ООО «С-Терра СиЭсПи»
124498, г. Москва, Зеленоград, Георгиевский проспект,
дом 5, помещение I, комната 33
Телефон/Факс: +7 (499) 940 9061
Эл. почта: information@s-terra.com
Сайт: http://www.s-terra.com

Программный комплекс С-Терра Клиент. Версия 4.1

Руководство администратора

РЛКЕ.00009-02 90 03

12.02.2016

Содержание

1. Комплект поставки

В комплект поставки «Программного комплекса С-Терра Клиент. Версия 4.1» входят:

- CD диск «С-Терра Клиент СР. Версия 4.1. Релиз 14905» либо CD диск «С-Терра Клиент СР. Версия 4.1. Релиз 14101», на котором записаны:
  - ♦ Каталог STerra_Client_CP_KC1_KC2 – дистрибутив С-Терра Клиент СР. Версия 4.1
  - ♦ Каталог STerra_KP – дистрибутив С-Терра КП. Версия 4.1
  - ♦ Каталог Documentation:
    - «Программный комплекс С-Терра Клиент. Версия 4.1. Руководство администратора» – CSP_VPN_Client_Admin_Guide_cp.pdf, Appendix_A.pdf.
    - «Программный комплекс С-Терра Клиент. Версия 4.1. Руководство пользователя» – CSP_VPN_Client_User_Guide_cp.pdf.
    - «Программный продукт С-Терра КП. Версия 4.1. Руководство администратора» – S-Terra_KP_Admin_Guide.pdf.
    - «Программно-аппаратный комплекс С-Терра VPN. Версия 4.1. Правила пользования» – Rules-CP.pdf.
    - «Программно-аппаратный комплекс С-Терра VPN. Версия 4.1. Формуляр» (исполнение «С-Терра Клиент»), (ФСБ России) – Formular_FSB.pdf.
    - «Программный комплекс С-Терра Клиент. Версия 4.1. Формуляр», (ФСТЭК России) – Formular_Client_FSTEK.pdf.
    - ♦ Каталог Certificates:
    - Копия сертификата ФСТЭК.
    - Копия сертификата ФСБ.

В печатном виде поставляются:

- ♦ Голографический специальный защитный знак ФСТЭК России.
- ♦ Лицензия на использование «Программного комплекса С-Терра Клиент. Версия 4.1».
- ♦ Лицензия на использование программного продукта «КриптоПро CSP Driver».

Получить дистрибутив продукта СКЗИ «КриптоПро CSP» можно с сайта компании «Крипто-Про» http://cryptopro.ru/downloads/howto, зарегистрировавшись и введя данные полученной лицензии на этот продукт.

Для защиты «Программного комплекса С-Терра Клиент. Версия 4.1» от несанкционированного доступа (НСД) может использоваться сертифицированное средство доверенной загрузки (ССДЗ): электронный замок «Соболь», «Аккорд-АМДЗ», АПМДЗ «КРИПТОН-ЗАМОК», «Тринити АПМДЗ», «МАКСИМ-М1».

В комплект поставки «Программного комплекса С-Терра Клиент. Версия 4.1» ССДЗ не входит и может быть приобретено дополнительно в нашей компании.

Внимание!

Перед началом работы с Продуктом ознакомьтесь с Правилами пользования.

2. Назначение и функции Продукта

«Программный комплекс С-Терра Клиент. Версия 4.1» функционирует на аппаратных платформах в архитектуре Intel x86/x86-64 под управлением операционных систем Microsoft Windows.

«Программный комплекс С-Терра Клиент. Версия 4.1» (далее Продукт S-Terra Client, Продукт, S-Terra Client, С-Терра Клиент) выполняет роль персонального экрана и VPN клиента.

S-Terra Client может устанавливаться на:

- персональный компьютер пользователя – для защиты индивидуального рабочего места пользователя при работе как в локальных, так и в открытых сетях (Интернет)
- автономный сервер
- специализированные устройства в составе платежных систем: банкоматы, расчетные терминалы, кассовые аппараты (POS-терминалы) и датчики автоматизированных систем управления технологическими процессами.

S-Terra Client предназначен для защиты от несанкционированного доступа, сетевых атак, создания защищенных VPN соединений между устройством, на котором он установлен, и другими взаимодействующими с ним доверенными VPN-шлюзами и VPN-клиентами.

Продукт S-Terra Client выполняет следующие функции:

- защиту трафика на уровне аутентификации/шифрования сетевых пакетов по протоколам IPsec AH и/или IPsec ESP
- пакетную и контекстную фильтрацию любого исходящего и входящего трафика на хост с использованием информации в полях заголовков сетевого, транспортного и прикладного уровней;
- фильтрацию с учетом входного и выходного сетевого интерфейса;
- фильтрацию запросов на установление виртуальных соединений;
- фильтрацию по любым значимым полям IP-заголовка и полям данных сетевого пакета;
- фильтрацию с учетом даты и времени;
- аутентификацию пользователя и аутентификацию узла сети;
- идентификацию и аутентификацию администратора при доступе с целью администрирования;
- событийное протоколирование;
- реализацию заданной дисциплины взаимодействия (аутентификацию и/или защиту трафика) для каждого защищенного соединения, доступ в заданном защищенном режиме только для зарегистрированных, в том числе и для мобильных партнеров по взаимодействию;
- регулируемую стойкость защиты трафика.

S-Terra Client осуществляет защиту трафика протоколов семейства TCP/IP в рамках международных стандартов IKE/IPsec:

- Security Architecture for the Internet Protocol – RFC2401.
- IP Authentication Header (AH) – RFC2402.
- IP Encapsulating Security Payload (ESP) – RFC2406.
- Internet Security Association and Key Management Protocol (ISAKMP) – RFC2408.
- The Internet Key Exchange (IKE) – RFC2409.
- The Internet IP Security Domain of Interpretation for ISAKMP (DOI) – RFC2407.

Продукт S-Terra Client использует в качестве внешней криптографической библиотеки средство криптографической защиты информации (СКЗИ) «КриптоПро CSP», разработанное компанией «Крипто-Про».

СКЗИ «КриптоПро CSP» реализует российские криптографические алгоритмы:

- ГОСТ 28147-89 – шифрование/расшифрование данных,
- ГОСТ Р 34.11-94 – алгоритм хэширования,
- ГОСТ Р 34.10-2001 – формирование и проверка электронно-цифровой подписи (ЭЦП),
- VKO GOST R 34.10-2001 [RFC 4357] – выработка общего сессионного ключа,
- а также генерацию случайных чисел.

Продукт S-Terra Client работает не только с криптоалгоритмами ГОСТ, но и с международными криптоалгоритмами.

Продукт S-Terra Client в режиме КС1 обеспечивают защиту конфиденциальной информации от внешнего нарушителя.

Продукты S-Terra Client в режиме КС2 и сертифицированное средство доверенной загрузки обеспечивают защиту конфиденциальной информации от внутреннего нарушителя.

S-Terra Client является продуктом для корпоративного использования в том смысле, что политику безопасности и настройки режимов этого Продукта осуществляет администратор безопасности предприятия, но он может дать разрешение на дальнейшее управление продуктом конечному пользователю.

Возможно централизованно-удаленное управление настройками S-Terra Client с использованием продукта «С-Терра КП. Версия 4.1». С его помощью можно обновить сертификаты, ключи, политику безопасности, лицензии и др.

В Продукте S-Terra Client по умолчанию для всех интерфейсов задается одинаковая политика безопасности. Для задания разной политики безопасности на интерфейсах используйте структуру NetworkInterface или программный продукт «С-Терра КП. Версия 4.1».

3. Требования на базовые платформы и совместимость

Продукт S-Terra Client 4.1 работает под управлением следующих ОС:

- MS Windows XP SP3 Russian Edition,
- MS Windows Vista SP2 Russian Edition (32-bit, 64-bit),
- MS Windows 7 Russian Edition (32-bit, 64-bit),
- MS Windows 8 Russian Edition (32-bit, 64-bit),
- MS Windows 8.1 Russian Edition (32-bit, 64-bit),
- MS Windows Server 2003 Edition 32-bit,
- MS Windows Server 2008 Edition (32-bit, 64-bit),
- MS Windows Server 2008R2 Edition 64-bit,
- MS Windows Server 2012 Edition 64-bit.

Программный комплекс С-Терра Клиент (исполнения класса защиты КС1) может функционировать в виртуальной среде (VMWare).

Продукт, работающий под управлением ОС Windows XP, Windows Vista, Windows Server 2003, Windows Server 2008, совместим с криптографическими библиотеками, разработанными компанией "Крипто-Про":

- "КриптоПро CSP 3.6" (версия 3.6.5402),
- "КриптоПро CSP 3.6R2" (версия 3.6.6497),
- "КриптоПро CSP 3.6R4",
- "КриптоПро CSP 3.9".

Продукт, работающий под управлением ОС Windows 7, Windows 8, Windows Server 2008R2, совместим с криптографической библиотекой, разработанной компанией "Крипто-Про":

- "КриптоПро CSP 3.6R2" (версия 3.6.6497),
- "КриптоПро CSP 3.6R4",
- "КриптоПро CSP 3.9".

Продукт, работающий под управлением ОС Windows Server 2012, совместим с криптографической библиотекой, разработанной компанией "Крипто-Про":

- "КриптоПро CSP 3.6R4",
- "КриптоПро CSP 3.9".

Продукт, работающий под управлением ОС MS Windows 8.1, совместим с криптографической библиотекой, разработанной компанией "Крипто-Про":

- "КриптоПро CSP 3.9".

Продукт S-Terra Client 4.1 совместим со следующими продуктами компании «С-Терра СиЭсПи»:

CSP VPN Gate – версии 3.1, 3.11,

С-Терра Шлюз – версии 4.1,

CSP VPN Server – версии 3.1, 3.11,

С-Терра КП – версии 3.11, 4.1.

В части реализации протоколов IPsec/IKE и их расширений Продукт совместим с Cisco IOS v.12.4 и v.15.x.x.

Продукт совместим с eToken PRO32k, eToken PRO64k, eToken NG-FLASH, eToken NG-OTP, eToken PRO (Java) производства компании Aladdin, а также eToken 5100 производства SafeNet Incorporation.

4. Процесс подготовки персонального инсталляционного пакета пользователя

Продукт S-Terra Client предназначен для виртуальных корпоративных сетей. Полагаем, что в таких сетях пользователь не имеет права на изменение политики безопасности корпоративной сети. Поэтому, Продукт S-Terra Client разработан таким образом, что администратор безопасности корпоративной сети формирует персонализованный инсталляционный пакет для каждого пользователя, при этом настройки для пользователя согласуются с его должностными обязанностями.

Подготовка персонализованного инсталляционного пакета пользователя производится следующим образом:

Шаг 1: Администратор безопасности на своем компьютере устанавливает СКЗИ «КриптоПро CSP» и административный пакет в виде отдельного Продукта, размещенного в каталоге Client_AdminTool_CP поставляемого диска с дистрибутивом и документацией (см. раздел «Подготовка рабочего места администратора безопасности»).

Шаг 2: Администратор безопасности создает инсталляционный файл S-Terra Client для пользователя, используя

- либо графический интерфейс административного пакета – S-Terra Client AdminTool cp (см. раздел «Графический интерфейс (GUI)»),
- либо утилиту командной строки `make_inst.exe` (см. раздел «Утилита make_inst»).

Шаг 3: Администратор безопасности вычисляет контрольную сумму для инсталляционного файла S-Terra Client при помощи утилиты `integr_mgr`, расположенной в корневой папке административного пакета:

```
integr_mgr calc –f filePath
```

`filePath` имя инсталляционного файла, включая полный путь к нему, для которого будет вычисляться контрольная сумма.

При вычислении контрольной суммы указанного файла, в той же папке будет создан файл с именем `filePath.hash`, содержащий значение контрольной суммы, которая в дальнейшем будет применяться для контроля целостности инсталляционного файла пользователя.

Примечание: если инсталляционный файл и контейнер с секретным ключом передаются пользователю по доверенному каналу связи, то контрольную сумму для инсталляционного файла можно не вычислять.

Шаг 4: Администратор передает пользователю инсталляционный пакет,состоящий из:

- инсталляционного файла S-Terra Client
- контейнера с секретным ключом на внешнем ключевом носителе
- утилиты `integr_mgr`
- файла с контрольной суммой инсталляционного файла S-Terra Client.

Контейнер и файл с контрольной суммой должны быть переданы пользователю по заслуживающему доверия каналу связи.

Шаг 5: Пользователь, получив от администратора персонализованный инсталляционный пакет, проверяет целостность инсталляционного файла (если это необходимо) и производит установку Продукта S-Terra Client на своем компьютере (см. «Руководство пользователя» раздел «Инсталляция S-Terra Client»).

5. Подготовка рабочего места администратора безопасности

Подготовка рабочего места для администратора безопасности осуществляется в несколько этапов:

- инсталляция СКЗИ «КриптоПро CSP», описана в разделе «Установка СКЗИ «КриптоПро CSP»;
- инсталляция административного пакета, описана в разделе «Инсталляция административного пакета».

Перед инсталляцией административного пакета можете убедиться в целостности дистрибутива, размещенного в каталоге Client_AdminTool_CP поставляемого диска. Проверка целостности описана в разделе «Контроль целостности дистрибутива».

5.1 Контроль целостности дистрибутива

Проверка целостности дистрибутива административного пакета осуществляется с использованием утилиты `cpverify`, разработанной компанией «Крипто-Про». Утилита `cpverify` размещена в каталоге установленного продукта КриптоПро CSP. Для вычисления контрольной суммы по каждому файлу дистрибутива, например, `setup.exe`, и выдачи результата на экран выполните команду (указав пути к файлам):

```
cpverify -mk setup.exe
```

Полученное значение сравните с эталонным значением контрольной суммы, записанным в файл `hashes` из состава дистрибутива, который содержит строки вида

```
<hash> <file_name>,
```

где

`<hash>` – эталонное значение контрольной суммы

`<file_name>` – имя файла, для которого подсчитана контрольная сумма.

Для вычисления контрольной суммы для файла дистрибутива и автоматического сравнения с эталонным значением, например, для файла `setup.exe`, выполните команду (указав пути к файлам):

```
cpverify setup.exe hash_from_file,
```

где

`hash_from_file` – эталонное значение контрольной суммы для файла `setup.exe`, скопированное из файла `hashes` (вставить в командную строку можно при помощи нажатия правой кнопки мыши и выбора предложения “Вставить”).

Если проверка прошла успешно, то на экран будет выдано сообщение:
`File <product_file_full_path> has been verified.`

При обнаружении ошибки выдается сообщение:
`File <product_file_full_path> was corrupted,`

где

`product_file_full_path` – полный путь к файлу дистрибутива, на котором произошла ошибка.

5.2 Установка СКЗИ «КриптоПро CSP»

При выполнении процедуры инсталляции СКЗИ «КриптоПро CSP 3.6/3.6R2/3.6R3/3.6R4» выберите:

- вид установки – ***Выборочная***,
- компоненты программы, которые необходимо установить – ***Криптопровайдер уровня ядра ОС***.

Настройка считывателей, носителей и их подключение в СКЗИ описано в документе «Приложение А».

5.3 Инсталляция административного пакета

В состав дистрибутива административного пакета, поставляемого в виде отдельного Продукта S-Terra Client AdminTool (cp) входит:

- `hashes` – файл с эталонными значениями контрольных сумм для каждого файла дистрибутива;
- `setup.exe` – утилита запуска Windows Installer;
- `setup.ini` – настроечный файл, необходимый для setup.exe;
- `sysdlls.cab` – хранилище системных DLL, необходимых для клиента;
- `version.txt` – текстовый файл, содержащий версию Продукта;
- `VPN_CLIENT_ADMIN.msi` – MSI-база инсталлятора (MSI – MicroSoft Installer);
- `VPN_CLIENT_ADMIN.cab` – хранилище файлов клиента.

Администратор должен установить административный пакет на своем компьютере.

