплуатации на ограниченной опытной зоне, в удобное время для минимизации влияния на работу системы. Полностью автоматическое обновление, без контроля со стороны администратора, не может быть рекомендовано для критически важных систем.

Новые версии программного обеспечения NSG доступны бесплатно на официальном сайте <http://www.nsg.ru> в разделе "Техническая поддержка/NSG Linux 2.0", а также на FTP-сайте <ftp://nsg.net.ru/pub/nsg-linux/binary/2.0/>. Сообщения о выпуске новых версий в статусе Release Candidate и стабильных версий публикуются в разделе новостей на сайте компании и рассылаются подписчикам по электронной почте. Сетевому администратору устройства NSG настоятельно рекомендуется зарегистрироваться на форуме NSG (это не накладывает на него обязанности что-либо писать в форум) и включить в своём профиле рассылку новостей.

Для своевременного информирования администратора о выпуске новых версий можно использовать также механизм автоматической проверки обновлений. Он проверяет номера крайних версий, указанные в определённом файле на сайте, и сравнивает их с текущей версией. Для выполнения проверки необходим доступ в Интернет, включая настроенного клиента DNS.

При желании можно запланировать регулярную проверку наличия обновлений, используя механизмы планировщика задач и обработчика событий (см. [Часть 4](#)). Задача создаётся и удаляется разовой командой `add-to-/remove-from-scheduler` в узле `.system.software.check-update`. Результаты проверки записываются в журнал, который администратор может просматривать в удобное для себя время. При необходимости можно добавить в эту задачу скрипт, который будет проверять последние строки журнала и, например, посылать уведомления по SMS и/или электронной почте.

§1.5.12. Перезагрузка устройства

Для программной перезагрузки устройства необходимо выполнить команду `.system.reboot`. Для выполнения команды необходимо подтверждение. В `nsgsh` следует ввести в ответ на запрос слово `yes` (полностью и с соблюдением регистра). В Web-интерфейсе необходимо набрать `yes` в поле ввода данной команды, и затем нажать кнопку ▶.

§1.5.13. Сторожевой таймер

Для автоматического восстановления работоспособности устройства в случае сбоев в работе программного обеспечения предусмотрен программный сторожевой таймер — процесс, который следит за активностью `nsgconfd`, основного процесса, управляющего системой. Если `nsgconfd` зависает и не рестартует самостоятельно, то по истечении заданного интервала ожидания программный таймер сохраняет доступную отладочную информацию (`ps`, `top` и т.п.) в файле `/etc/postmortem.dump` и затем рестартует устройство. Полученная информация может быть полезна для анализа и поиска проблем.

Уменьшение интервала работы таймера позволяет быстрее реагировать на зависания системы. Однако чрезмерно малый интервал может привести к ложным срабатываниям и рестарту системы без реальной на то необходимости. Увеличение интервала приводит к более медленной реакции системы. Значение по умолчанию рекомендуется для большинства конфигураций, ручная настройка интервала требуется только при явной необходимости в этом.

§1.6. Виртуальные маршрутизаторы

§1.6.1. Общие сведения о механизме виртуальных маршрутизаторов

Система виртуальных маршрутизаторов в устройствах NSG реализована на основе решения Linux *network namespaces* (*netns*). Она позволяет организовать в одном физическом устройстве несколько дополнительных виртуальных маршрутизаторов. При этом каждый из маршрутизаторов (физический или виртуальный) имеет свой набор портов и интерфейсов, свой стек IP, свою таблицу маршрутизации, свою конфигурацию и т.п. Разграничение между устройствами определяется следующими правилами:

- Все логические маршрутизаторы функционируют независимо друг от друга и могут управляться одновременно, каждый своим администратором.
- Факт существования каждого виртуального маршрутизатора и его "аппаратный" состав определяются администратором физического устройства. Каждый порт или иной неделимый физический или логический объект (напр. встроенный коммутатор или клиент NTP) может принадлежать одному и только одному маршрутизатору (физическому или логическому).
- Каждый виртуальный маршрутизатор имеет свою полную конфигурацию, включая дерево конфигурационных параметров, пароли, ключи, сертификаты, пользовательские скрипты и другие файлы.
- На каждом виртуальном маршрутизаторе существуют свои пользователи *nsg* и *root*, а также могут существовать дополнительные пользователи, созданные им. Всем им доступен вход только в их собственный маршрутизатор.
- Пользователи *nsg* и *root* физического маршрутизатора могут со своим именем и паролем входить в любой из виртуальных маршрутизаторов через Web-интерфейс (см. §1.6.3). Это же относится к дополнительным пользователям, если на виртуальном маршрутизаторе существует пользователь с таким же именем (но не обязательно с таким же паролем).
- Пользователь *root* физического маршрутизатора при доступе через консольную оболочку имеет неограниченные права доступа к конфигурации как физического маршрутизатора (директория */etc*), так и к конфигурациям виртуальных маршрутизаторов, которые монтируются для него в директории */etc/netns/имя*.
- Операции импорта-экспорта и применения конфигурации виртуального маршрутизатора относятся только к нему самому.
- Экспорт-импорт и применение полной конфигурации (*/etc*) физического устройства включают в себя также конфигурации всех виртуальных устройств. Экспорт-импорт и применение одного только дерева конфигурации (*/etc/nsgconfig*) физического маршрутизатора относятся только к нему самому (если при этом не затрагивается "аппаратный" состав виртуальных маршрутизаторов).

§1.6.2. Создание виртуальных маршрутизаторов

Для создания виртуального маршрутизатора необходимо добавить его имя в список *.netns*. Далее в этом узле администратор может, с помощью разовых команд, рестартовать виртуальное устройство (аналогично грубому выключению питания для физического устройства) и посмотреть журнал его работы как цельного объекта в рамках физического устройства. Другие способы вмешательства в работу виртуального маршрутизатора не предусмотрены.