Запуск инсталляции производится командой `setup.exe` из административного пакета, появляется окно визарда с приглашением к инсталляции:

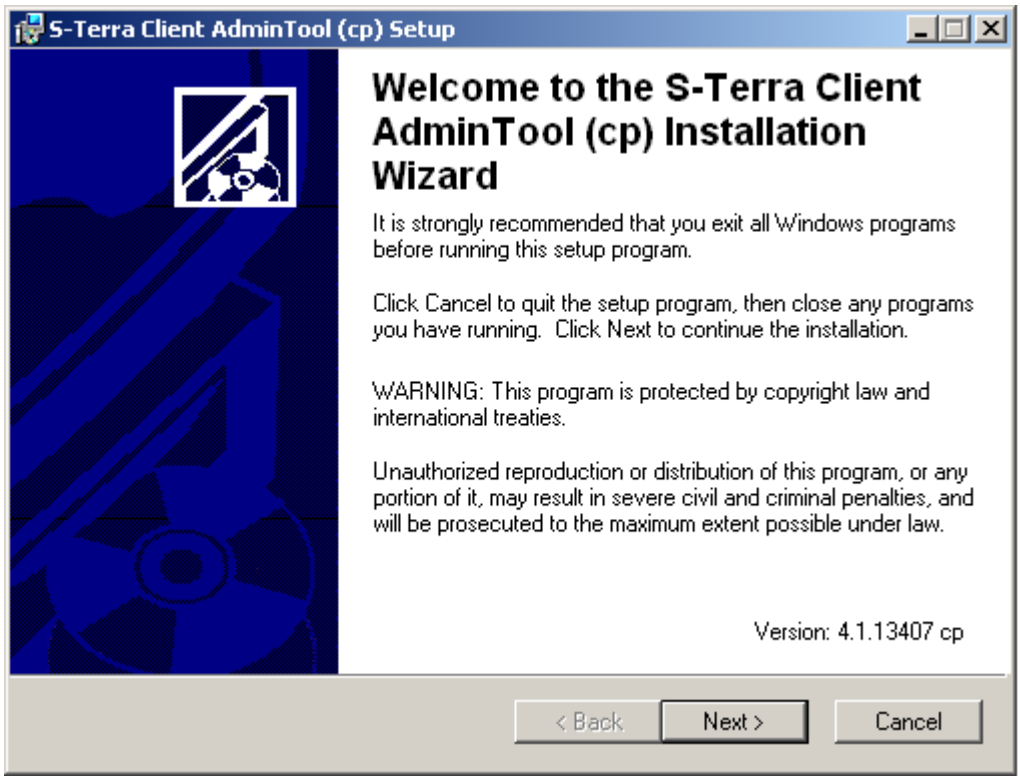

Рисунок 1

В окне с Лицензионным Соглашением установите переключатель в положение "*I accept the license agreement*", кнопка *Next* становится доступной:

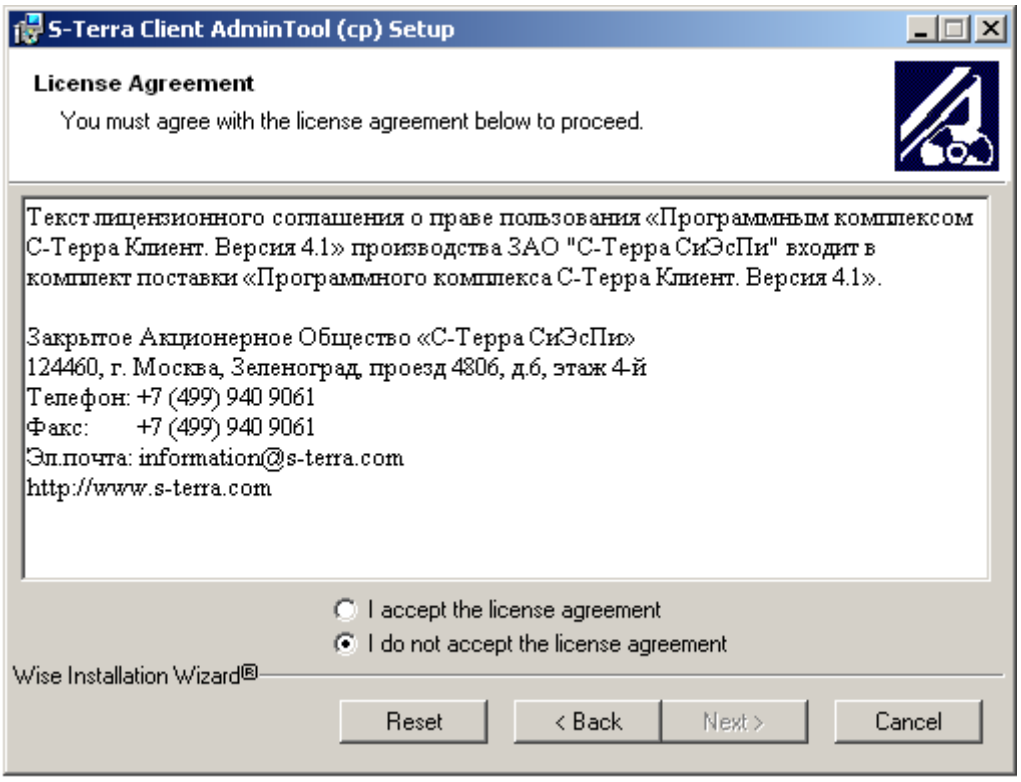

Рисунок 2

Выберите папку, в которую будет установлен административный пакет, нажав на клавишу *Browse*:

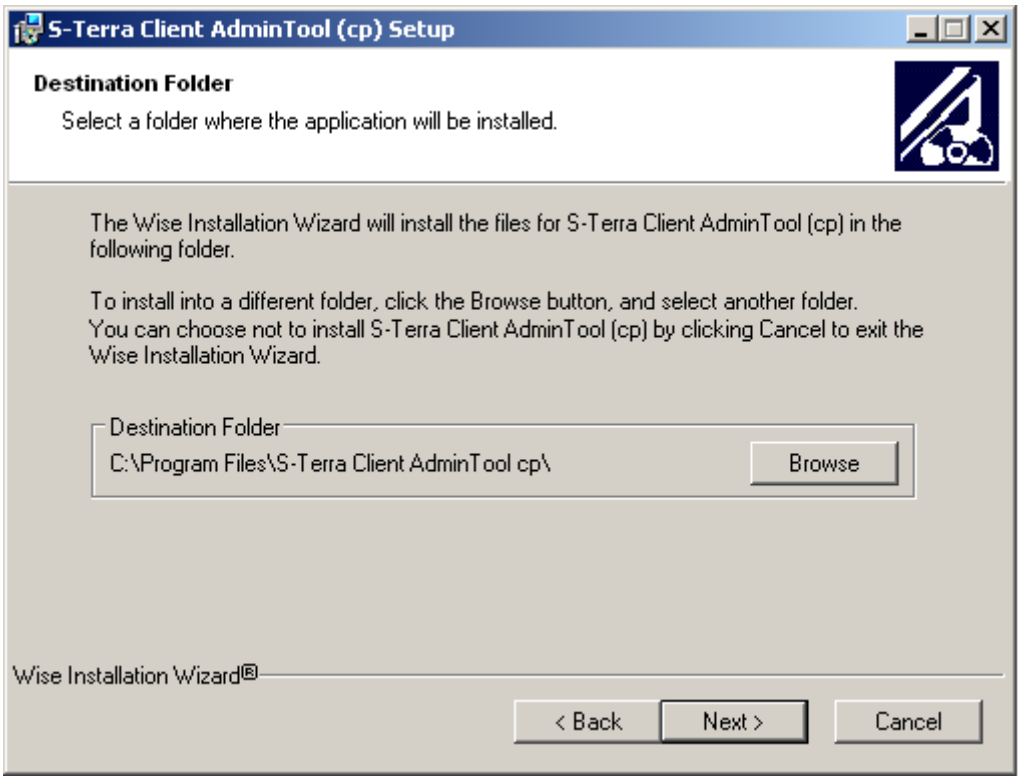

Рисунок 3

Для начала процесса инсталляции нажмите клавишу *Next*:

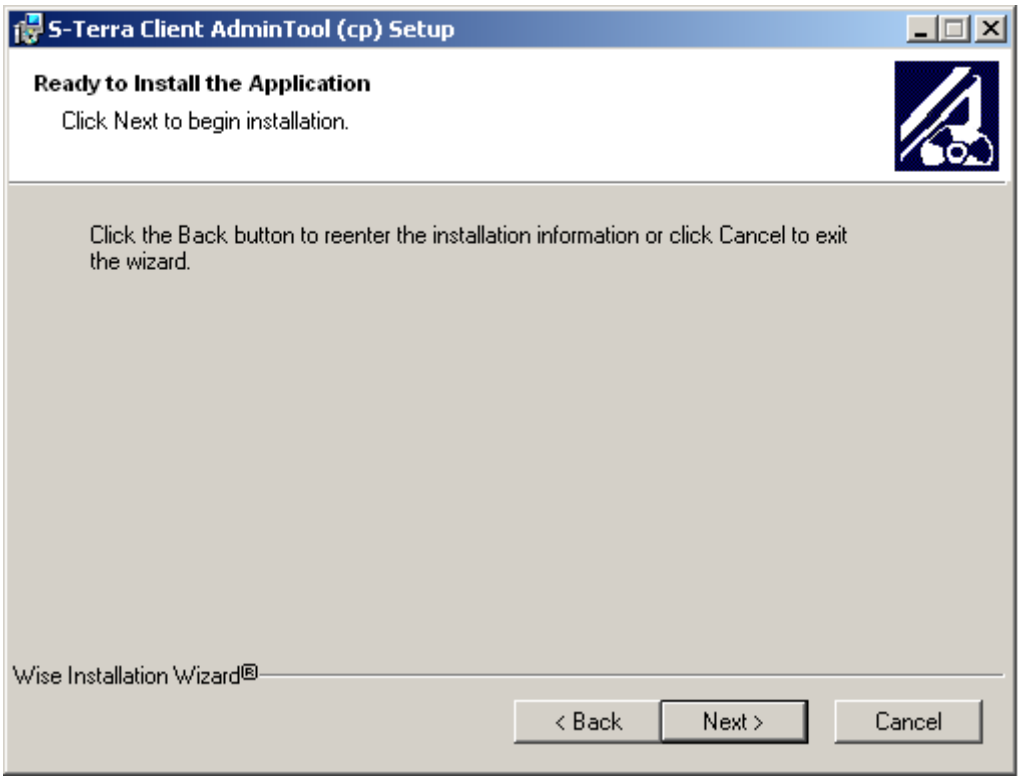

Рисунок 4

Индикатор процесса инсталляции:

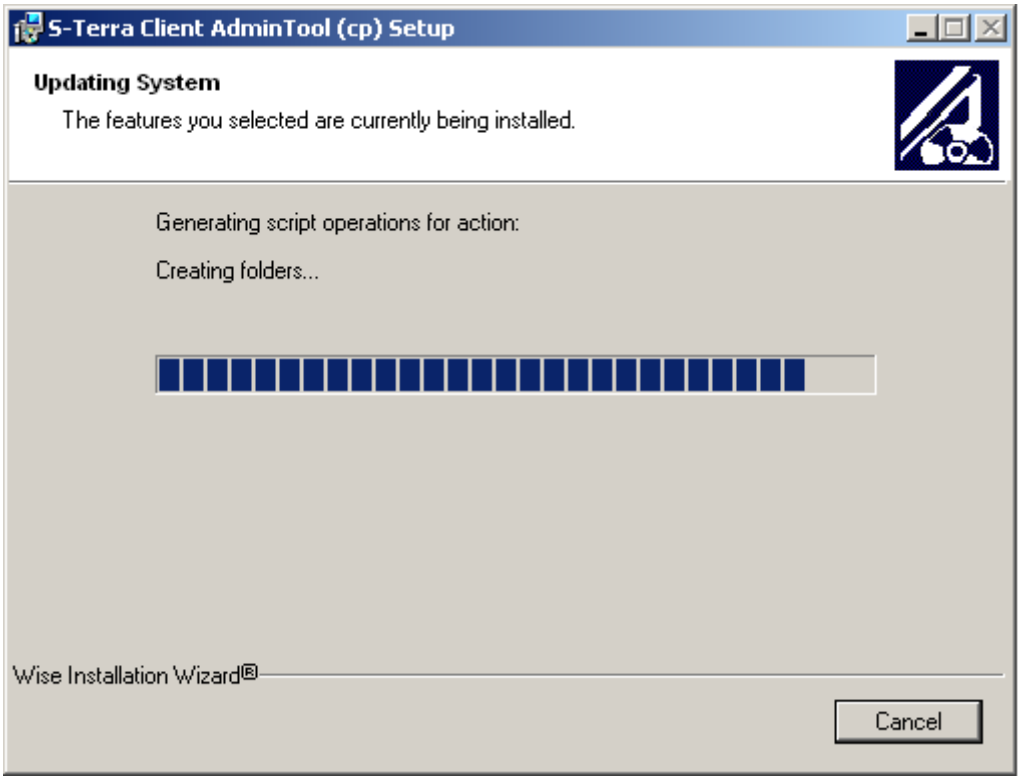

Рисунок 5

Инсталляция завершена, нажмите кнопку *Finish*:

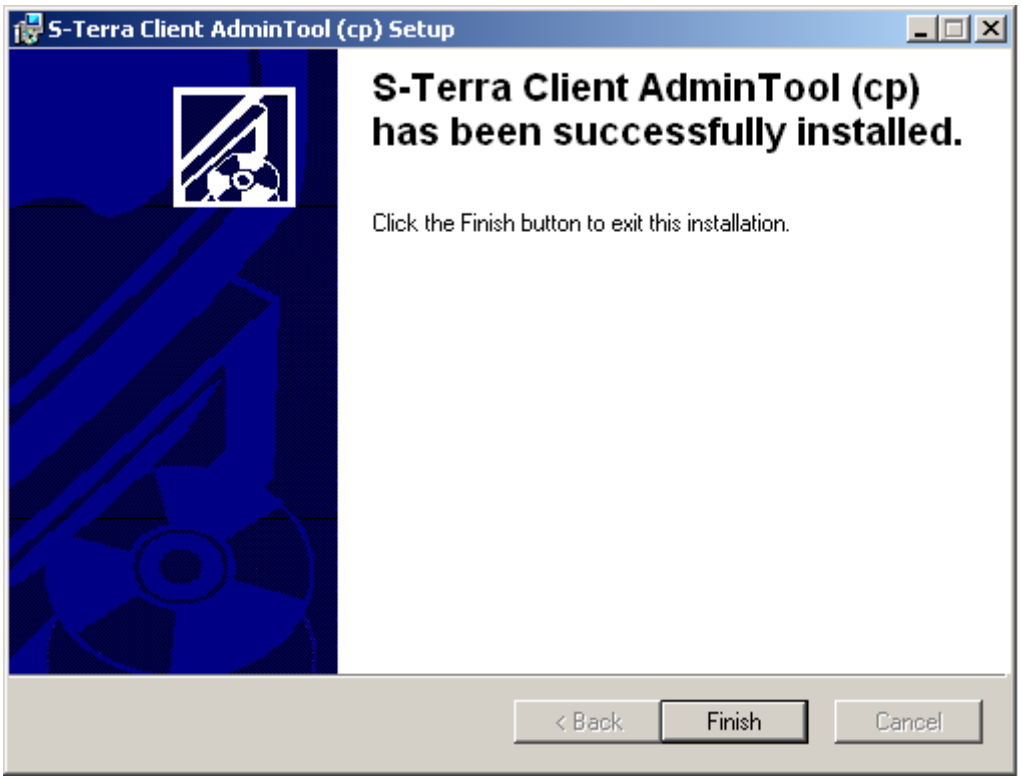

Рисунок 6

При появлении ошибок во время инсталляции административного пакета обращайтесь к разделу «Сообщения об ошибках при инсталляции».

Установленный административный пакет состоит из следующих папок и файлов:

Корневая папка:

- `make_inst.exe` – утилита командной строки для создания инсталляционного файла пользователя (описана в разделе «Утилита make_inst»);
- `pkg_maker.exe` – утилита графического интерфейса для создания локальной политики, локальных настроек и инсталляционного файла пользователя (описана в разделе «Описание GUI»);
- `integr_mgr.exe` – утилита командной строки для проверки целостности информационной части Продукта (описана в разделе «Процесс подготовки персонального инсталляционного пакета пользователя»);
- `version.txt` – текстовый файл с версией Продукта;
- `pkg_maker.chm` – файл, содержащий `Help`;
- вспомогательные файлы (`dll`, `ini`) для обеспечения работы утилит.

Папка `Agent` содержит основные файлы инсталлятора.

Папка `SFX` содержит служебные файлы, необходимые для сборки SFX-архива.

Далее перейдите к разделу «Атрибуты аутентификации».

6. Атрибуты аутентификации

Для аутентификации взаимодействующих сторон протоколу IKE также необходима некоторая аутентификационная информация. Такой аутентификационной информацией может быть:

- предопределенный (разделяемый) ключ (Preshared Key),
- сертификат открытого ключа стандарта X.509.

6.1 Предопределенный ключ

Предопределенный ключ – произвольная последовательность байтов, которая может быть записана в файл. Самый простой способ создать предопределенный ключ – записать в файл любую произвольную последовательность символов.

6.2 Сертификат открытого ключа

При подготовке ***сертификатов*** возможны несколько сценариев, описанных ниже. Более подробное описание приведено в «Приложении А», в разделе «Получение сертификата пользователя».

6.2.1 Режим защиты КС1

Первый сценарий

Все действия по созданию ключевой пары, формированию запроса и созданию сертификата пользователя выполняются администратором СА. При этом контейнер с секретным ключом записывается на внешний ключевой носитель, например, eToken, поддерживаемый СКЗИ «КриптоПро CSP». Администратор безопасности получает Корневой сертификат Удостоверяющего Центра (Trusted CA Certificate), сертификат пользователя, импортированные в файлы, и контейнер на внешнем носителе.

Второй сценарий

Все действия по созданию ключевой пары и формированию запроса на сертификат пользователя производятся на компьютере пользователя (на котором в дальнейшем и будет установлен Продукт S-Terra Client) либо администратором безопасности, либо пользователем. При этом контейнер с секретным ключом размещается на компьютере пользователя в локальном хранилище, например, в Реестре.

Подробно эти действия описаны в «Приложении А» в разделе «Создание ключевой пары и формирование запроса на сертификат пользователя».

6.2.2 Режим защиты КС2

Для режима защиты КС2 ПК от НСД создание ключевой пары и запроса на сертификат пользователя должны выполняться на компьютере с установленным ССДЗ («Соболь» или «Аккорд») (это может быть либо компьютер пользователя, либо администратора). В этом случае при создании ключевой пары будет использоваться не биологический ДСЧ, а аппаратный. Контейнер с секретным ключом должен размещаться на внешнем ключевом носителе, например, eToken.

Добавление аппаратного ДСЧ в «КриптоПро CSP» и создание ключевой пары и запроса на сертификат описано в «Приложении А».

Особенности генерации ключевой пары для режима защиты КС2, если для ССДЗ не поддерживается функциональность ДСЧ описаны в соответствующем разделе в «Приложении А».

6.2.3 Работа с eToken

При инициализации eToken не устанавливайте флажок «При первом входе необходимо изменить пароль» (Рисунок 7).

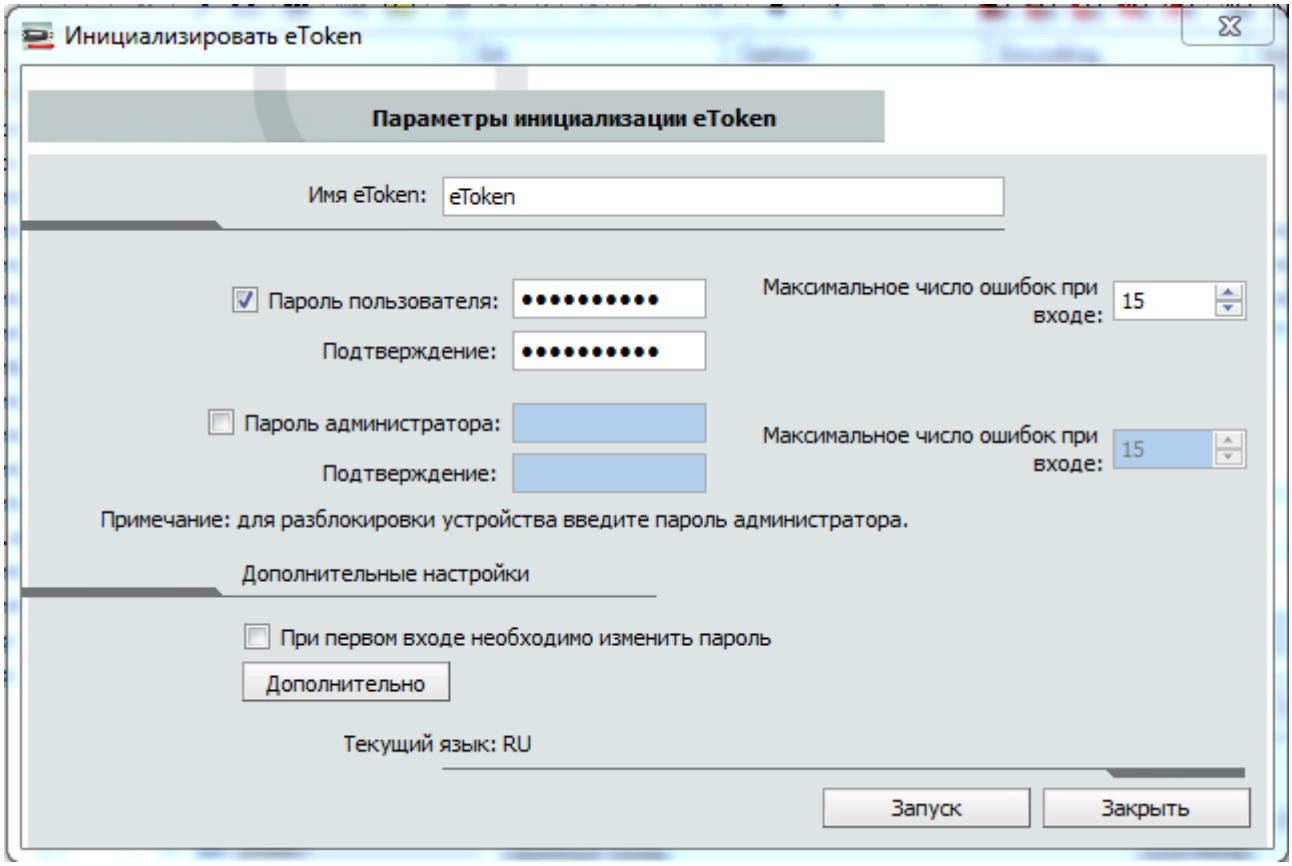

Рисунок 7

6.2.4 Расширения сертификата

Имеются некоторые ***ограничения*** при работе с расширениями сертификата (Extensions), которые помечены как критичные. В Таблица 1 приведен список расширений сертификата, которые будут распознаваться и обрабатываться Продуктом, если у них установлен признак критичности TRUE. Если в сертификате присутствуют другие расширения, имеющие признак криптичности TRUE и не указанные в таблице, такой сертификат не используется. Если же расширение отсутствует в таблице, но является некритичным, то оно игнорируется, а сертификат используется для аутентификации.

Таблица 1

Name	OID value
Subject Key Identifier	2.5.29.14
Key Usage	2.5.29.15
Subject Alternative Name	2.5.29.17

Issuer Alternative Name	2.5.29.18
Basic Constraints	2.5.29.19
Name Constraints	2.5.29.30
CRL Distribution Points	2.5.29.31
Authority Key Identifier	2.5.29.35

Описания значений и полный список Certificate Extensions можно посмотреть в документе RFC 5280 (http://tools.ietf.org/html/rfc5280#section-4.2).

Можно изменить реакцию Продукта на отдельные расширения сертификата, помеченные как критичные и отсутствующие в вышеприведенной таблице. Администратор может настроить список расширений сертификата, который будут игнорироваться Продуктом, как если бы эти расширения являлись некритичными. Эти расширения надо описать в файле `x509opts.ini`, который расположен в корневой папке установленного административного пакета. Расширения описываются в секции `IgnoringUnsupportedCriticalExtentions`.

Игнорируемое `Critical Extention` задается в формате `<KEY>=<OID>`, где:

`<KEY>` – имя расширения, состоящее из букв и цифр и не содержащее разделителей, должно быть уникальным в передах секции;

`<OID>` – OID игнорируемого расширения, состоящий из десятичных чисел, разделенных точками. Распознавание расширения происходит по OID.

Пример файла x509opts.ini:

```
[IgnoringUnsupportedCriticalExtentions]
!!
! Key name is any Alpha-Numerical well-known name of OID
! Key names of different OIDs cannot match
!!
subjectDirectoryAttributes=2.5.29.9
CertificatePolicies=2.5.29.32
QcStatements=1.3.6.1.5.5.7.1.3
HcRole=1.0.21091.2.0.5
```

Примечание 1: следует подчеркнуть, что таким образом нельзя проигнорировать распознаваемые Продуктом `Critical Extention`s, например `BasicConstraints`.

Примечание 2: секция `IgnoringUnsupportedCriticalExtentions`, даже пустая, обязательно должна присутствовать в файле x509opts.ini.

7. Графический интерфейс (GUI)

7.1 Сценарии подготовки инсталляционного пакета с помощью GUI

Утилита `pkg_maker.exe` предоставляет администратору безопасности удобный графический интерфейс для создания локальной политики безопасности для пользователя, локальных настроек Продукта S-Terra Client и создания инсталляционного файла продукта S-Terra Client для пользователя.

При использовании ***предопределенного ключа*** для аутентификации сторон GUI предоставляет возможность считать созданный ключ либо из файла, либо ввести его с клавиатуры, задать локальную политику безопасности для данного пользователя, персональные настройки, и создать инсталляционный файл S-Terra Client, который и будет передан пользователю. Далее перейдите в раздел «Аутентификация с использованием Preshared Key».

При использовании ***сертификатов открытого ключа*** для аутентификации сторон возможны два сценария подготовки инсталляционного пакета, которые отличаются тем, имеет ли администратор на своем рабочем месте доступ к контейнеру с секретным ключом сертификата пользователя. Контейнер с секретным ключом должен быть уровня компьютера.

Первый сценарий

Шаг 1: Администратор безопасности получает от администратора CA Корневой сертификат Удостоверяющего Центра (Trusted CA Certificate) и сертификат пользователя, импортированные в файлы, и также контейнер на внешнем носителе.

Поэтому в данном сценарии возможно на компьютере администратора провести проверку соответствия сертификата пользователя и секретного ключа в контейнере при создании инсталляционного файла.

Шаг 2: Администратор безопасности на своем рабочем месте с помощью GUI задает локальную политику безопасности для данного пользователя, путь к локальному и CA сертификату, имя контейнера с секретным ключом – где он будет размещен на компьютере пользователя, локальные настройки, создает инсталляционный файл S-Terra Client.

Шаг 3: Администратор безопасности передает пользователю подготовленный инсталляционный пакет, состоящий из:

- инсталляционного файла S-Terra Client
- контейнера с секретным ключом на внешнем ключевом носителе
- утилиты `integr_mgr`
- файла с контрольной суммой инсталляционного файла S-Terra Client.

Контейнер и файл с контрольной суммой должны быть переданы пользователю по заслуживающему доверия каналу связи. Инсталляционный файл S-Terra Client содержит базовый инсталляционный файл, локальную политику безопасности, сертификат пользователя и CA сертификат, персональные настройки.

Если администратор подготовил пользовательский токен, то пользователю передается подготовленный инсталляционный пакет, состоящий из:

- инсталляционного файла S-Terra Client
- пользовательского токена с записанным на нем CA сертификатом, локальным сертификатом, контейнером с секретным ключом и локальной политикой безопасности
- утилиты `integr_mgr` для вычисления контрольной суммы
- файла с контрольной суммой инсталляционного файла S-Terra Client.

Пользовательский токен и файл с контрольной суммой должны быть переданы пользователю по заслуживающему доверия каналу связи.

Второй сценарий

Шаг 1: На компьютере пользователя создается ключевая пара и запрос на сертификат пользователя, который отсылается в Удостоверяющий Центр. Контейнер с секретным ключом размещается на компьютере пользователя в локальном хранилище (в Реестре). Администратор безопасности получает Корневой сертификат Удостоверяющего Центра (Trusted CA Certificate) и сертификат пользователя, импортированные в файлы.

В результате администратор безопасности на своем рабочем месте не имеет доступа к контейнеру, поэтому в данном сценарии невозможно на компьютере администратора провести проверку соответствия сертификата пользователя и секретного ключа в контейнере при создании инсталляционного файла.

Шаг 2: Администратор безопасности на своем рабочем месте с помощью GUI задает локальную политику безопасности для данного пользователя, путь к локальному и CA сертификату, имя контейнера на компьютере пользователя, локальные настройки, и создает инсталляционный файл S-Terra Client.

Шаг 3: Администратор безопасности передает пользователю подготовленный инсталляционный пакет, состоящий из:

- инсталляционного файла S-Terra Client
- утилиты `integr_mgr` для вычисления контрольной суммы
- файла с контрольной суммой инсталляционного файла S-Terra Client.

Файл с контрольной суммой должен быть передан пользователю по заслуживающему доверия каналу связи. Инсталляционный файл S-Terra Client содержит базовый инсталляционный файл, локальную политику безопасности, CA сертификат, локальный сертификат, ссылку местоположения контейнера на компьютере пользователя и персональные настройки.

Три режима GUI

Подготовить инсталляционный пакет можно в одном из 3 режимов GUI:

- основной режим (ГОСТ) (описан в разделе "GUI. Основной режим");
- режим пользовательского токена (описан в разделе "GUI. Режим пользовательского токена"). В этом режиме для аутентификации сторон используются только сертификаты;
- режим международных алгоритмов (описан в разделе "GUI. Режим международных алгоритмов").

В ***основном режиме*** при создании инсталляционного пакета S-Terra Client используются алгоритмы ГОСТ при шифровании, проверки целостности пакетов, формировании и проверки ЭЦП.

В ***режиме международных алгоритмов*** при создании инсталляционного пакета S-Terra Client используются международные алгоритмы шифрования, проверки целостности пакетов и ЭЦП.

В ***режиме пользовательского токена*** создается универсальный инсталляционный файл S-Terra Client, который может работать с пользовательским токеном любого пользователя. В этом режиме также создается пользовательский токен для данного пользователя. Продукт S-Terra Client, созданный в этом режиме, будет работать только при наличии подключенного пользовательского токена.

При наличии у администратора сертификатов, выберите режим GUI и перейдите к созданию политики безопасности с использованием GUI.

7.2 Описание GUI

При запуске утилиты **pkg_maker.exe** `(Пуск – Программы – S-Terra Client AdminTool cp - Package Maker)` открывается окно главной формы GUI (Рисунок 8).

Главная форма представляет собой диалоговое окно со вкладками.

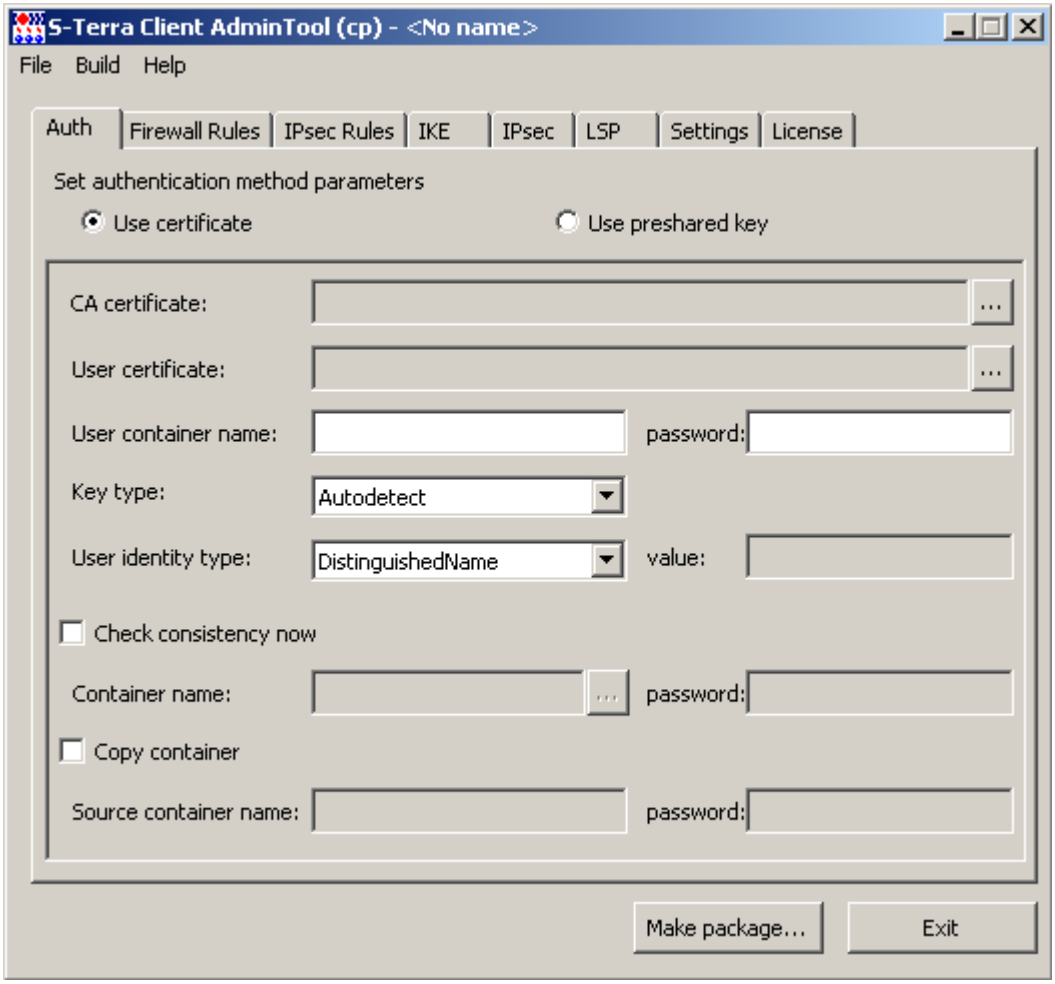

Рисунок 8

Главную форму можно переключать в 3 разных режима работы:

- Основной режим (ГОСТ).
- Режим пользовательского токена (меню `Build –Token pattern`.
- Режим международных алгоритмов `(меню Build – International pattern)`.