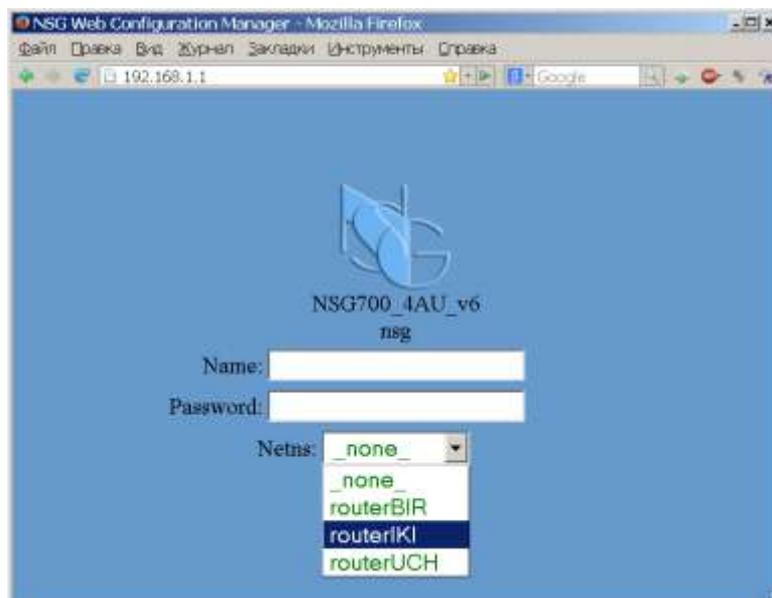
Далее, если в устройстве создан хотя бы один виртуальный маршрутизатор, то в меню объектов, которые могут быть ему переданы, появляется свой параметр *netns*. Значением параметра может быть один из списка существующих виртуальных маршрутизаторов. Если оно не установлено, то объект принадлежит физическому устройству и управляется его администратором. Если установлено имя одного из виртуальных маршрутизаторов, то данный объект включается в его состав; при этом в конфигурации виртуального устройства появляется соответствующий узел, а в конфигурации физического устройства для него недоступны никакие другие узлы, кроме *.netns*. Данная процедура относится к следующим объектам:

- Физическим портам (*.port.**), псевдо-интерфейсам (*.pseudo-interface.**) и клонам Ethernet-подобных интерфейсов (*macvlan*).
- Встроенному аппаратному коммутатору (*.ethernet.switch*)
- Реализации NetFlow в ядре (*ip.filter.NETFLOW, ipv6.filter.NETFLOW*)
- Отдельным общесистемным объектам (*.system.**) — например, обновление ПО или управление системным временем (*ntp, timezone*) могут быть поручены администратору одного из виртуальных устройств
- Настраиваемым светодиодным индикаторам (*.tools.led*)

Порты RS–232, 1–Wire и т.п. являются, по определению, однопоточными объектами и не могут использоваться совместно несколькими виртуальными маршрутизаторами одновременно. Широковещательные Ethernet, в отличие от них, возможно использовать одновременно в нескольких маршрутизаторах. Для этого следует создать необходимое число их клонов (MAC VLAN, см. Часть 2) и передать по одному клону в каждый из маршрутизаторов. При этом все виртуальные маршрутизаторы будут совместно использовать общую полосу пропускания данного физического порта. Это же относится и к другим Ethernet-подобным объектам: мостам (*bridge groups*), агрегированным портам (*bond groups*), портам WLAN и WMAN, туннелям GRE с инкапсуляцией Ethernet-over-GRE, туннелям OpenVPN в режиме *tap*, и т.п.

§1.6.3. Вход в виртуальный маршрутизатор и особенности его настройки

Вновь созданный виртуальный маршрутизатор имеет, как правило, абсолютно пустую конфигурацию по умолчанию. Получить к нему доступ возможно только через страницу входа в физическое устройство. Если на нём созданы виртуальные устройства и они не пустые (т.е. им переданы какие-либо объекты), то дополнительно к обычным полям (имя и пароль) на этой странице появляется меню со списком этих устройств. Если в данном списке не выбран ни один из них, осуществляется вход в физическое устройство, если выбран — то в соответствующий виртуальный маршрутизатор.



По умолчанию, на каждом из виртуальных маршрутизаторов имеется обычный набор пользователей: *nsg* с пустым паролем и *root* с неизвестным случайным паролем. Войти на виртуальный маршрутизатор можно также с реквизитами пользователей физического устройства.

После входа в виртуальный маршрутизатор необходимо настроить на нём хотя бы один интерфейс, через который можно будет получить к нему доступ напрямую. В отличие от физического устройства, интерфейсы, включённые в состав виртуального маршрутизатора, находятся в двух узлах:

- физические порты — в узле `.port`
- клоны Ethernet-подобных интерфейсов — в узле `.pseudo-interfaces`

В остальном дерево конфигурации выглядит аналогично таковому для физического устройства, за исключением того, что в нём отсутствуют узлы `netns` (т.е. создавать вложенные виртуальные устройства уже невозможно).

После того, как хотя бы одному из интерфейсов виртуального маршрутизатора назначен IP-адрес, дальнейший доступ к устройству и управление им могут производиться обычным образом по этому адресу.

§1.7. Безопасность устройства

§1.7.1. Общие рекомендации

Для безопасной работы любого оборудования в сетях общего пользования необходимо соблюдать, как минимум, следующие правила безопасности:

1. Назначать уникальный и неочевидный пароль для входа на устройство. В устройствах под управлением NSG Linux 2.0 необходимо назначить пароль, как минимум, для пользователя `nsg`. Пароли остальных пользователей необходимы только в том случае, если такие пользователи нужны для работы. В частности, пользователь `root` в заводской конфигурации имеет непустой, но неизвестный никому (в т.ч. и компании NSG) пароль, поэтому вход в систему с этим именем невозможен.
2. Отключить все службы удалённого управления, в которых нет необходимости. В заводской конфигурации устройств под управлением NSG Linux 2.0 включены серверы Telnet и HTTP, отключены SSH, HTTPS, SNMP, SMS-управление. Как правило, этот набор служб целесообразно перенастроить.
3. Использовать безопасные протоколы (SSH, HTTPS) вместо их открытых аналогов (Telnet, HTTP).
4. Не использовать устаревшие версии безопасных протоколов, поскольку на сегодняшний день они не являются достаточно надёжными. В частности, в устройствах под управлением NSG Linux 2.0 используется только SSH v2; версия SSH v1 запрещена. Для STunnel, HTTPS, *u*TCP, OpenVPN и других безопасных туннелей на основе SSL, в соответствии с IETF RFC-7568 и отраслевыми стандартами (в частности, PCI DSS 3.1), следует использовать только TLS.
5. Запретить фильтрами входящий трафик (как транзитный, так и локальный) со стороны публичной сети по всем неиспользуемым протоколам и портам, в особенности по протоколам сетевого управления.

Как следствие п.1 из данного списка, необходимо иметь резервную конфигурацию устройства и хранить её в надёжном месте. В случае утраты пароля администратора восстановить доступ к устройству можно только путём сброса всей конфигурации в заводские настройки; это относится и к устройствам NSG. После этого необходимо восстановить конфигурацию из резервной копии — но за исключением старого пароля!

Данные рекомендации относятся также к изолированным корпоративным и технологическим сетям, поскольку нельзя исключить, что злоумышленник, единожды проникнув в сеть через слабо защищённое устройство или брешь в человеческом факторе, пойдёт по сети дальше с устройства на устройство.