Вкладки главной формы предназначены:

- ***Auth*** – для задания способа аутентификации сторон.
- ***Firewall Rules*** – для задания правил маркирования и фильтрации трафика.
- ***IPsec Rules*** – для задания правил защиты трафика.
- ***IKE*** – для задания параметров IKE соединений.
- ***IPsec*** – для задания параметров IPsec соединений.
- ***LSP*** – для просмотра, редактирования локальной политики безопасности (LSP) и задания дополнительных настроек.
- ***Settings*** – для задания локальных настроек и политики по умолчанию.
- ***License*** –для задания параметров Лицензии на Продукт.

Кроме того, главная форма содержит Меню и две функциональные кнопки.

Меню содержит три раздела:

- Раздел **File** имеет следующие предложения:
  - ***New Project*** – открывает новый проект. Проект – это файл в текстовом формате с расширением `dsc`, в котором будет записана LSP с параметрами, заданными во вкладках, и локальными настройками.
  - ***Open Project…*** – открывает существующий (ранее созданный) проект. Для работы с проектом, созданным с помощью предыдущей версии AdminTool, воспользуйтесь пунктом меню *Import Old Project.*
  - ***Import Old Project*** – открывает существующий (ранее созданный) проект, созданный с помошью AdminTool предыдущей версии. Во время чтения проекта, созданного с помошью версии 3.1, зачитываются партнерские сертификаты, сохраненные отдельно от проекта. Если по какой-либо причине не удается прочитать какой-то из них, то выдается предупреждение: «Can't read some of partner certificates: <перечисление сертификатов и ошибок> Do you wish to continue?». При отрицательном ответе загрузка прерывается, при положительном – загрузка продолжается, и в списке партнерских сертификатов отображаются только те, которые удалось прочитать.
  - ***Save Project*** – сохраняет текущее состояние проекта. При попытке сохранения изменений в проекте, созданном в предыдущих версиях AdminTool, будет выдано предупреждение: «Project file will be saved in new format thus backward compatibility with older versions of tool may be broken. Do you wish to continue?» При отрицательном ответе процесс сохранения изменений будет прерван.
  - ***Save Project As…*** – сохраняет текущее состояние проекта в указанный файл.
  - ***Advanced Project Settings*** – вызывает окно для задания дополнительных настроек проекта. В данной версии можно указать сертификаты партнеров, CRL при методе аутентификации сторон с использованием сертификатов.
  - ***Exit*** – выход из GUI.
- Раздел **Build** имеет предложения:
  - ***Make package…*** – запускает процесс создания инсталляционного файла Продукта S-Terra Client (аналогично кнопке *Make package...*).
  - ***Make token...*** – запускает процесс создания пользовательского токена с записью на него CA сертификата, локального сертификата и LSP.
  - ***Token pattern*** – переключает главную форму в режим пользовательского токена.
  - ***International pattern*** – переключает главную форму в режим международных алгоритмов.
- Раздел **Help** имеет три предложения:
  - ***Contents*** – вызывает окно Help-системы с активной вкладкой *Содержание.*
  - ***Index*** – вызывает окно Help-системы с активной вкладкой *Указатель.*
  - ***About…*** – открывает окно с названием Продукта, версии, номера сборки, копирайта и логотипом компании.

Функциональные кнопки:

- ***Make package*** – кнопка для запуска процесса создания инсталляционного файла Продукта S-Terra Client.
- ***Exit*** – выход из GUI.

Формат заполняемых полей

Все поля графического интерфейса, в которые вводится имя папки и файла, могут содержать парные кавычки, пробелы в начале и в конце строки. Все эти символы игнорируются.

Для всех других полей любой введенный символ является значимым.

7.3 GUI. Основной режим (ГОСТ)

7.3.1 Вкладка Authentication

Вкладка **Auth** предназначена для задания метода аутентификации и ввода идентификационных данных пользователя. Поддерживаются два метода аутентификации – с использованием сертификата стандарта X.509 или предопределенного ключа.

7.3.1.1 Аутентификация с использованием сертификатов. Задание корневого и локального сертификатов

При аутентификации сторон с использованием сертификатов поставьте переключатель в положение **Use certificate**:

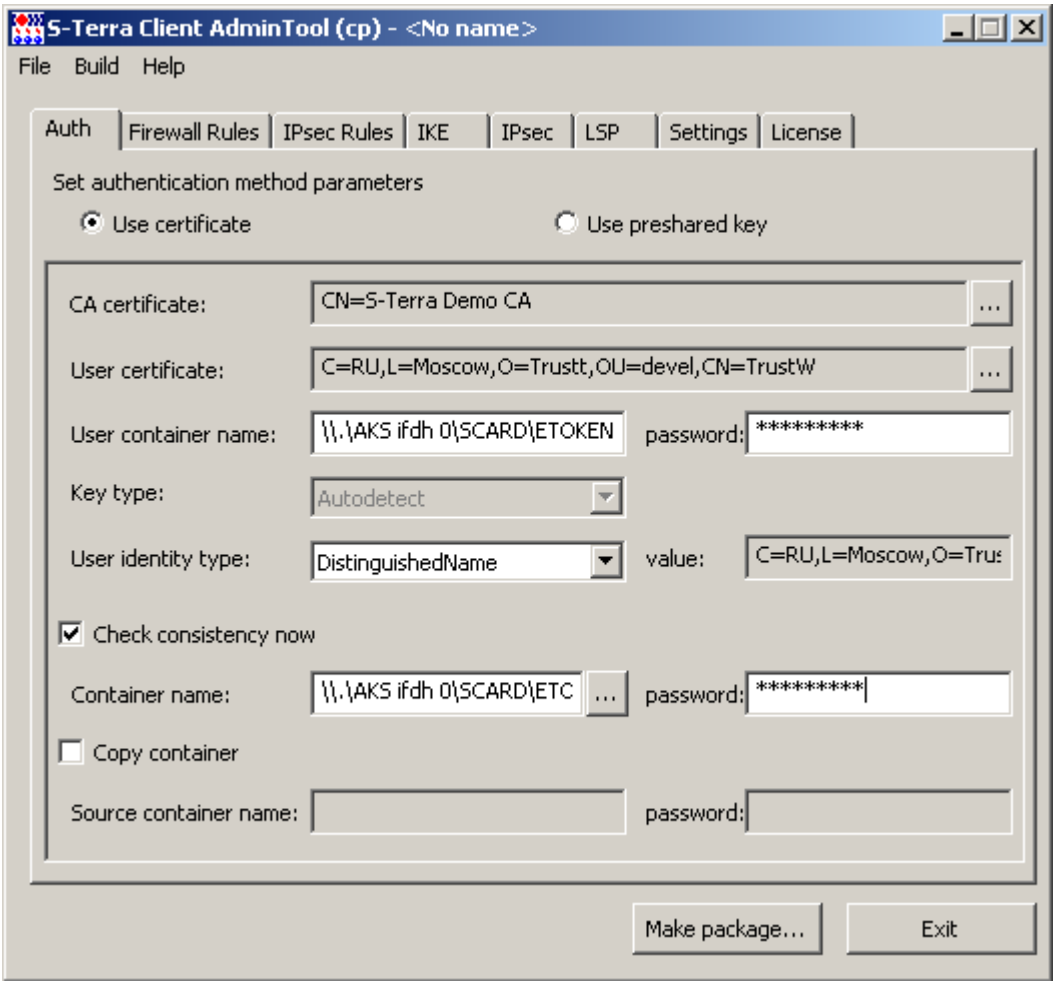

Рисунок 9

При этом становятся доступными для заполнения следующие поля (кнопка с тремя точками в конце поля `[...] (Open)` означает, что элемент этого поля должен быть доступен (размещен) на компьютере администратора):

- **CA certificate** – здесь отражается поле Subject корневого сертификата Удостоверяющего Центра (Trusted CA Certificate). Для этого разместите на компьютере администратора файл с Trusted CA сертификатом и в конце поля нажмите кнопку `[...]`, в открывшемся окне выберите данный файл с CA сертификатом. Обязательный параметр.

- **User certificate** – здесь отражается поле Subject локального сертификата пользователя. Для этого разместите на компьютере администратора файл с сертификатом пользователя и в конце поля нажмите кнопку `[...]`, в открывшемся окне выберите данный файл с сертификатом. Обязательный параметр.
    - **User container name** – уникальное имя контейнера, размещенного на компьютере пользователя, на который будет установлен Продукт S-Terra Client. Контейнер содержит служебную информацию и секретный ключ сертификата пользователя, и не является каталогом файловой системы. Контейнеры с секретными ключами должны быть уровня компьютера. Уникальное имя контейнера включает имя считывателя, имя ключевого носителя и имя контейнера. Должен быть указан тот считыватель и ключевой носитель, на котором будет расположен контейнер на компьютере пользователя. Контейнер должен быть размещен на носителе, который поддерживается СКЗИ «КриптоПро CSP». Обязательный параметр.

      Если контейнер, например, `cont1` находится в Реестре, то уникальное имя контейнера имеет формат:

      `\\.\REGISTRY\REGISTRY\\cont1` или `REGISTRY\\cont1`

      Если контейнер `cont1` находится на дискете, то уникальное имя контейнера имеет, например, формат:

      `\\.\FAT12_A\FAT12\5800BE8B\cont1.000\6264` или `FAT12\5800BE8B\cont1.000\6264`

      Если контейнер находится на eToken Java, то уникальное имя контейнера имеет, например, формат:

      `\\.\AKS ifdh 0\SCARD\ETOKEN_JAVA_00412ff9\CC00\66CF` или `SCARD\ETOKEN_JAVA_00412ff9\CC00\66CF`

      Если в СКЗИ зарегистрировано более одного считывателя для eToken, то в имени контейнера обязательно укажите имя считывателя – `AKS ifdh 0` или `AKS ifdh 1`.

      Чтобы узнать уникальное имя контейнера, размещенного на внешнем ключевом носителе, выполните следующие действия:

      - подключите этот носитель к компьютеру администратора (см. в «Приложении А» разделы «Подключение внешних ключевых считывателей», «Настройка внешнего считывателя и ключевого носителя в «КриптоПро CSP»)
      - установите флажок **Check consistency now** (описан ниже)
      - нажмите кнопку `[...]` в конце поля **Container name**
      - открывшееся окно Container list (Рисунок 10) содержит список доступных контейнеров с уникальными именами, включающее считыватель, ключевой носитель и имя контейнера в hex-цифрах. Выберите нужный контейнер и нажмите кнопку ОК
      - скопируйте уникальное имя контейнера из поля **Container name** в поле **User container name**. После этого флажок **Check consistency now** можно снять.

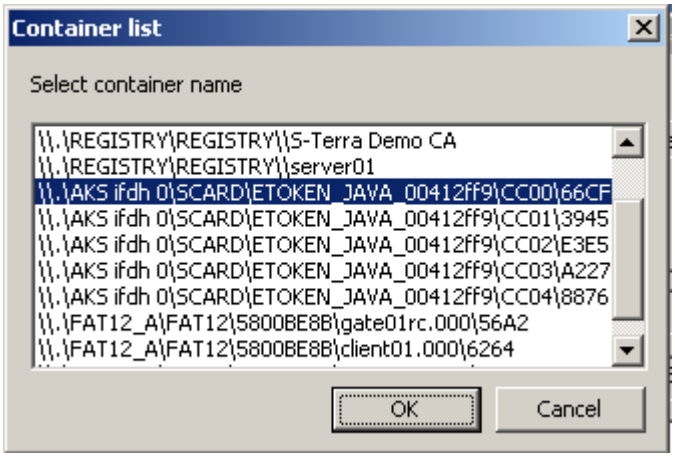

Рисунок 10

- **User container password** – пароль к контейнеру. При использовании eToken в этом поле нужно указать PIN-код к токену.
- **Key type** – тип секретного ключа, хранящегося в контейнере. Этот выпадающий список имеет три значения:
  - *Autodetect* – тип ключа будет определяться автоматически при первом обращении к контейнеру секретного ключа. Определение типа ключа основано на проверке соответствия открытого ключа сертификата пользователя и секретного ключа в контейнере. Значение по умолчанию.
  - *Signature* – ключ для подписи.
  - *Exchange* – ключ для обмена.

    Это поле доступно для выбора, если не будет установлен флажок **Check consistency now**. Если создание ключевой пары и запроса на сертификат пользователя производились средствами MS CA (см. «Приложение А») и был указан тип ключа *both* или *exchange*, то и здесь нужно выбрать *exchange*, а если был указан *signature*, то и здесь нужно выбрать *signature*. Если администратору тип ключа неизвестен, то рекомендуется выбирать значение *Autodetect.*
- **User identity type** – тип идентификационной информации, пересылаемой партнеру при создании защищенного соединения. Обязательный параметр. Поле содержит выпадающий список со следующими значениями:
  - *Distinguished Name* – в качестве идентификатора партнеру будет высылаться значение *Subject* из сертификата пользователя, показываемое в поле **User identity value**, если оно там задано. Значение по умолчанию.
  - *Email* – в качестве идентификатора партнеру будет высылаться значение поля *E-mail* расширения сертификата пользователя, показываемое в поле User identity value, если оно там задано.
  - *FQDN* – в качестве идентификатора партнеру будет высылаться значение доменного имени хоста, считываемое из поля *DNS* расширения сертификата и показываемое в поле **User identity value**, если оно там задано.
  - *IPV4Addr* – в качестве идентификатора партнеру будет высылаться первый IP-адрес, указанный в расширении сертификата, и показываемый в поле **User identity value**, если он там задан.
  - *Local IP address* – в качестве идентификатора партнеру будет высылаться действительный IP-адрес хоста, на котором будет установлен S-Terra Client.
- **User identity value** – идентификационная информация, пересылаемая партнеру. Поле доступно только для чтения и заполняется автоматически, соответствующим типу идентификатора значением, считываемым из сертификата пользователя. Заполнение происходит в момент выбора типа идентификатора или изменения имени файла с сертификатом пользователя. Параметр обязательный.

- **Check consistency now** – установка этого флажка означает, что при создании инсталляционного файла будет проведена проверка соответствия сертификата пользователя и секретного ключа в контейнере. Для этого внешний носитель с контейнером надо подключить к компьютеру администратора. Имя контейнера указывается в поле **Container name**, а пароль к нему – в поле **Container password**.
- **Container name** – уникальное имя контейнера для проведения проверки на компьютере администратора. При нажатии кнопки […] появится окно **Container list** (Рисунок 10) со списком контейнеров на всех ключевых носителях, подключенных к компьютеру администратора. Уникальное имя контейнера включает считыватель, ключевой носитель и имя контейнера в hex-цифрах. Выберите нужный контейнер для проверки и нажмите *OK*. В поле **Container name** появится уникальное имя контейнера.
- **Container password** – пароль к контейнеру с секретным ключом.
- **Copy container** – установка этого флажка означает, что во время инсталляции S-Terra Client на компьютере пользователя будет проведено копирование контейнера с именем, указанным в поле **Source container name**, в контейнер с именем, указанным в поле **User container name**.
- **Source container name** – имя контейнера на компьютере пользователя, который будет скопирован.
- **Source container password** – пароль к контейнеру с секретным ключом.

7.3.1.2 Аутентификация с использованием Preshared Key

При аутентификации сторон с использованием предопределенного ключа поставьте переключатель в положение **Use preshared key**:

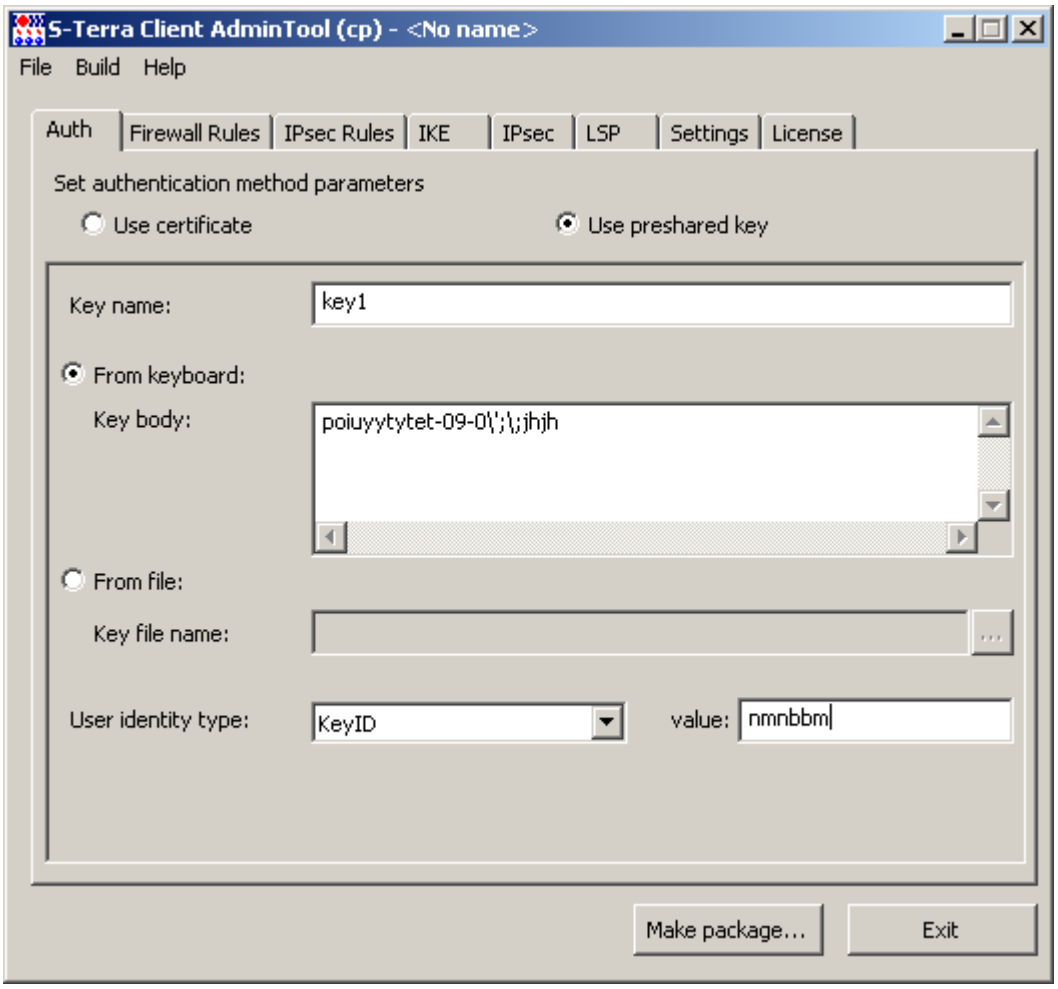

Рисунок 11

Следующие поля доступны для заполнения:

- **Key name** –имя предопределенного ключа. Может состоять из латинских букв, цифр, символов "_" и "-", и должен начинаться с латинской буквы или символа "_". Обязательный параметр.

Для ввода предопределенного ключа имеется переключатель с двумя положениями:

- **From keyboard** – предопределенный ключ нужно ввести с клавиатуры.
  Примечание: Если предопределенный ключ задан несколькими строками, то каждый перенос в теле ключа будет представлен двумя символами 0x0D 0x0A (символ возврата и перевода каретки) и тогда при подготовке предопределенного ключа для партнера должны быть использованы эти символы.
- **From file** – предопределенный ключ считывается из файла с именем, указанным в поле **Key file name**.
- **User identity type** – тип идентификационной информации, пересылаемой партнеру при создании защищенного соединения. Обязательный параметр. Поле содержит выпадающий список со следующими значениями:
  - *IPV4Addr* – в качестве идентификатора партнеру будет высылаться IP-адрес, который нужно задать в поле **User identity value**.

- *KeyID* – в качестве идентификатора партнеру будут высылаться данные из поля **User identity value**. В это поле нужно ввести любую последовательность символов, которая может включать в себя пробелы и русские буквы. Во вкладке **LSP** атрибуту *KeyID* будет присвоено шестнадцатеричное представление заданной последовательности символов, которое и будет высылаться партнеру в качестве идентификатора.
- *Local IP address* – в качестве идентификатора партнеру будет высылаться действительный IP-адрес компьютера, на котором будет установлен S-Terra Client. Значение по умолчанию.

- **User identity value** – значение идентификатора, пересылаемое партнеру. Допустимые значения определяются значением поля **User identity type**. Вводится вручную. Параметр обязательный.

7.3.2 Порядок обработки пакетов

В следующих вкладках **Firewall Rules** и **IPsec Rules** задаются правила обработки пакетов исходящего и входящего трафиков на интерфейс хоста, на котором установлен S-Terra Client.

Для исходящего трафика порядок обработки следующий – маркирование, защита трафика, пакетная и контекстная фильтрация. Для входящего трафика – пакетная и контекстная фильтрация, декапсуляция, маркирование трафика.

Далее описаны разделы, в которых следует задавать правила для перечисленных этапов обработки пакетов.

7.3.2.1 Исходящий трафик

1. Сначала для исходящего трафика ищется подходящее правило из набора правил *Outbound classification*, заданного во вкладке **Firewall Rules** (Рисунок 12). В нем задаются правила классификации и маркирования трафика, но могут быть заданы и правила пакетной фильтрации перед инкапсуляцией в IPsec. Если для исходящего пакета не нашлось подходящего правила из набора правил – пакет уничтожается, и дальнейший поиск правил прекращается. Если же подходящее правило найдено и применено, то поиск правил в этом наборе прекращается, и пакет передается на дальнейшую обработку.
2. Далее начинается поиск подходящего правила из набора правил, заданного во вкладке **IPsec Rules**. Это правило IPsec-инкапсуляции трафика. Если подходящее правило из **IPsec Rules** найдено – пакет обрабатывается и передается дальше. В противном случае – пакет уничтожается.
3. И наконец, для исходящего пакета осуществляется поиск подходящего правила из набора правил *Outbound Filter*, заданного во вкладке **Firewall Rules**. В нем задаются правила пакетной и контекстной фильтрации, но могут быть заданы и правила классификации и маркирования трафика. Если правило найдено – пакет обрабатывается, в противном случае – уничтожается.

7.3.2.2 Входящий трафик

1. Сначала для входящего трафика ищется подходящее правило из набора правил *Inbound Filter*, заданного во вкладке **Firewall Rules**. В нем задаются правила пакетной и контекстной фильтрации, но могут быть заданы правила классификации и маркирования трафика.
2. Далее начинается поиск подходящего правила из набора правил, заданного во вкладке **IPsec Rules**, – декапсуляция IPsec-трафика.
3. И наконец, осуществляется поиск подходящего правила из набора правил раздела *Inbound classification*, заданного во вкладке **Firewall Rules**. В нем задаются правила классификации и маркирования трафика, но могут быть заданы и правила пакетной фильтрации.

Если для входящего пакета найдено и применено подходящее правило из первого набора правил, то дальнейший поиск правил в этом наборе прекращается, и начинается поиск подходящего правила из следующего набора.

Если для входящего пакета не нашлось подходящего правила из набора правил – пакет уничтожается, и дальнейший поиск правил прекращается.

7.3.3 Вкладка Firewall Rules

Во вкладке **Firewall Rules** (Рисунок 12) можно создавать, редактировать, удалять правила пакетной и контекстной фильтрации трафика, а также классификации и маркирования трафика.

Для исходящего и входящего трафиков задаются разные правила.

Правила классификации и маркирования исходящего трафика задаются в наборе правил *Outbound classification*, входящего трафика – в *Inbound classification*.

Правила пакетной и контекстной фильтрации для исходящего трафика – в наборе правил *Outbound filter*, для входящего трафика – в *Inbound filter*.

В скобках указывается количество правил в каждом наборе.

Правила в списке *Rule list* каждого набора должны быть расположены в порядке убывания приоритета. В списке должно находиться хотя бы одно правило.

При получении TCP/IP пакета правила будут просматриваться в порядке убывания приоритета и сравниваться параметры заголовка пакета с такими же параметрами в правиле до нахождения первого подходящего правила. Если для пакета разрешительное правило не найдено – пакет уничтожается.

Refuse inbound TCP connections – установка этого флажка запрещает создание TCP соединений, инициированных извне, в том числе и из защищенной сети. При формировании LSP добавляются соответствующие правила.

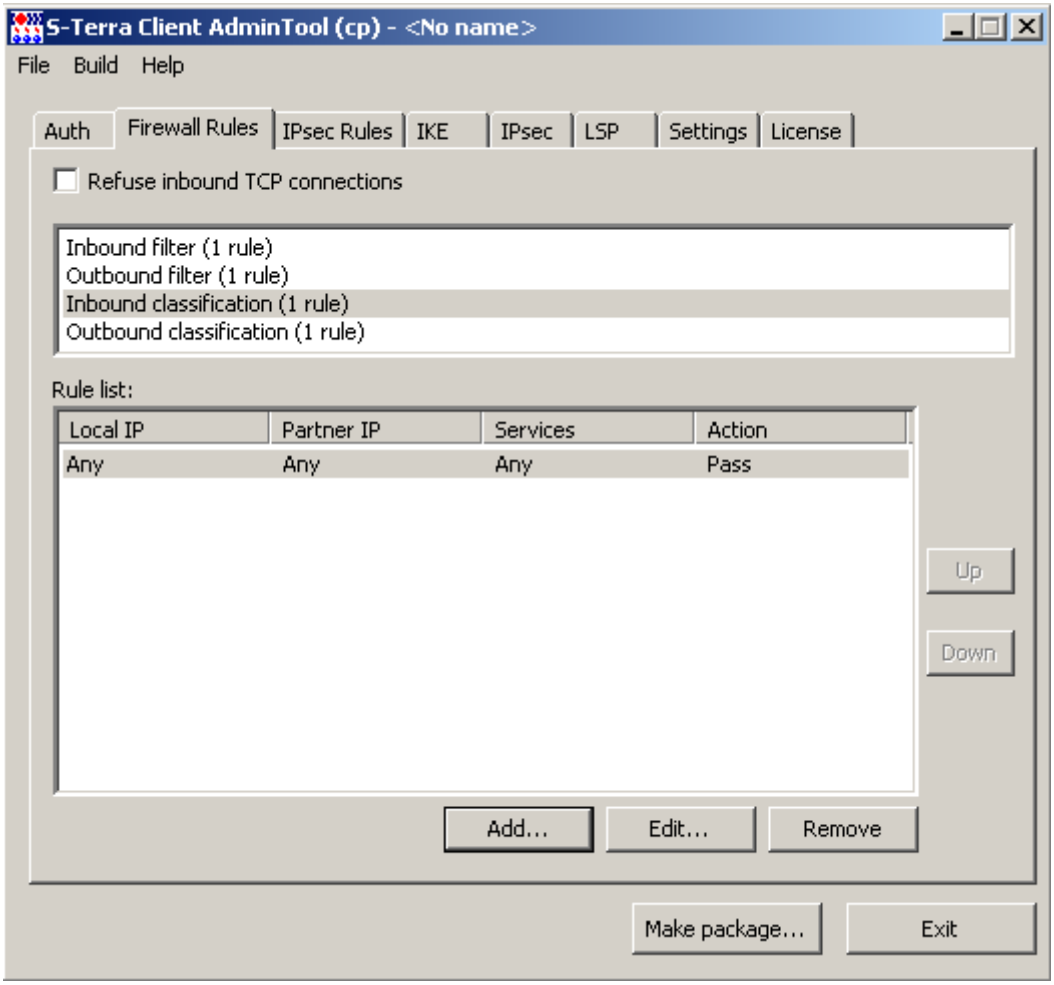

Рисунок 12

Кнопки управления:

- **Add** – вызывает окно для создания нового правила.
- **Edit** – вызывает окно для редактирования выделенного правила.
- **Remove** – удаляет выделенное правило с требованием подтверждения операции удаления. Если в списке только одно правило – оно не удаляется.
- **Up** – при нажатии этой кнопки выделенное правило в списке перемещается на одну строчку вверх, увеличивая свой приоритет.
- **Down** – при нажатии этой кнопки выделенное правило в списке перемещается на одну строчку вниз, уменьшая свой приоритет.

7.3.3.1 Маркирование пакетов. Задание правил в Outbound classification и Inbound classification

При выделении одного из наборов правил *Outbound classification* или *Inbound classification* и нажатии кнопки **Add** (Рисунок 12) появляется окно **Add Rule** (Рисунок 13) для создания правила.

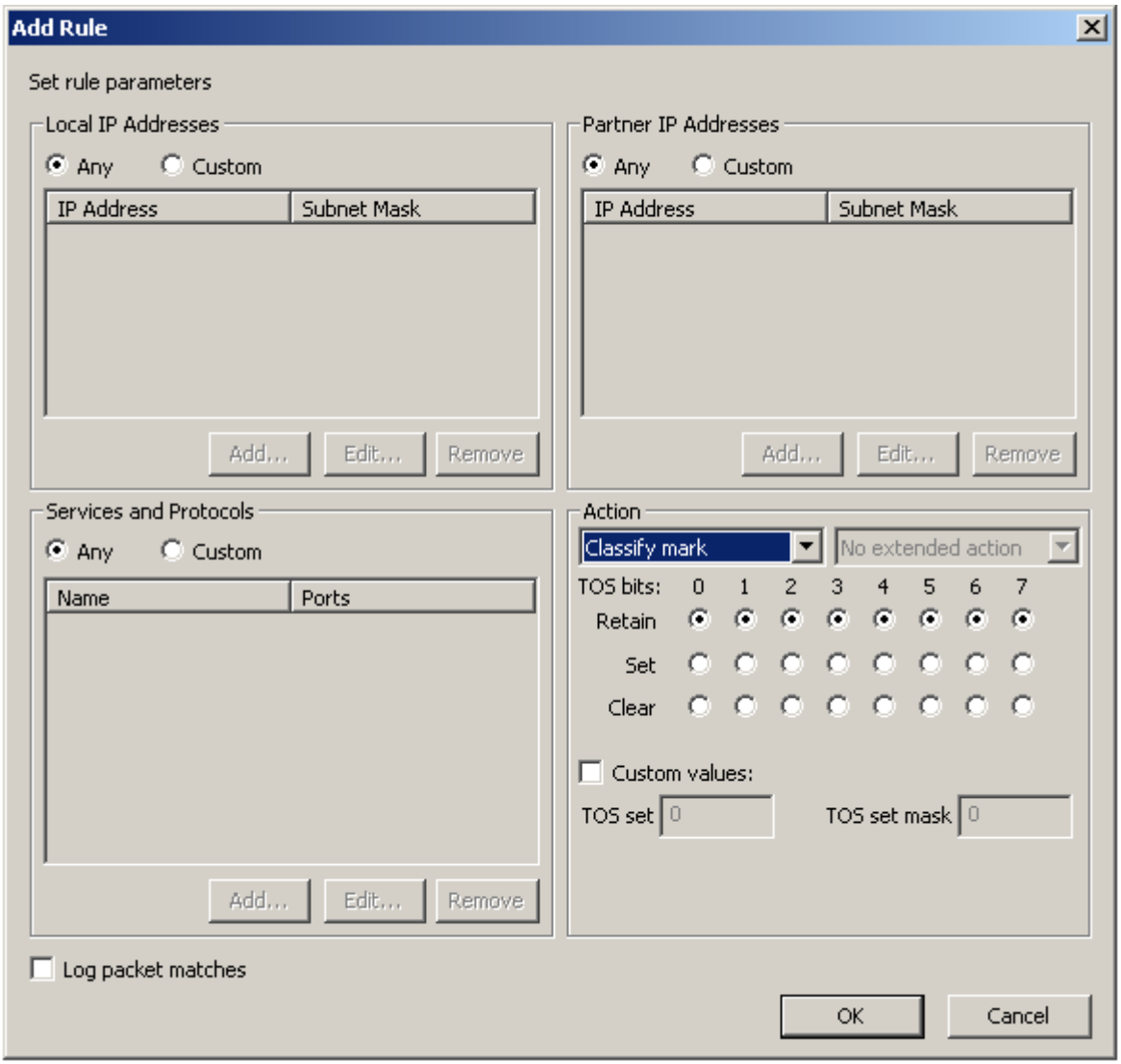

Рисунок 13

Диалоговое окно **Add Rule** имеет 4 области для задания правила:

- **Local IP Addresses** – в этой области задаются IP-адреса локального VPN устройства или подсети, на которые будет распространяться правило. Область имеет переключатель с двумя положениями:
  - *Any* – используется любой IP-адрес.
  - *Custom* – становится доступным окно для ввода IP-адреса и маски подсети.
- **Partner IP Addresses** – в этой области задаются IP-адреса или подсети партнеров, на которые распространяется правило.
- **Services and Protocols** – область для задания сетевых сервисов и протоколов, на которые распространяется правило.
- **Action** – в этой области задается действие, которое будет применено к пакету, если он попадает под данное правило.

Log packet matches – установка флажка задает протоколирование событий обработки пакетов, подпадающих под данное правило.

Кнопки управления:

- **Add** – вызывает окно **Add IP Address** (Рисунок 14) для ввода IP-адреса и маски хоста или подсети.
- **Edit** – вызывает окно для редактирования выделенной записи.
- **Remove** – удаляет выделенную запись с требованием подтверждения операции удаления.

Задание IP-адреса и маски подсети в правиле

В областях **Local IP Addresses** и **Partner IP Addresses** для задания/редактирования IP-адреса хоста (подсети) и маски подсети в правиле следует установить переключатель в положение *Custom* и кнопкой **Add** или **Edit** вызвать окно **Add/Edit IP Address** (Рисунок 14). Если задается IP-адрес хоста, то сетевая маска равна 255.255.255.255. Адрес не может быть нулевым.