Помимо этого, для замкнутых сетей, не предполагающих выход в сети общего пользования, рекомендуется использовать тот факт, что передача данных должна производиться только по заранее известным адресам, протоколам и номерам портов. К таким системам могут относиться, например, удалённые офисы в корпоративной сети VPN, сообщающиеся только с головным офисом без непосредственного выхода в Интернет; банкоматы и POS-терминалы, соединяющиеся только с процессинговым центром; удалённые технологические площадки, работающие только с центром управления, и т.п. В подобных случаях рекомендуется дополнительно:

1. Не создавать маршрутов по умолчанию. Вместо них достаточно создать только ограниченное число маршрутов на нужные IP-адреса и подсети.
2. Явным образом разрешить трафик с/на нужные IP-адреса, протоколы и/или номера портов, и запретить все остальные пакеты.

Вышеперечисленные правила безопасности отчасти дублируют друг друга. Тем не менее, рекомендуется использовать их комплексно, с тем, чтобы возможные человеческие ошибки в одном из них блокировались остальными механизмами безопасности.

§1.7.2. Аутентификация пользователей

Аутентификация пользователей при входе в систему может производиться разными способами в зависимости от конкретной задачи. Настройка аутентификации производится в узле `.system.aaa`. В данной версии предусмотрены отдельные политики аутентификации для следующих случаев:

- Вход в систему через консольный порт или по Telnet. Эта же политика используется при входе в Web-интерфейс как по HTTP, так и по HTTPS. (Если для HTTPS требуются какие-либо дополнительные инструменты аутентификации, то их следует использовать на уровне туннеля STunnel, например, двусторонний обмен сертификатами X.509).
- Вход в систему по SSH.
- Подключение со стороны сети к порту RS-232 устройства, или к его эквиваленту (внутреннему порту сотового или dial-up модема и т.п.) по Reverse Telnet, без доступа к самому устройству NSG.
- Подключение удалённого пользователя PPP, PPPoE, PPTP к сети IP через устройство NSG в качестве сервера доступа, без доступа к самому устройству NSG.
- Политика входа по умолчанию. Используется в том случае, если для требуемой политики не определено ни одного способа аутентификации.

Каждая политика может содержать несколько способов аутентификации, которые используются в порядке их нумерации. Способ определяет источник аутентификации (локальный список пользователей на данном устройстве, централизованный сервер RADIUS или TACACS+) и дополнительные параметры для обращения к нему (IP-адрес, порт, ключ и т.п.). Если используется несколько серверов аутентификации (основной и резервные), то для каждого из них следует создать отдельный способ.

Специальный способ аутентификации `deny` запрещает вход пользователей в систему по данной политике. Он всегда даёт отрицательный результат и может использоваться, например, для того, чтобы временно запретить тот или иной вход, не удаляя существующих настроек.

Если попытка аутентифицироваться по некоторому способу даёт положительный результат, аутентификация считается завершённой успешно. Если нет — то делается попытка по следующему способу. Если все способы, входящие в данную политику, дали отрицательный результат, то аутентификация считается неудачной.

ВНИМАНИЕ Независимо от используемого способа аутентификации (в т.ч. RADIUS и TACACS+), необходимо создать в системе локального пользователя (`.system.users`) с данным именем. Это отдельная операция, поскольку аутентификация — лишь одна из процедур и атрибутов, применяемых к пользователю. Подробно о процедуре создания пользователей см. §1.5.3.

Если требуется, чтобы пользователи аутентифицировались различными способами, например, часть — локально, а другие — через RADIUS, то для вторых следует задать в первом способе фиктивные пароли (произвольные длинные случайные строки, которые немедленно забыть). В этом случае система не сможет аутентифицировать пользователя первым способом и пойдёт на второй способ, в котором установлен уже реальный пароль.

§1.7.3. Аудит системы

Для контроля за работой пользователей в системе предусмотрена служба системного аудита, построенная на основе стандартных компонент ОС Linux: демона `auditd` и утилит для его запуска, конфигурации, просмотра результатов и генерации отчётов. Настройка аудита производится в узле `.system.audit`.

Настройка собственно демона `auditd` производится в узле `conf` в терминах стандартного файла конфигурации `auditd.conf`. Переменные конфигурации, отсутствующие в списке по умолчанию, можно добавлять с помощью команды `_new` или `+`. Настройка правил аудита производится в узле `rules` в терминах файла `audit.rules`.

ВНИМАНИЕ По умолчанию, файл аудита пишется во временную директорию `/var` на RAM-диске в оперативной памяти устройства. При перезагрузке устройства содержимое этой директории теряется. Чтобы сохранить его (в частности, чтобы злоумышленник не мог скрыть следы своих действий, инициировав перезагрузку устройства), следует перенаправить журнал в другой файл, расположенный на внешнем носителе (USB Flash, SD card) или на HDD (в директорию `/etc` либо в отдельный раздел), в зависимости от их наличия на данном устройстве. Подробные рекомендации о выборе места для хранения журналов см. в описании службы `syslog` (см. §1.2.13).

В узле `show` можно просмотреть текущую конфигурацию службы аудита (включая настройки по умолчанию), журнал аудита в сыром виде, а также сгенерировать отчёт средствами утилиты `augreport`. Подробную информацию о службе системного аудита в ОС Linux см. в *man pages* по вышеперечисленным утилитам и файлам.

§1.8. Сервисный режим

§1.8.1. Общие операции в сервисном режиме

Сервисный режим — специальный служебный режим работы, предназначенный для выполнения ответственных операций и/или восстановления работоспособности системы. В сервисном режиме вместо основного программного обеспечения запускается резервная инсталляция NSG Linux 2.0 (Recovery Linux), хранящаяся на отдельном разделе Flash ROM. Данный раздел записывается единственный раз при изготовлении устройства и не перезаписывается пользователем, за исключением отдельных особых случаев.

Сервисный режим следует использовать в следующих случаях:

- Возникновение неработоспособной конфигурации устройства (как правило, вследствие ручного редактирования файла конфигурации).
- Утрата паролей для обоих административных пользователей.
- Утрата IP-адресов, назначенных устройству (при отсутствии консольного порта или его перенастройке для передачи пользовательских данных).
- Повреждение основного раздела Flash ROM или раздела с конфигурацией (например, вследствие отключения электропитания в момент обновления программного обеспечения штатными средствами).

Для входа в сервисный режим на большинстве устройств NSG необходимо:

1. Выключить электропитание устройства.
2. Нажать защищённую кнопку Service, Cold Start (CS) или включить микропереключатель Cold Start, в зависимости от модели устройства.
3. Удерживая кнопку Service нажатой, включить электропитание устройства.
4. Через 5 сек. отпустить кнопку Service.

При старте системы в сервисном режиме устройство всегда имеет заводскую конфигурацию. Пользовательская конфигурация полностью игнорируется. В частности:

- Порту eth0 назначается IP-адрес 192.168.1.1/24.
- Вход в систему возможен только через Web-интерфейс по протоколу HTTP.
- Локали не используются, интерфейс поддерживает только язык международного делового и технического общения.

Исключением являются устройства NSG-1000/GW, NSG-1000e. Для входа в сервисный режим на этих устройствах необходимо:

1. Выключить электропитание устройства.
2. Подключить клавиатуру к разъёму Kbd или к любому из портов USB.
3. Включить электропитание устройства.
4. После того, как на порту Eth0 загорится светодиод:
 - NSG-1000e — Act/Lnk (левый, жёлтый)
 - NSG-1000/GW — Lnk (левый)нажимать на подключенной к устройству клавиатуре клавишу "0" с интервалом 1 раз в 2 секунды в течение 30 сек. (В дальнейшем на этой клавиатуре нажимать клавиши не потребуется, она необходима только для запуска устройства в сервисном режиме.) Через 15–20 сек после этого установить HTTP-соединение с устройством по адресу 192.168.1.1.
5. Если вход на устройство по HTTP невозможен, это значит, что устройство загрузилось в обычном режиме вместо сервисного. Следует выключить устройство и повторить процедуру. Для контроля можно подключить монитор к разъёму VGA или DVI-D и дождаться меню из 4 пунктов, где пунктом 0 будет вход в сервисный режим. Меню держится 4 сек, в течение этого времени нужно нажать клавишу "0".

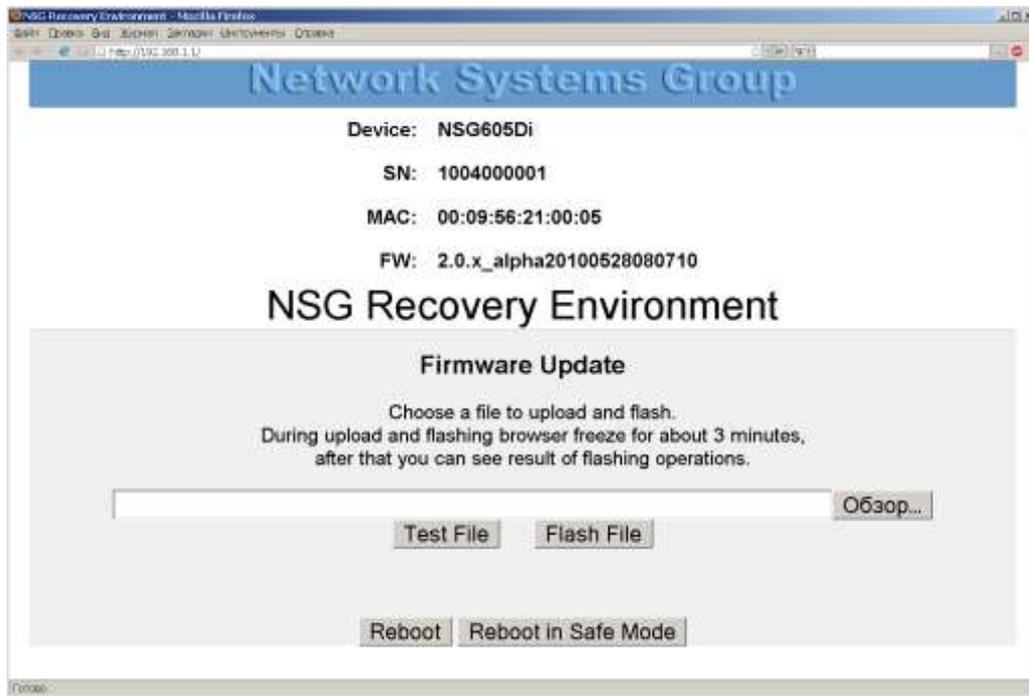
ПРИМЕЧАНИЕ Другие клавиши выбора для данной серии устройств:

- 1 Загрузка без отработки конфигурации (Safe Mode, см. ниже)
- 2 Загрузка в штатном режиме без поддержки консольного порта
- 3 Загрузка в штатном режиме с поддержкой консольного порта

По умолчанию, после задержки в 4 сек. выбирается пункт 2 или 3, в зависимости от опции, установленной на странице расширенной конфигурации сервисного ПО (см. след. параграф).

В сервисном режиме доступен Web-интерфейс с ограниченным набором функций. Он позволяет загрузить файл с программным обеспечением, проверить и установить его. При этом область памяти, отведённая для хранения основного программного обеспечения и его конфигурации, перезаписывается полностью. Текущая конфигурация системы безусловно утрачивается (в отличие от переустановки программного обеспечения штатными средствами на работающей системе, см. §1.5.10).

ПРИМЕЧАНИЕ Имя и формат файла с программным обеспечением (*.bin, *.tar.gz и др.) являются специфическими для конкретной серии устройств. В любом случае, действия пользователя не зависят от них и являются одинаковыми для всех устройств.



Восстановление заводской конфигурации, как отдельная операция, в NSG Linux 2.0 не предусмотрено и является частью процедуры переустановки программного обеспечения в сервисном режиме.

Для выполнения процедуры следует:

1. Сохранить файл с новым программным обеспечением на компьютере пользователя. Имя файла может быть любым.
2. Подключиться к устройству по адресу 192.168.1.1, используя любой стандартный Web-браузер. Имя и пароль для входа в Recovery Linux не требуются. В верхней части окна выводится информация об устройстве и текущей версии программного обеспечения.
3. С помощью кнопки "Обзор" выбрать файл с новым программным обеспечением.
4. С помощью кнопки "Flash File" загрузить файл на устройство, проверить его целостность и записать в энергонезависимую память устройства. При этом полностью удаляются прежние программное обеспечение и конфигурация. Процедура записи может занимать время до 3 минут, в течение которого устройство не отвечает на запросы. По завершении записи выводится информация о результате.
5. После сообщения о завершении процедуры — рестартовать устройство кнопкой Reboot или выключением питания.

Дополнительно, кнопка "Test File" позволяет загрузить файл на устройство и проверить его целостность (без записи в энергонезависимую память).

Кнопка "Reboot" предназначена для перезагрузки устройства в обычном режиме.

Кнопка "Reboot in Safe Mode" позволяет загрузить основное программное обеспечение NSG Linux 2.0 без отработки конфигурации. (Кроме устройств серии NSG-1000, в которых эта функция реализована через меню загрузчика, см. выше.) Это может потребоваться в том случае, если в конфигурации имеются ошибки (вероятнее всего, в результате ручного редактирования файла /etc/nsgconfig) и устройство не может стартовать обычным образом.

ВНИМАНИЕ При переустановке программного обеспечения в сервисном режиме удаляются полностью все элементы конфигурации, включая пароли пользователей, ключи, сертификаты и т.п. После обновления их необходимо установить или сгенерировать заново.

§1.8.2. Расширенные настройки сервисного режима

На отдельных типах устройств сервисный режим предусматривает некоторые специфические настройки, которые выполняются за рамками основного ПО NSG Linux 2.0 и определяют его дальнейшую установку и работу. Вид основной страницы сервисного режима для устройств NSG-1000e, NSG-1000/GW показан на рисунке.