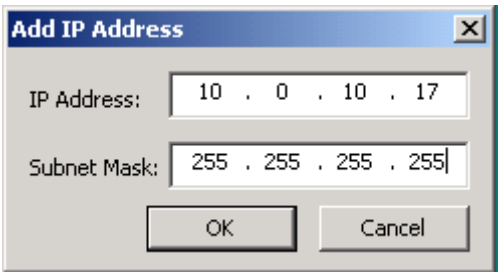

Рисунок 14

Задание сетевого сервиса или протокола в правиле

В области **Services and Protocols** установить переключатель в положение *Custom* и кнопкой **Add** вызвать окно **Add Service** (Рисунок 15):

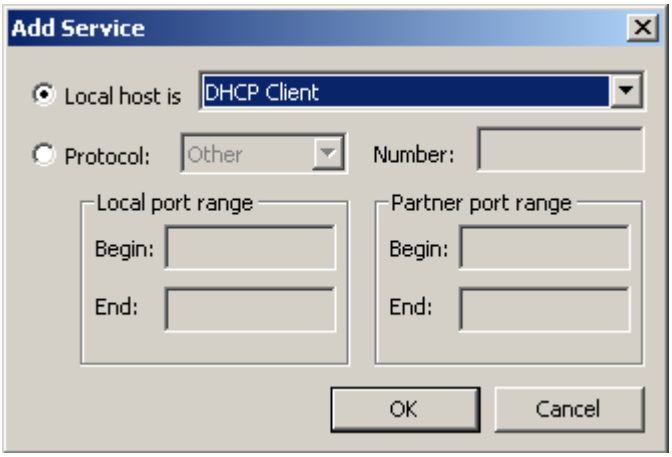

Рисунок 15

Здесь имеется переключатель с двумя положениями:

- **Local host is** – при установке переключателя в это положение выбрать из выпадающего предопределенного списка сервис и в каком качестве выступает локальное устройство – клиент или сервер.

  Список предлагаемых сетевых сервисов и протоколов выбран в соответствии с перечнем IANA (http://www.iana.org/assignments/port-numbers):

  - *DHCP Client* – все пакеты протокола TCP и UDP, идущие на порт 67 компьютера партнера и все пакеты протокола UDP, идущие на порт 68 локального компьютера.
  - *DHCP Server* – все пакеты протокола TCP и UDP, идущие на порт 68 компьютера партнера и все пакеты протокола UDP, идущие на порт 67 локального компьютера.
  - *HTTP Client* – все пакеты протокола TCP, UDP и SCTP, идущие на(с) порт(порта) 80 компьютера партнера.
  - *HTTP Server* – все пакеты протокола TCP, UDP и SCTP, идущие на(с) порт(порта) 80 локального компьютера.
  - *LDAP Client* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 389 компьютера партнера.
  - *LDAP Server* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 389 локального компьютера.
  - *LDAPS Client* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 636 компьютера партнера.
  - *LDAPS Server* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 636 локального компьютера.
  - *RTELNET Client* – все пакеты протокола TCP, и UDP идущие на(с) порт(порта) 107 компьютера партнера.
  - *RTELNET Server* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 107 локального компьютера.
  - *SMTP Client* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 25 компьютера партнера.
  - *SMTP Server* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 25 локального компьютера.
  - *SNMP* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 161 локального компьютера.
  - *SNMP Trap* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 162 компьютера партнера.

- *SSH Client* – все пакеты протоколов TCP, UDP и SCTP, идущие на(с) порт(порта) 22 компьютера партнера.
- *SSH Server* – все пакеты протоколов TCP, UDP и SCTP, идущие на(с) порт(порта) 22 локального компьютера.
- *TELNET Client* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 23 компьютера партнера.
- *TELNET Server* – все пакеты протокола TCP и UDP, идущие на(с) порт(порта) 23 локального компьютера.

- **Protocol** – при установке переключателя в это положение выбирается протокол из следующего списка: *EGP, GGP, HMP, ICMP, PUP, RDP, RVD, TCP, UDP, SCTP, XNS-IDP* и одновременных перечислений нескольких протоколов: «TCP, UDP» и «TCP, UDP, SCTP» (Рисунок 16). В поле **Number** будет автоматически выводиться номер выбранного протокола. Задать протокол можно и по номеру из зарезервированного пространства (0-255). При указании протокола возможно указание диапазона портов (в тех протоколах, в которых это возможно). Область *Local port range* предназначена для задания портов на локальном устройстве, а область *Partner port range* – для задания портов на компьютере партнера. В полях *Begin* и *End* задается порт или диапазон портов из зарезервированного пространства (0-65535). Значение в поле *Begin* должно быть меньше или равно значению в поле *End*.

Редактирование выделенного сервиса или протокола производится в окне **Edit Service**, совпадающем с окном **Add Service**.

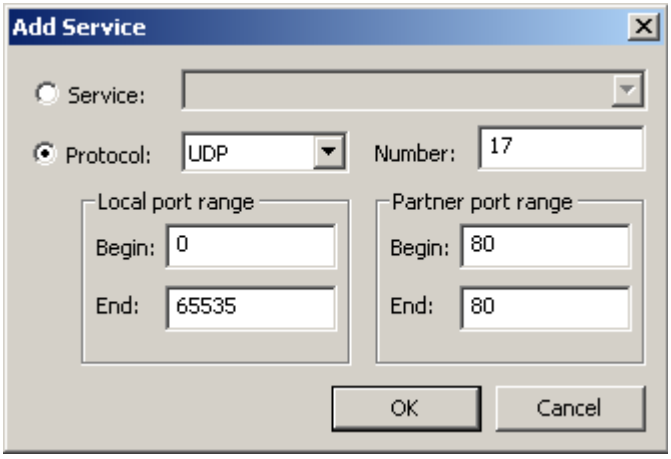

Рисунок 16

Задание действия в правиле

В области **Action** задается действие, которое будет применено к пакету, если пакет попадает под данное правило (Рисунок 24). Из выпадающего списка выбирается одно из действий:

- *Pass* – пропускать пакет.
- *Drop* – не пропускать пакет.
- *Classify mark* – маркировать пакет.

В соседнем поле можно указать дополнительное действие – проверку TCP-флагов.

Действия *Pass* и *Drop* без проверки TCP-флагов используются для создания правил пакетной фильтрации.

Действия *Pass* и *Drop* с проверкой TCP-флагов могут быть использованы для разрешения (запрещения) инициировать TCP-соединение с локального адреса или извне системы (зависит от правила – для исходящего или входящего трафика).

Действие **Pass**:

- *No extended action* – пропускать пакет без проверок (Рисунок 17)
- *TCP flags* – провести проверку пакета на TCP-флаги (Рисунок 18):
  - *Connection initiate* – пропускать первый пакет для инициации TCP-соединения (разрешается инициировать TCP-соединение).
  - *Connection established* – пропускать пакет, принадлежащий установленному TCP-соединению.

Действие **Drop**:

- *No extended action* – не пропускать пакет.
- *TCP flags* – провести проверку пакета на TCP-флаги:
  - *Connection initiate* – не пропускать первый пакет для инициации TCP-соединения (запрещается инициировать TCP-соединение).
  - *Connection established* – не пропускать пакет, принадлежащий установленному TCP-соединению.

Действия *Pass* и *Drop* с проверкой TCP-флагов могут быть использованы для разрешения (запрещения) инициировать TCP-соединение с локального адреса или извне системы.

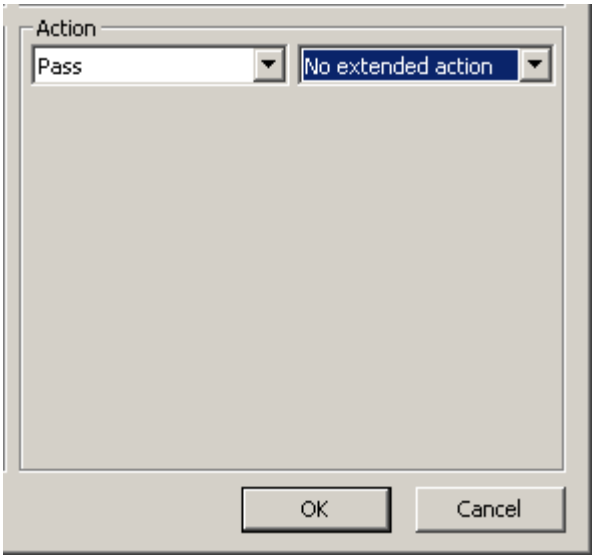

Рисунок 17

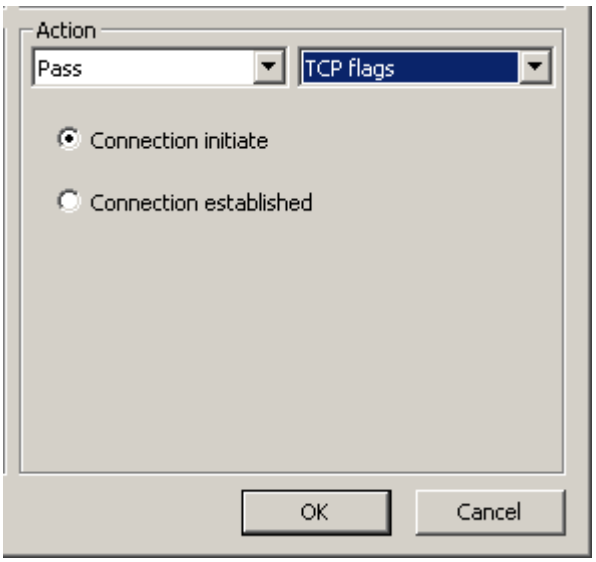

Рисунок 18

Действие **Classify mark**:

Для указания приоритета обработки пакета и эффективного продвижения пакетов по маршруту от одного узла сети к другому, для обеспечения качества обслуживания (QoS) выполняется маркирование пакетов. Для маркирования используется либо значение *IP Precedence (IPP),* либо значение *DSCP*.

На Рисунок 19 показана область *Action* при выборе действия *Classify mark*. Биты ToS-байта пронумерованы так же, как в документе по RFC 2474. Для маркировки следует использовать биты с 0 по 2 (IPP) или с 0 по 5 (DSCP), биты 6 и 7 зарезервированы и используются для специальной сигнализации, и для них переключатель должен стоять в положении «retain». В противном случае, при попытке сохранения правила выдается предупреждение: «Two less significant bits of TOS byte should be retained unless you know exactly what you are doing. Do you want to proceed?»

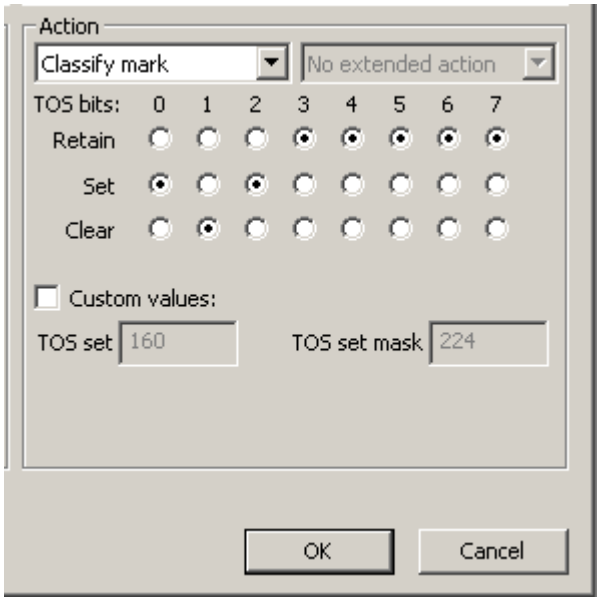

Рисунок 19

Для указания значений битов можно использовать либо переключатель с тремя положениями, либо флажок *Custom values*. Переключатель имеет 3 положения:

- *Retain* – оставить значение бита без изменений, как в пришедшем пакете.
- *Set* – в бите установить значение 1.
- *Clear* – в бите установить значение 0.

При изменении значений битов с помощью переключателя, в поле *TOS set* отображается значение байта в десятичном виде, а в поле *TOS set mask* – значение маски в десятичном виде. При установке флажка *Custom values* в поля *TOS set* и *TOS set mask* значения нужно ввести вручную в десятичном виде.

Значение IP Precedence

Значение *IP Precedence (IPP)* вносится в старшие 3 бита ToS-байта IP-заголовка пакета – в биты с номерами от 0 до 2. В Таблица 2 указаны значения приоритетов.

Таблица 2

Приоритет	Ключевое слово	Рекомендация к использованию	IPP (0-2 биты)
0	Routine – обычный пакет	По умолчанию	000
1	Priority – приоритетный (предпочтительный) пакет	Для приложений данных	001
2	Immediate – немедленный пакет	Для приложений данных	010

3	Flash – мгновенный (срочный) пакет	Для сигнализации вызовов	100
4	Flash-override – быстрее, чем мгновенный (экстренный) пакет	Для видеоконференций и потокового видео	100
5	Critical – критический пакет	Для голосового трафика	101
6	Internet – пакет межсетевого управления	Зарезервирован, не используется	
7	Network – пакет управляющей информации	Зарезервирован, не используется	

Значение DSCP

Значение *DSCP* вносится в старшие 6 битов ToS-байта – в биты с номерами от 0 до 5 (Рисунок 20). В Таблица 3 и Таблица 4 указаны значения *DSCP* для модели дифференцированного обслуживания (DiffServ).

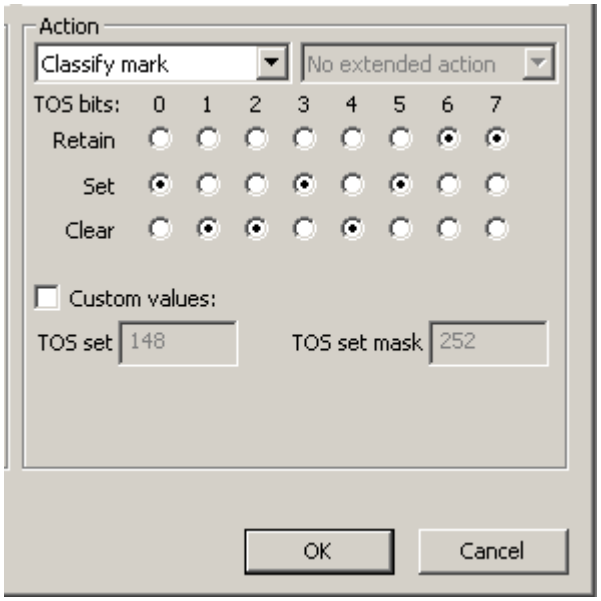

Рисунок 20

Дифференцированное обслуживание не гарантирует определенный уровень сервиса, а стремится упорядочить весь трафик по классам таким образом, чтобы каждый класс получил лучший или худший уровень обслуживания по отношению к остальным.

Значение *DSCP* может быть выражено в цифровой форме или с использованием специальных ключевых слов, называемых поведением сетевых участков (PHB – Per-Hop Behavior). Определено три класса *DSCP* маркировки (Таблица 3):

- доставка по возможности (BE – Best Effort или DSCP 0);
- гарантированная доставка (AF – Assured Forwarding) (RFC 2597);
- срочная доставка (EF – Expedited Forwarding) (RFC 2598).

В дополнение к этим трем определенным классам существуют коды селектора классов (CS1-CS7), которые идентичны значениям *IP Precedence* (1-7).

В гарантированной доставке определены еще 4 класса. Обозначение класса начинаются с AF и далее следуют две цифры. Первая цифра определяет AF класс и принимает значения от 1 (низкий приоритет обработки) до 4 (высокий приоритет обработки пакета). Вторая цифра определяет уровень вероятности сброса пакета в пределах каждого класса и принимает значения от 1 (низкая вероятность сброса) до 3 (высокая вероятность сброса) (Таблица 4).

Негарантированная доставка пакетов имеет значение DSCP 0.

Для немедленной передачи пакетов указывается DSCP 101110.

Чем больше значение DSCP, тем больше приоритет обслуживания. Иногда такое количество классов избыточно, и последние 3 бита заполняют нулями.

Таблица 3

Код селектора классов (CS)	**Описание**		**PHB-политика**	**DSCP**
DSCP 0	Best Effort (BE) – default – 000000		PHB-политика негарантированной доставки пакетов, доставка по возможности. Рекомендуется для трафика данных – передача файлов, приложения электронной почты, HTTP и др.	000000
CS1	Class 1	Assured Forwarding (AF)	PHB-политика гарантированной доставки пакетов. Используется для видеотрафика, видеоконференций и рекомендуется значение DSCP 100010 (AF41).	100010 см Таблица 4
CS2	Class 2			
CS3	Class 3			
CS4	Class 4			
CS5	Express Forwarding (EF) – 101110		PHB-политика немедленной передачи пакетов, срочная доставка. Рекомендуется для голосового трафика	101110
CS6	Stays the same (used for IP routing protocols)			
CS7	Stays the same (link layer and routing protocol keep alive)			

Таблица 4

Классы гарантированной доставки пакетов

Приоритет отбрасывания пакета	**Приоритет обработки**			
	Class 1 (низкий)	**Class 2**	**Class 3**	**Class 4 (высокий)**
Низкий	001010 AF11	010010 AF21	011010 AF31	100010 AF41
Средний	001100 AF12	010100 AF 22	011100 AF32	100100 AF42
Высокий	001110 AF13	010110 AF23	011110 AF33	100110 AF43

Класс 4 обрабатывается более приоритетно, чем класс 3, класс 3 – более приоритетно, чем класс 2 и т.д.