Специфическими для данной серии устройств являются следующие опции:

Forced partitioning for NSG Root disk

По умолчанию для всех устройств, при установке нового ПО из сервисного режима производится переразбиение и форматирование устройства хранения данных, на которое это ПО будет устанавливаться. (Если только это не то же самое устройство, на которое установлено ПО сервисного режима.) Это имеет целью исправить возможные ошибки на носителе, исключить сбойные области и т.п. Однако данная серия устройств включает в себя модели, оснащённые HDD, и для них переразбиение может быть нежелательно, поскольку NSG Linux 2.0 занимает только часть диска, а в свободной области пользователь мог создать свои собственные разделы, например, для хранения журналов и статистики. В этом случае, чтобы сохранить их, необходимо установить для данной опции значение no — установщик не будет переразбивать диск, а только отформатирует первый раздел с NSG Linux 2.0 и запишет на него новую версию.

NSG Root parameter >

Установка дополнительных параметров для вызова ядра NSG Linux 2.0.



В текущей версии предусмотрен один настраиваемый параметр — системный раздел NSG Linux 2.0. Как правило, устройства серии NSG-1000 имеют два накопителя данных, из которых на первом (обозначаемом `/dev/sda`) первый раздел (`/dev/sda1`) используется для хранения ПО сервисного режима, а основное ПО NSG Linux 2.0 устанавливается на первый раздел второго диска (`/dev/sdb1`). При необходимости оно может быть переустановлено на `/dev/sda2` — например, если второй диск (это может быть HDD, SSD или DOM — не существенно) отсутствует, неисправен, или пользователь желает занять его целиком под собственные разделы.

System console settings >

Установка параметров консольного порта. Выбор консольного порта, его наличие, значения по умолчанию зависят от конкретной модификации устройства.



Name of serial port

Имя консольного порта. Если на устройстве имеется несколько портов, которые могут быть полностью использованы в качестве консольных, то в данном пункте предлагается меню для выбора. Если такой порт только один, то он выводится безальтернативно, данный пункт имеет справочный характер. Если данная модификация устройства не имеет COM-портов, то в качестве консольного может использоваться переходник USB-COM, входящий в комплект устройства; предполагается, что такой переходник один, поэтому его имя всегда `/dev/ttyUSB0`.

Speed of serial port

Скорость передачи данных в порту, бит/с. При этом формат асинхронной посылки всегда `8n1`.

Output to serial port

Консольный порт выполняет, по существу, две различные функции: вывод диагностических сообщений ядра Linux и работу пользователя в терминальном режиме. В NSG Linux 2.0 эти функции включаются раздельно. По умолчанию, вывод диагностики отключён для совместимости с ранними версиями NSG Linux 2.0, а также для удобства работы.

Login via serial port

Возможность входа пользователя через данный физический порт для работы в терминальном режиме. По умолчанию, если консольный порт задействован при старте системы, то он используется только в этом качестве.

Default booting

Режим загрузки NSG Linux 2.0 по умолчанию: без поддержки консольного порта или с ним. Если система стартует без поддержки консольного порта, то вышеописанные опции не имеют смысла. Режим загрузки можно также выбрать разово вручную при старте системы (см. примечание в предыдущем параграфе).

На обеих дочерних страницах кнопка "Default value" устанавливает для всех параметров значения, принятые для данной модификации устройства по умолчанию. Кнопка "Save" сохраняет сделанные изменения. Кнопка "Home" возвращает на главную страницу сервисного режима.

ПРИМЕЧАНИЕ Если описанные выше опции отсутствуют в меню, это означает, что ПО сервисного режима заведомо устарело и требует обновления. Обновление может требоваться также и по другим причинам; процедура обновления описана в §1.8.4.

§1.8.3. Установка NSG Linux 2.0 на устройства NSG–700

Программное обеспечение NSG Linux 2.0 может быть установлено, по усмотрению пользователя, на устройства серии NSG–700, поставленные ранее с версией NSG Linux 1.0. Для использования NSG Linux 2.0 необходимо только, чтобы устройство было оснащено кнопкой ColdStart (CS) на лицевой панели. Устройства ранних выпусков, не оснащённые данной кнопкой, в NSG Linux 2.0 не поддерживаются.

ПРИМЕЧАНИЕ В части устройств "промежуточной" версии кнопка CS была физически установлена, но не выведена на лицевую панель. Чтобы достоверно убедиться в её наличии или отсутствии, следует снять крышку устройства; кнопка находится рядом с кнопкой "Reset". Если она физически присутствует, для её использования достаточно проколоть иглой наклейку на лицевой панели. Если кнопка не напаяна, но имеется разводка под неё, можно замкнуть эти две контактные площадки металлическим предметом.

Процедура перехода на NSG Linux 2.0 является полностью обратимой для пользователя. Для восстановления NSG Linux 1.0 следует воспользоваться процедурой полной переустановки загрузчика U–Boot средствами низкоуровневого аппаратного загрузчика (см. документ NSG: *Мультипротокольные маршрутизаторы NSG. Программное обеспечение NSG Linux 1.0. Руководство пользователя. Часть 1. Приложение 1–Б.*) и процедурой установки NSG Linux 1.0 средствами U–Boot (см. там же).

ПРИМЕЧАНИЕ Замена программного обеспечения, при условии корректного её проведения, не создаёт угрозы необратимого повреждения устройства и не влечёт за собой утрату гарантийных обязательств NSG.

Конфигурация версий NSG Linux 1.0 и 2.0 несовместима друг с другом и при смене программного обеспечения утрачивается полностью. Настоятельно рекомендуется сохранить предыдущую конфигурацию в файле и на бумаге, чтобы затем воспроизвести её средствами нового программного обеспечения.

Установка NSG Linux 2.0 производится в следующем порядке:

1. Загрузить устройство в NSG Linux 1.0 (желательно *build 5* или выше) и уточнить, какая именно аппаратная версия шасси идентифицируется автоматически, командой `chassis ?`. Эта информация потребуется далее на шаге 14. Ориентироваться на версию, написанную на материнской плате (5 либо 6 без дополнительных букв), следует только в старых устройствах с более старыми версиями NSG Linux. Шасси с материнской платой V7 или V7R2, но без дополнительного порта Ethernet 10Base–T, следует идентифицировать как *ver.6t* (без порта USB, основная часть выпуска) либо *ver.6u* (с портом USB на передней панели).
2. Обеспечить гарантированное электропитание устройства в процессе выполнения операции.
3. Сохранить файл Recovery Linux (`nsg700recovery.bin`) на сервере TFTP, доступном по локальной сети, и файл с рабочей версией NSG Linux 2.0 для данной серии (`nsg700-image.bin`) на ПК. Проверить контрольную сумму MD5 файла Recovery.
Назначить серверу адрес 192.168.0.250. Если это невозможно, то другие адреса серверу и устройству NSG могут быть назначены ниже на шаге 6.
4. Подключить любой из встроенных портов Fast Ethernet устройства к локальной сети, консольный — к COM-порту ПК с параметрами:

Baud Rate	9600
Parity	None
Data Bits	8
Stop Bits	1
Flow Control	Hardware

5. Включить питание устройства. В программе эмуляции терминала войти в меню загрузчика U–Boot. Для этого необходимо нажать любую клавишу в течение 5 сек. после появления подсказки:

Hit any key to stop autoboot: <секунды>

Будет выведено приглашение:

u-boot>

6. Если установленные по умолчанию адреса 192.168.0.250 (сервер) и 192.168.0.1 (устройство NSG–700) не подходят для данной сети, установить нужные адреса с помощью переменных окружения в соответствии с фактической конфигурацией сети:

```
setenv ipaddr <ip-адрес устройства NSG>
setenv serverip <ip-адрес сервера TFTP>
```

7. Проверить доступность сервера:

```
u-boot> ping 192.168.0.250
host 192.168.0.250 is alive
```

Перед исполнением данной команды порт рестартует. Если устройство подключено к ПК напрямую кросс-кабелем, то порт ПК в этот момент переходит в DOWN, и восстанавливается не сразу, поэтому *ping* не проходит. Кроме того, установленный на ПК сервер TFTP (в частности, TFTP32) может при этом перейти на другой сетевой интерфейс. Рекомендуется соединять устройство с ПК не напрямую, а через коммутатор Ethernet.

8. Загрузить файл Recovery Linux в оперативную память устройства:

```
u-boot> tftp 20200000 nsg700recovery.bin
TFTP from server 192.168.0.250; our IP address is 192.168.0.1
Filename 'nsg700recovery.bin'.
Load address: 0x20200000
Loading: #####
.....
done
Bytes transferred = 1641784 (190d38 hex)
```

Следует обратить внимание на размер файла, выведенный в последней строке. Это значение понадобится ниже на шаге 11.

ВНИМАНИЕ Приведенное выше значение 190d38 является примером и его не следует копировать буквально. Размер файла меняется для каждой новой версии Recovery.

Если в процессе загрузки выводятся предупреждения T (Timeout) и checksum bad, их, в принципе, можно игнорировать, поскольку эти ошибки и потери пакетов нормально компенсируются средствами TFTP. Однако большое число таких ошибок свидетельствует о некорректной работе сети, сетевого адаптера ПК, его операционной системы или перегрузке аппаратной части (процессора, диска, подсистемы ввода-вывода и т.п.)

Операции, выполненные до этого момента, не затрагивают непосредственно содержимого FlashROM и могут быть прерваны в любой момент. Этапы 9–11 составляют критическую часть процедуры.

9. Снять программную защиту с области, в которой хранится U–Boot и его конфигурация:

```
u-boot> pro off 10000000 1003ffff
Un-Protected 2 sectors
```

(На устройствах старых выпусков — 4 сектора, в зависимости от типа микросхемы Flash-памяти).

10. Стереть область FlashROM, необходимую для размещения Recovery Linux:

```
u-boot> erase 10000000 101fffff
..... done
Erased 16 sectors
```

(На устройствах старых выпусков — 32 сектора).

11. Скопировать Recovery Linux из оперативной памяти во FlashROM. Последним параметром команды является размер файла, полученный на шаге 8, в шестнадцатеричном виде:

```
u-boot> cp.b 20200000 10000000 190d38
Copy to Flash... done
u-boot>
```

12. Перезагрузить устройство в сервисном режиме. Для этого необходимо нажать кнопку CS и, не отпуская её, нажать кнопку RESET (либо выключить и включить электропитание устройства). Через 5 сек. отпустить кнопку CS.

13. Изменить адрес ПК на 192.168.1.250. Подключиться к устройству с помощью Web-браузера по адресу 192.168.1.1. Будет выведена специальная страница для установки типа и серийного номера устройства.



ВНИМАНИЕ Данная страница доступна только единственный раз после установки Recovery Linux. Следует проявлять особое внимание при её заполнении. Параметры, зашитые в устройство на этом шаге, не могут быть изменены впоследствии.

В первом поле необходимо выбрать тип и модификацию устройства из предложенного списка, например, NSG700_4AU_v5. От правильного выбора типа шасси может зависеть работоспособность отдельных его портов и модулей расширения. Во втором поле необходимо ввести заводской номер устройства — 10 цифр. В отличие от NSG Linux 1.0, номер вводится полностью, включая первые 4 цифры (год и месяц выпуска). MAC-адрес генерируется автоматически из заводского номера. По завершении необходимо нажать кнопку "Save Settings".

ПРИМЕЧАНИЕ Заводской номер должен содержать ровно 10 цифр. Для очень старых устройств, имеющих 8-значный номер (YYMMNNNN), необходимо вставить в середину номера два нуля: YYMM00NNNN.

14. После сохранения параметров окно принимает вид, приведённый в §1.8.1. Следует выполнить установку основного программного обеспечения средствами Recovery Linux, как описано в указанном параграфе.

§1.8.4. Обновление ПО сервисного режима

В отдельных случаях возможны существенные доработки и изменения основного ПО NSG Linux 2.0, требующие внесения изменений не только в него самого, но и в загрузчик и в ПО сервисного режима. Обновление производится с помощью штатной процедуры установки частичных обновлений (*service packs*), предусмотренной в NSG Linux 2.0. Процедура доступна как через Web-, так и через консольный интерфейс в узле `.system.software.service-pack`.

ВНИМАНИЕ Замена ПО сервисного режима является критически ответственной процедурой и должна производиться только при электропитании устройства от источника бесперебойного питания. В случае отказа электропитания в ходе процедуры устройство остаётся в неработоспособном состоянии. В этом случае восстановление программного обеспечения возможно только в компании NSG, гарантия аннулируется, транспортировка устройства в NSG и обратно производится за счёт заказчика.

Для обновления ПО необходимо выполнить следующие действия:

1. Ознакомиться с предварительными условиями и требованиями для данного обновления, приведёнными в Приложении 1–Б к настоящему документу, а также в файле `_readme_1st_!!!` на сайте. В случае необходимости, обновить основное ПО NSG Linux 2.0 до указанной промежуточной версии, необходимой для работы данного обновления.
2. Загрузить файл с новым ПО с FTP-сайта NSG и разместить его на доступном сервере TFTP, FTP или HTTP. Текущие версии *recovery* для разных устройств находятся по адресу
http://www.nsg.ru/nsg-linux/binary/2.0/recovery_current/
ftp://ftp.nsg.net.ru/pub/nsg-linux/binary/2.0/recovery_current/
3. Проверить доступность этого сервера с устройства NSG при помощи *ping*.
4. Указать URL файла (с префиксом `tftp://`, `ftp://` или `http://`, соответственно).
5. В поле ввода команды `install` набрать `yes` и нажать кнопку . (Для консольного режима — ввести команду `install`, на запрос подтверждения ответить `yes`.)

Дальнейший вывод команды зависит от конкретной модели устройства.

Для устройств серии NSG–1000: После загрузки и проверки файла с новым ПО будет выведено предупреждение о необходимости обеспечения гарантированного электропитания. Необходимо ввести `yes` (в консольном окне Web-интерфейса или в консольной сессии, соответственно), чтобы принять данные условия. После этого выполняется перезапись ПО сервисного режима. Лог процедуры приведён ниже.

	Total	used	free	shared	buffers
Mem:	1807644	27568	1780076	0	136
Swap:	0	0	0		
Total:	1807644	27568	1780076		

Start download file /nsg1000-x86-recovery-updater from tftp://192.168.1.250:69

#####

File saved

This service pack is going to update the NSG Recovery.

WARNING!

Please make sure that the NSG device is powered with uninterruptible power supply (UPS). Otherwise, you are doing this update on your own risk and responsibility.

In case of power failure during this operation, the device may be rendered inoperable and need repair at NSG facilities. Your warranty will be voided and you will pay for transportation to NSG and back to you.

Enter "yes" to agree with these conditions: **yes**

NSG Recovery updater for NSG-1000.

Updater version: 2.0.2_20120621

Current version: 2.0.2_20111004

Start updating ...

... done successfully.

_C_L_O_S_E_D_

Для устройств NSG-700 и NSG-1800: Если процедура выполняется успешно, то выводится следующий лог:

```
Start download file ... from ...
#####
File saved
Update Recovery procedure started!
Erasing 128 Kibyte @ 1a0000 -- 92 % complete.
Writing started. Please wait!....
Writing finished.
Update Recovery procedure finished successfully!
_C_L_O_S_E_D_
```

Если версия основного ПО NSG Linux 2.0 слишком старая для применения данного обновления, то

```
Start download file ... from ...
#####
File saved
Update Recovery procedure started!
Error: NSG-software is too old.
Update the NSG-software to 2.0.3-rc1 or newest.
_C_L_O_S_E_D_
```

После обновления немедленной необходимости в перезагрузке устройства нет, обновление выполняется в процессе работы основного ПО и не мешает ему. Однако для того, чтобы воспользоваться новыми функциями сервисного режима (настроить консоль, установить самую новую версию основного ПО и т.п.), естественно, следует перезагрузить устройство в сервисном режиме.

Приложение 1–А. Настройка UTF–8 в консольных программах

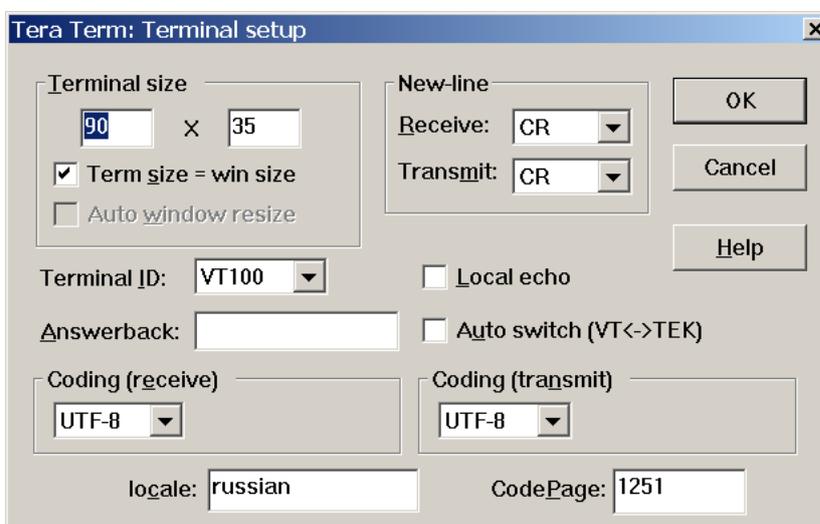
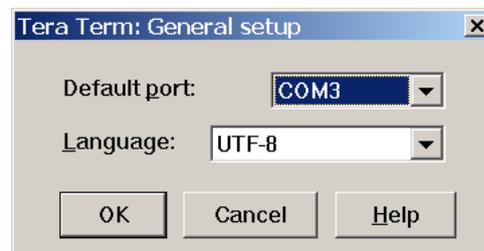
§1–А.1. TeraTerm 4.6x

TeraTerm (<http://ttssh2.sourceforge.jp/>) — удобный клиент Telnet, SSH и эмуляции терминала с графическим интерфейсом пользователя, обладающий богатыми возможностями. Продукт свободно распространяемый, в том числе для коммерческого применения.

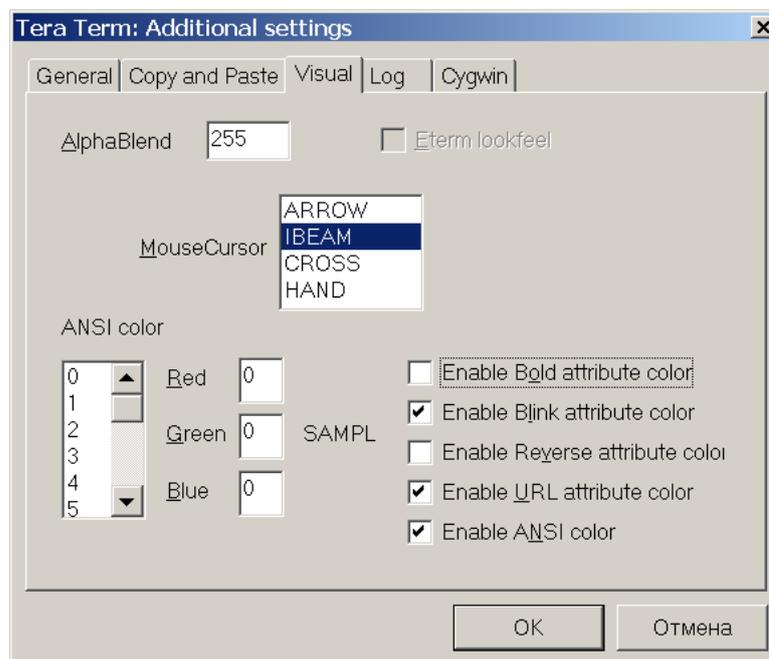
Для настройки UTF–8 необходимо:

- Выбрать в меню Setup — General — Language: UTF–8
- Убедиться, что в меню Setup — Terminal установлено:

Coding (Receive):	UTF–8
Coding (Transmit):	UTF–8
locale:	russian
CodePage	1251 (для ОС Windows) или koi8-r в зависимости от кодовой страницы, используемой системой.