7.3.3.2 Пакетная и контекстная фильтрация. Задание правил в Outbound filter и Inbound filter

В наборах правил *Outbound filter* или *Inbound filter* вкладки **Firewall Rules** (Рисунок 12) задаются правила пакетной и контекстной фильтрации трафика. При нажатии кнопки **Add** появляется окно **Add Rule** (Рисунок 21) для создания правила. Это окно такое же, как и для наборов правил *Outbound classification* и *Inbound classification* (Рисунок 13), отличается только областью *Action*. Поэтому только область *Action* и опишем далее подробно.

Задание действия в правиле

К трафику, заданному в областях *Local IP Addresses*, *Partner IP Addresses*, *Services and Protocols*, применяется действие из области *Action*. Из выпадающего списка выбирается одно из действий:

- **Pass** – пропускать пакет.
- **Drop** – не пропускать пакет.
- **Classify mark** – маркировать пакет.
- **Inspect TCP** – проверять TCP-трафик.
- **Inspect FTP** – проверять FTP-трафик.

Действия *Pass*, *Drop*, *Classify mark* были ранее описаны в разделе «Задание действия в правиле» в предыдущем параграфе.

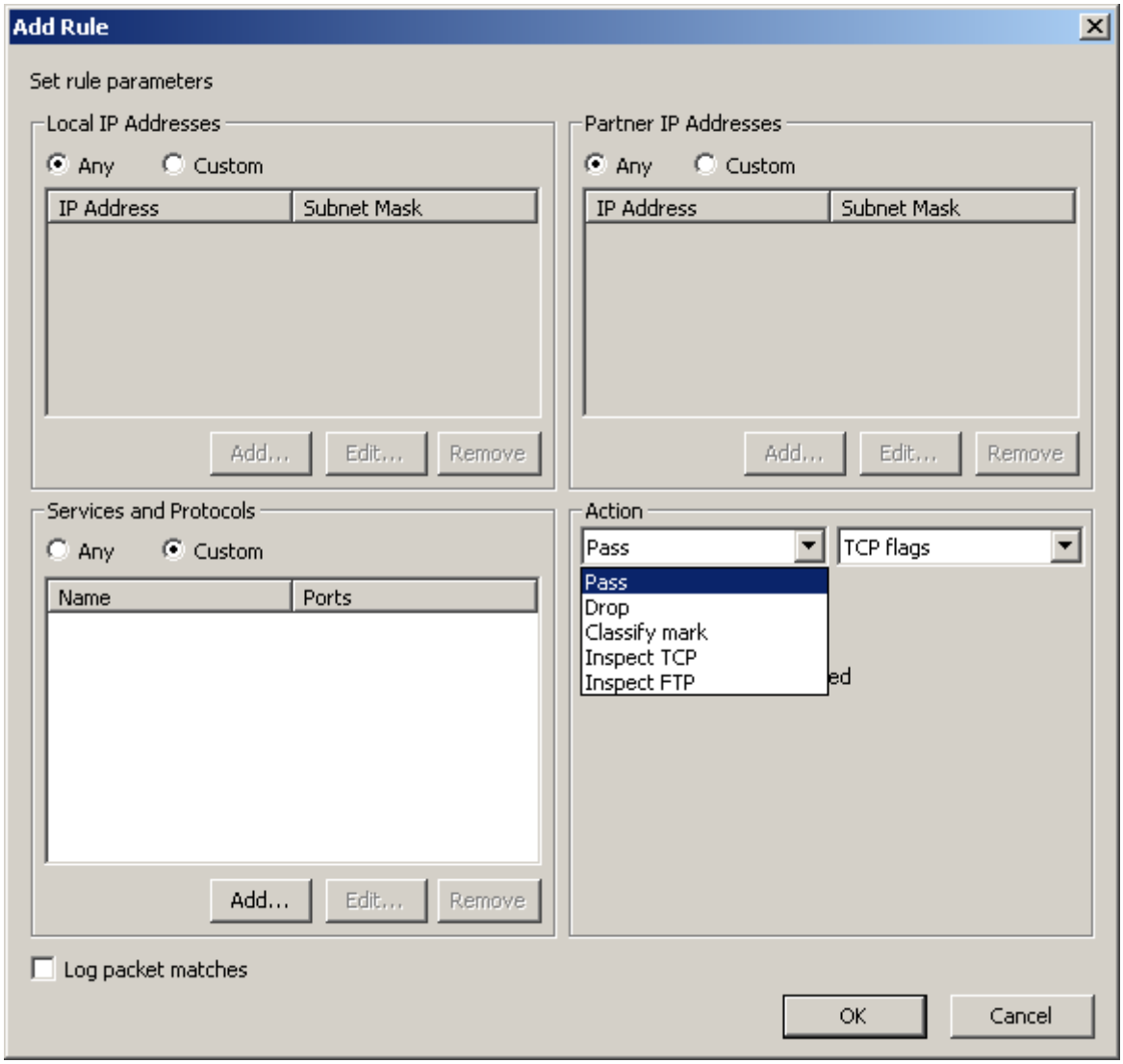

Рисунок 21

Действия *Inspect TCP* и *Inspect FTP* используются для создания правил контекстной фильтрации. Этими правилами будут проверяться только TCP-пакеты и FTP-пакеты.

Действие **Inspect TCP** – при выборе этого действия отслеживается состояние TCP-соединения, меняется время жизни записи о соединении в соответствии с текущим состоянием соединения. Для пропуска пакетов в обе стороны, добавляются динамические правила фильтрации для входящего и исходящего трафика. Динамические правила удаляются вместе с записью о соединении.

Действие **Inspect FTP** – при выборе этого действия отслеживается состояние FTP-соединения, меняется время жизни записи о соединении в соответствии с текущим состоянием соединения. Для пропуска пакетов в обе стороны, добавляются динамические правила фильтрации для входящего и исходящего трафика. Динамические правила удаляются вместе с записью о соединении. Кроме того, отслеживаются некоторые команды FTP, создаются правила для пропуска соединения для данных FTP, определяются и блокируются некоторые подозрительные команды, которые могут являться атакой на FTP сервер.

Если нужно разрешить только исходящий с клиента TCP-трафик и ответный на него трафик, то разрешительное правило контекстной фильтрации должно быть только в разделе Outbound filter, в разделе Inbound filter такого правила быть не должно. Если разрешительного правила нет, значит трафик запрещен.

При выборе действия *Inspect TCP* (*Inspect FTP*) область *Action* приобретает следующий вид (Рисунок 22). Второй выпадающий список с флагами – не активен.

- *Audit* – при установке этого флажка при закрытии соединения создаются сообщения со статистической информацией, выдаваемые в syslog-файл.
- *No alert* – при установке этого флажка не выдаются сообщения о потенциальных атаках (попытках взлома).
- *Timeout* – переопределяет время жизни в секундах установленного соединения по данному правилу. (Глобальные настройки для всех соединений установлены во вкладке **LSP** – кнопка Advanced).

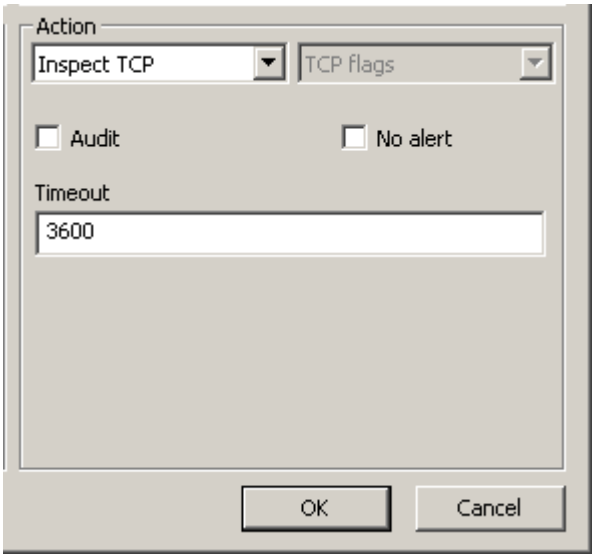

Рисунок 22

7.3.4 Вкладка IPsec Rules

Во вкладке **IPsec Rules** задаются правила для защиты трафика с использованием протокола IPsec. Допустимые алгоритмы протоколов AH и ESP задаются во вкладке IPsec.

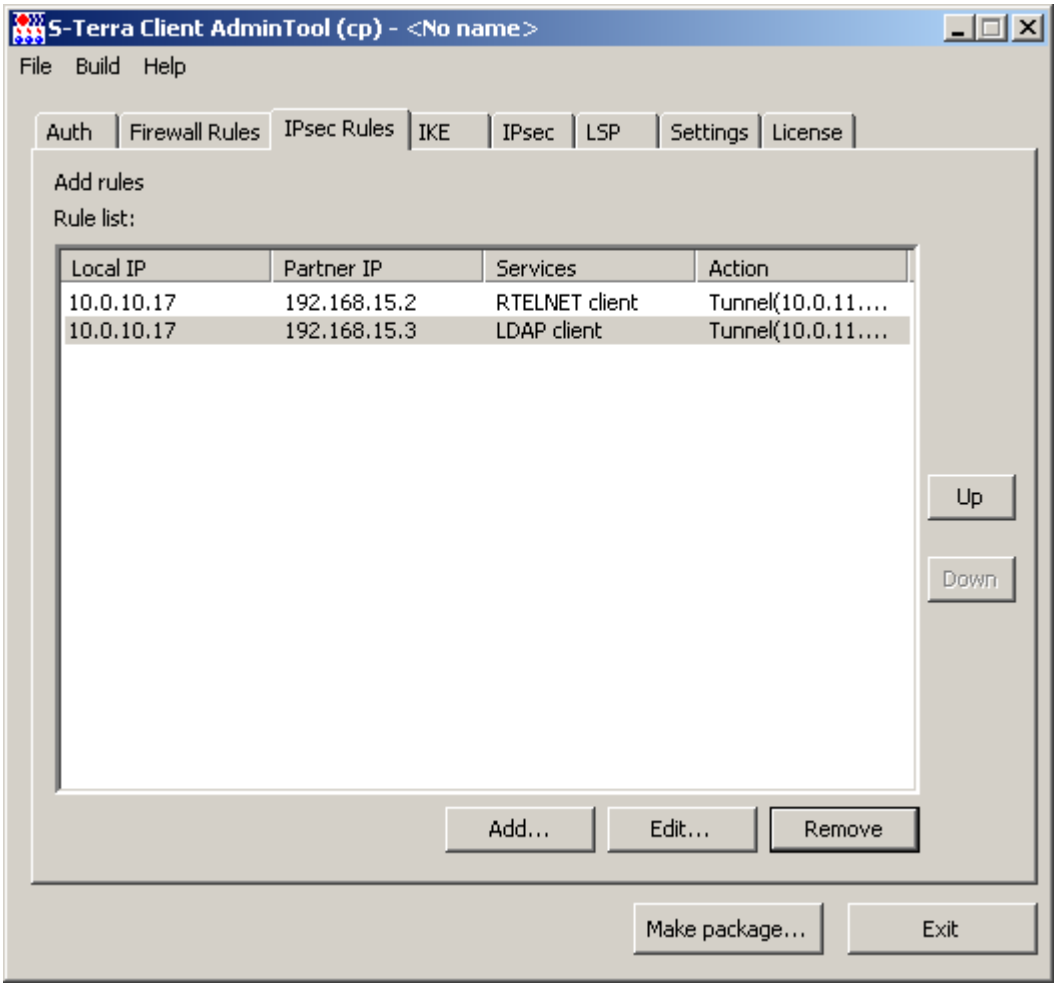

Рисунок 23

7.3.4.1 Создание и редактирование правила

Создание и редактирование правила производится в окне **Add/Edit Rule**, которое вызывается кнопкой **Add** или **Edit** во вкладке **IPsec Rules**:

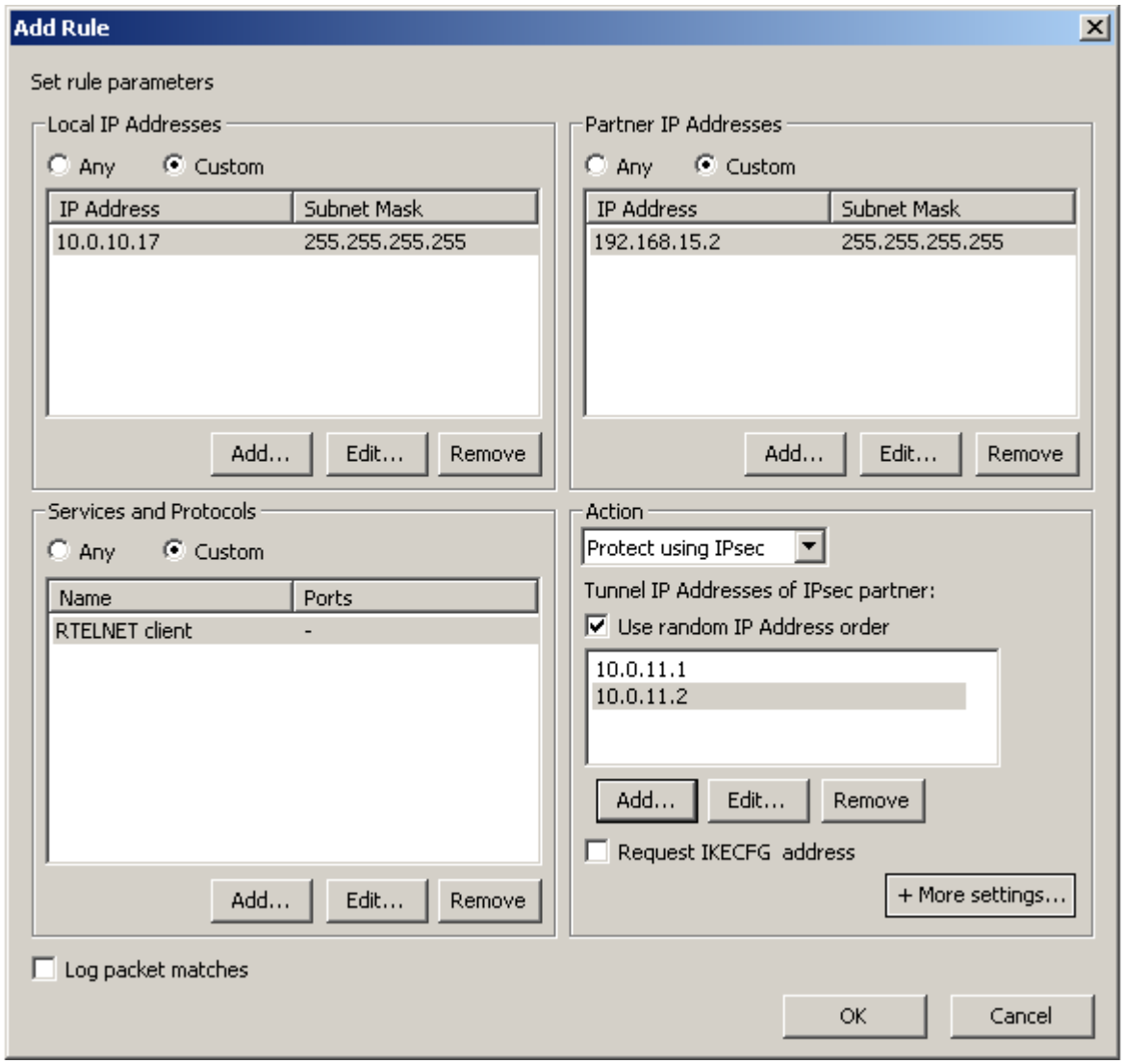

Рисунок 24

Области *Local IP Addresses*, *Partner IP Addresses*, *Services and Protocols* такие же, как во вкладке **Firewall Rules**. Отличается только область *Action*.

Задание действия в правиле

В области *Action* задается действие, которое будет применено к пакету, если пакет подпадает под действие данного правила (Рисунок 24). Из выпадающего списка выбирается одно из действий:

- **Pass** – пропускать трафик без обработки.
- **Drop** – не пропускать трафик.

- **Protect using IPsec** – защищать трафик с использованием протоколов IPsec (алгоритмы протоколов AH и ESP задаются во вкладке IPsec). Трафик между хостом с локальным IP-адресом и IP-адресом партнера защищается на интервале между локальным IP-адресом и туннельным IPsec адресом партнера (это может быть адрес интерфейса шлюза безопасности, защищающего подсеть, в которой находится партнер). В результате этого строится защищенное IPsec соединение – IPsec SA (туннель) (Рисунок 25).