Дополнительно рекомендуется отключить опцию Setup — Additional settings — Visual — Enable Bold attribute color, чтобы корректно отображать атрибуты текста в nsgsh (полужирный как полужирный, а не заменять цветом).



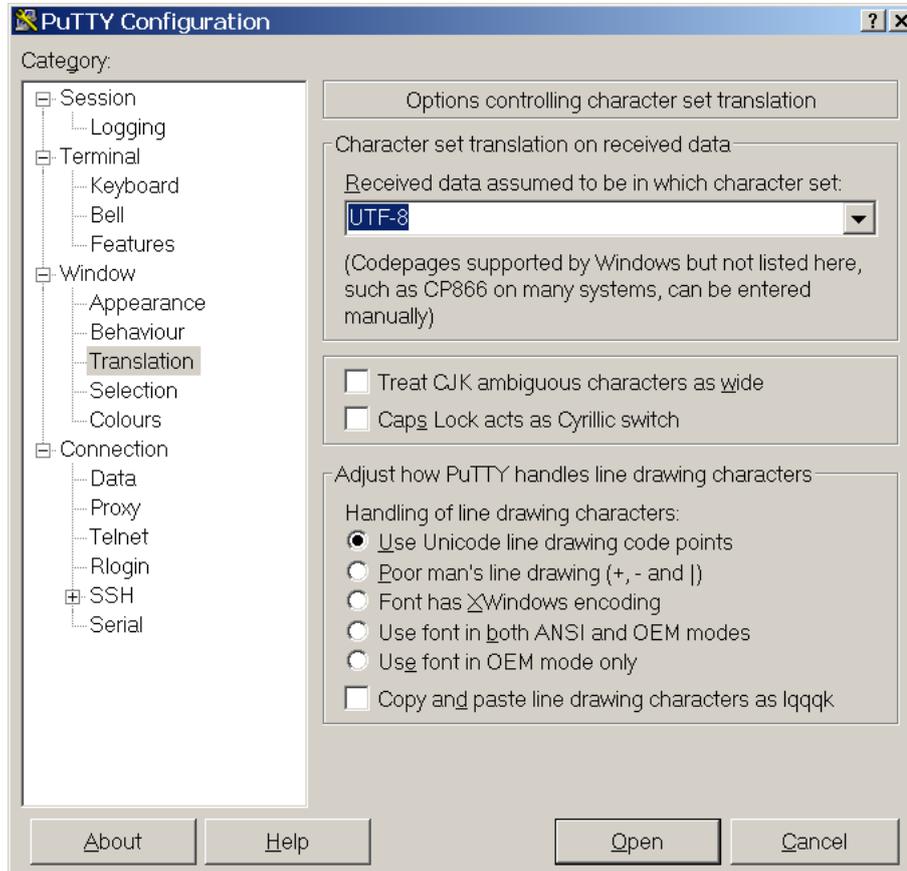
После настройки целесообразно сохранить полученную конфигурацию (Setup — Save setup).

§1–А.2. PuTTY 0.60

PuTTY (<http://www.chiark.greenend.org.uk/~sgtatham/putty/>) — популярный клиент Telnet, SSH и эмуляции терминала с традиционным консольным интерфейсом (за исключением настроек и установления соединения). Продукт свободно распространяемый в исходных кодах.

Для использования UTF-8 необходимо в свойствах сеанса в узле Translation установить:

Received data assumed to be in which character set: UTF-8



После настройки целесообразно сохранить данную сессию.

Приложение 1–Б. Корректировка команд при переходе на новую версию NSG Linux 2.0

Разработчики программного обеспечения уделяют особое внимание обратной совместимости новых версий со старыми. Однако в отдельных случаях такую совместимость не удастся сохранить, поскольку дальнейшее развитие ПО требует изменения синтаксиса некоторых команд или их переноса в другое меню. В этом случае конфигурация существующего устройства может оказаться неработоспособной в новой версии ПО, и ее необходимо откорректировать вручную. Несовместимые изменения команд приведены ниже.

Объекты конфигурации	Было в предыдущей версии	Стало в текущей версии
NSG Linux 2.0.1, 1.0		
<i>Все объекты</i>	<i>Исходный синтаксис</i>	
NSG Linux 2.0.2 beta 1		
NSG–1000		Размер ПО превосходит предел 14МБ, установленный в ранних версиях. Для возможности прошивки ПО из сервисного режима (или срочной необходимости сделать это в будущем) необходимо обновить ПО самого сервисного режима в режиме <i>service-pack</i> (см. §1.8.4). Обновление следует производить при переходе от любой версии 2.0.1.x к любой 2.0.2.x или более поздней. При прошивке из ранних версий, не поддерживающих обновление в режиме <i>service-pack</i> , следует предварительно обновить версию основного ПО до 2.0.1.5.
NSG–700		Изменены внутренние имена асинхронных портов, в т.ч. консольного. Обновление с предыдущих версий на данную и последующие необходимо выполнять только через <i>recovery mode</i> .
NSG Linux 2.0.3		
NSG–1000		Требуется обновление ПО сервисного режима до версии от 26.12.2012 или более поздней.
PPPoE, PPTP	Допускались произвольные имена туннелей.	Имена туннелей должны соответствовать шаблону <i>pppoeN</i> , <i>pptpN</i> , соответственно.
NSG Linux 2.0.4		
NSG–1800		Размер ПО превосходит предел 12МБ, установленный в более ранних версиях. Для установки данной версии необходимо обновить ПО самого сервисного режима в режиме <i>service-pack</i> (см. §1.8.4), а затем установить эту версию из сервисного режима. Обновление возможно для версий основного ПО 2.0.3_rc1 и выше; при прошивке из более ранних версий следует предварительно обновить версию основного ПО до 2.0.3.x.
NSG Linux 2.0.5		
пакетный вызов <i>nsgsh</i>	Если в значении параметра требовалась точка, то она указывалась последовательностью из двух точек, например: <code>nsgsh .port.eth0.ifAddress.prefix.10..0..0..1/24</code>	Точка и другие спецсимволы в имени узла, параметра или значении параметра вводятся единообразно в виде <i>esc</i> -последовательности или в одинарных/двойных кавычках: <code>nsgsh .port.eth0.ifAddress.prefix.10\0\0\0.1/24</code> <code>nsgsh .port.eth0.ifAddress.prefix."10.0.0.1/24"</code> При указании значения параметра через знак = точка в теле значения может вводиться как есть: <code>nsgsh .port.eth0.ifAddress.prefix=10.0.0.1/24</code>
спецсимволы <code>\r</code> , <code>\n</code> , <code>\t</code> , <code>\ </code> и т.п.	При вводе строки в кавычках преобразуются в соответствующие спецсимволы. Без кавычек интерпретируются буквально.	Преобразуются в спецсимволы во всех случаях. Для ввода буквально — необходимо указывать <code>\</code> как <code>\\</code> .