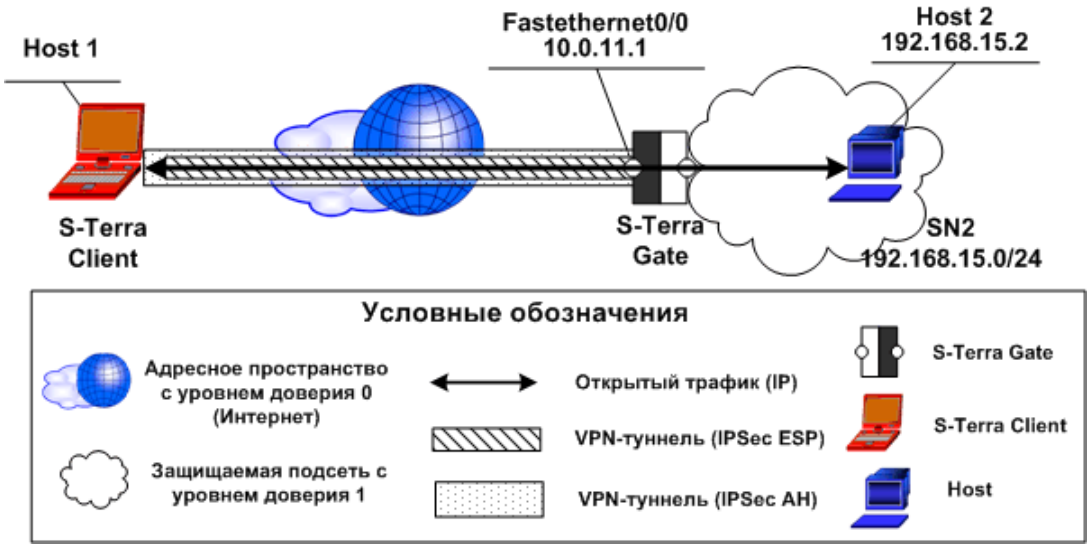

Рисунок 25

Для указания туннельного IPsec адреса партнера нажмите кнопку *Add* в области *Action* и в открывшемся окне **Add IP Address** (Рисунок 26) укажите IP-адрес интерфейса, до которого будет построен туннель от локального IP-адреса клиента. Адрес не может быть нулевым.

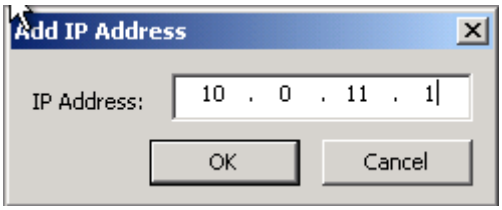

Рисунок 26

Можно указать список IP-адресов, до которых возможно построить туннель. Адреса в списке надо расположить в порядке убывания приоритета – первый в списке имеет самый высокий приоритет. Если не удалось построить туннель до интерфейса с первым указанным адресом, производится попытка построить туннель со вторым IP-адресом и т.д. Кнопки *Up* и *Down* предназначены для изменения приоритета адресов в списке (Рисунок 27).

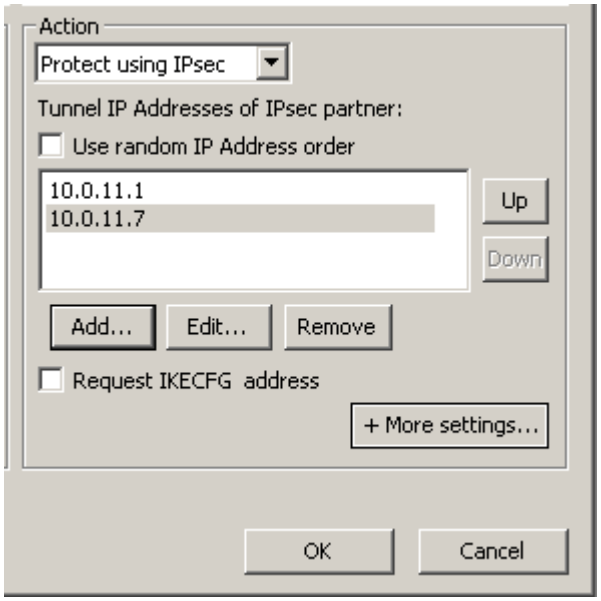

Рисунок 27

Используя кнопки *Add*, *Edit* и *Delete*, можно добавлять, редактировать и удалять адреса из списка.

Use random IP Address order – при установке этого флажка IPsec-адрес партнера будет выбираться из списка случайным образом. При неудачной попытке построить туннель с этим адресом, следующий туннельный адрес будет выбираться также случайным образом.

Request IKECFG address – установка этого флажка позволяет клиенту запрашивать адрес у IKECFG-сервера при построении защищенного соединения по данному правилу. Поэтому в этом правиле обязательно нужно указать туннельный адрес партнера, у которого и будет запрошен IKECFG-адрес (Рисунок 28). Интерфейс, с присвоенным ему IKECFG-адресом, будем называть виртуальным интерфейсом.


Note

Существует ограничение в применении Продукта S-Terra Client, когда запрашивается адрес из IKECFG-пула:

Можно задать только одно правило с запросом IKECFG-адреса, причем в этом правиле нельзя задать фильтрацию по локальным адресам, протоколам и портам (сервисам) (положение переключателя *Custom* недоступно). Это связано с атрибутом PersistentConnection=TRUE в структуре *IPsecAction*.

Созданная политика безопасности будет неработоспособна, если весь трафик защищается по протоколу IPsec (трафик между любым локальным IP-адресом и любыми адресами партнеров, указанными в областях *Local IP Addresses* и *Partner IP Addresses*).

Политика безопасности не будет работать, если туннельный IPsec адрес партнера (*Tunnel IP Addresses of IPsec partner*) совпадает с IP-адресом или подсетью партнера (*Partner IP Addresses*), на которые распространяется правило фильтрации.

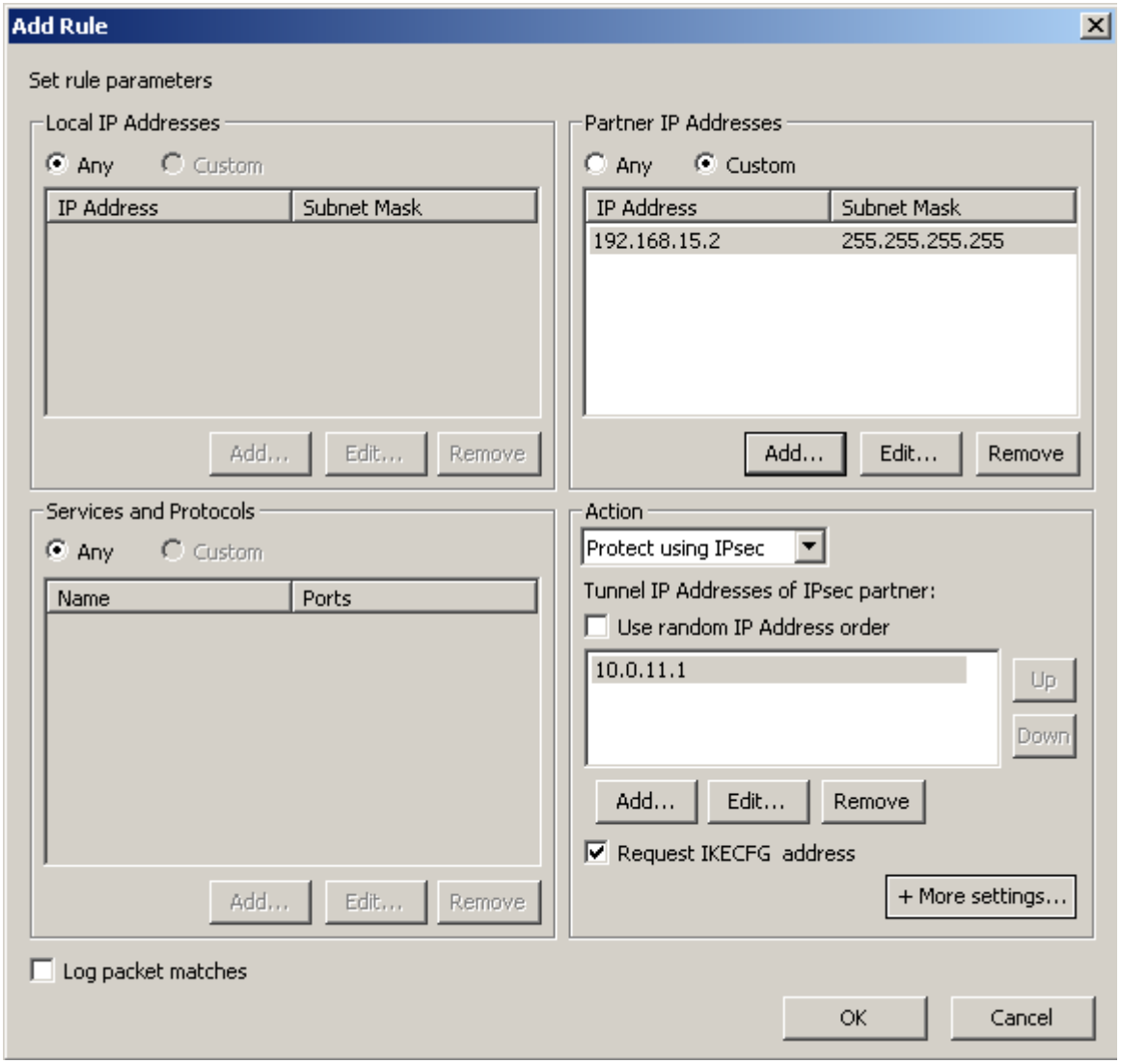

Рисунок 28

+More settings – при нажатии на кнопку *+More settings* (Рисунок 28) появляется окно (Рисунок 29) для дополнительных настроек.

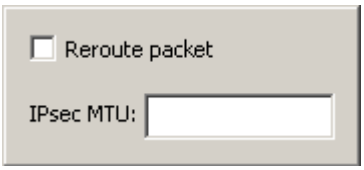

Рисунок 29

- **Reroute packet** – при установке этого флажка пакет будет подвергаться повторной маршрутизации (при создании вложенного туннеля или при отправке пакета с виртуального IKECFG интерфейса после IPsec обработки). При создании вложенного туннеля для исходящего пакета опять выполняется поиск подходящего правила из набора правил *Outbound classification*, а затем из вкладки **IPsecRules**. С виртуального IKECFG интерфейса после IPsec обработки пакет не может быть отправлен в интернет, поэтому он перенаправляется на другой интерфейс.
- **IPsec MTU** – в этом поле можно задать значение MTU для IPsec SA, построенному по данному правилу. Допустимые значения MTU – от 0 до 65535. Значение по умолчанию – 0, при этом MTU будет определяться автоматически, и в LSP в структуре *IPsecAction* значение MTU не указывается.

7.3.5 Вкладка IKE

В этой вкладке определены наборы политик для защиты соединений IKE, которые предлагаются партнеру для согласования при создании ISAKMP SA.

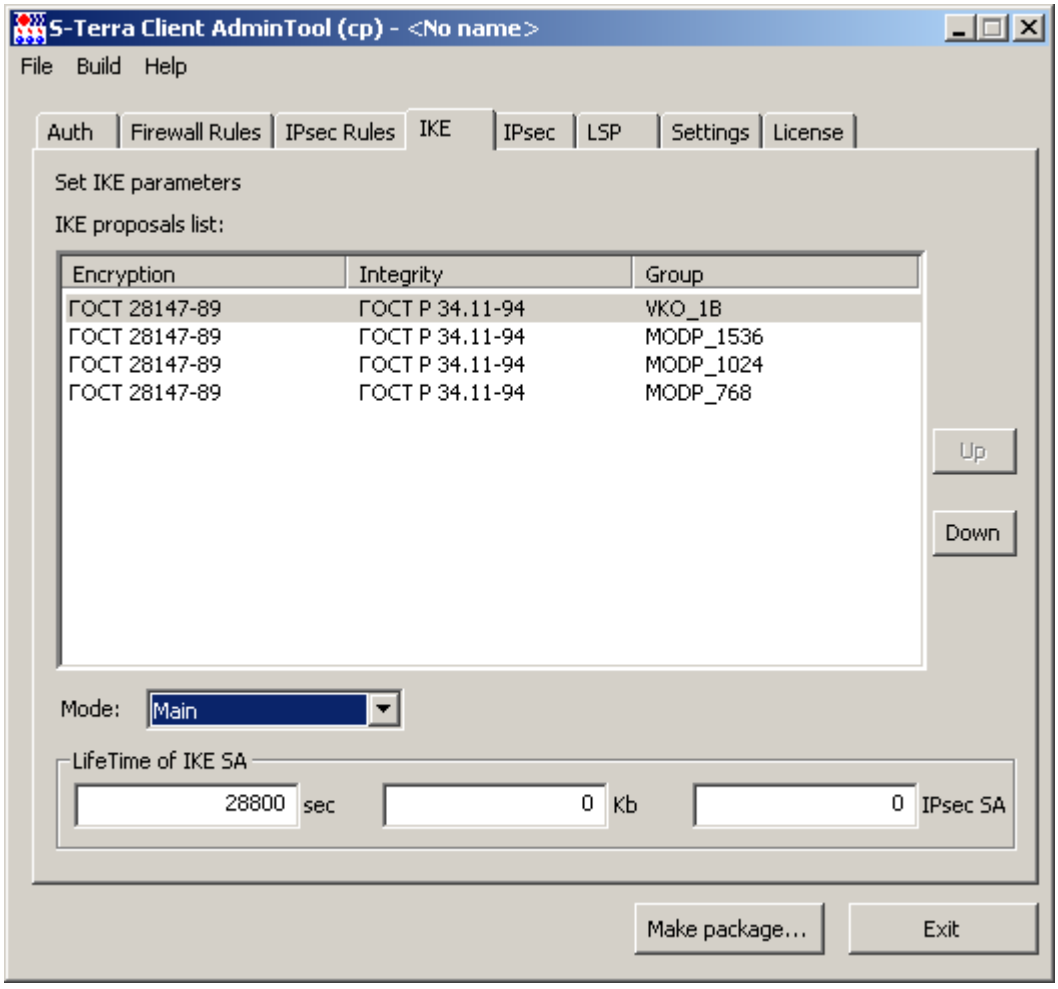

Рисунок 30

IKE proposals list – упорядоченный список IKE предложений по приоритету. В верхней строчке находится предложение с наивысшим приоритетом.

Encryption – предлагаемые алгоритмы шифрования пакетов. Предлагается только один российский криптографический алгоритм:

- *ГОСТ 28147-89* – российский криптографический алгоритм, представленный в конфигурации (во вкладке LSP) как G2814789CPRO1-K256-CBC-65534.

Integrity – предлагаемые алгоритмы проверки целостности пакетов. Предлагается только один российский криптографический алгоритм:

- *ГОСТ Р 34.11-94* – российский криптографический алгоритм, представленный в конфигурации (во вкладке LSP) как GR341194CPRO1-65534.

Имена алгоритмов шифрования пакетов и проверки целостности данных считываются из файла `admintool.ini`, размещенного в папке Продукта S-Terra Client AdminTool cp. Для изменения имен алгоритмов необходимо отредактировать этот файл, описанный в разделе «Формат задания имен алгоритмов в файле admintool.ini» , и перезапустить графический интерфейс.

Group – параметры выработки общего сессионного ключа:

- *VKO_1B* – используется алгоритм VKO ГОСТ Р 34.10-2001 [RFC4357], (длина ключа 256 бит).
- *MODP_768* – группа 1 (768-битовый вариант алгоритма Диффи-Хеллмана).
- *MODP_1024* – группа 2 (1024-битовый вариант алгоритма Диффи-Хеллмана).
- *MODP_1536* – группа 5 (1536-битовый вариант алгоритма Диффи-Хеллмана).

Mode – режим обмена информацией о параметрах защиты и установления IKE SA. Имеет два значения:

- *Main* – в этом режиме партнеру высылаются все IKE политики для выбора и согласования.
- *Aggresssive* – в этом режиме партнеру высылается только первая IKE политика из списка, имеющая самый высокий приоритет. При выборе этого режима выдается об этом предупреждение. Если для аутентификации используется предопределенный ключ и выбран тип идентификатора *KeyID*, то должен использоваться только режим *Aggressive*.

LifeTime of IKE SA (sec) – время в секундах, в течение которого ISAKMP SA будет существовать. Возможное значение – целое число из диапазона 0..2147483647. Рекомендуемое значение – 28800, которое выставлено при открытии нового проекта. Значение 0 означает, что время действия SA не ограничено. Пустая строка – недопустима, при создании инсталляционного файла будет выдано сообщение об ошибке.

LifeTime of IKE SA (Kb) – указывает объем данных в килобайтах, который могут передать стороны во всех IPsec SA, созданных в рамках одного ISAKMP SA. Возможное значение – целое число из диапазона 0..2147483647. Рекомендуемое значение – 0, которое выставлено при открытии нового проекта. Значение 0 означает, что объем данных в килобайтах не ограничен. Пустая строка – недопустима, при создании инсталляционного файла будет выдано сообщение об ошибке.

IPsec SA – количество IPsec SA, созданных в рамках одного ISAKMP SA. Значение 0 означает, что количество IPsec SA не ограничено.

Кнопки **Up** и **Down** предназначены для упорядочивания списка предложений по приоритету.

7.3.6 Вкладка IPsec

В данной вкладке задаются политики IPsec защиты в виде набора преобразований, каждый из которых есть комбинация AH преобразования и ESP преобразования. Партнеру направляется список наборов преобразований, по протоколу IKE происходит согласование и выбор конкретного набора преобразований, который будет использоваться для защиты трафика для одного SA.

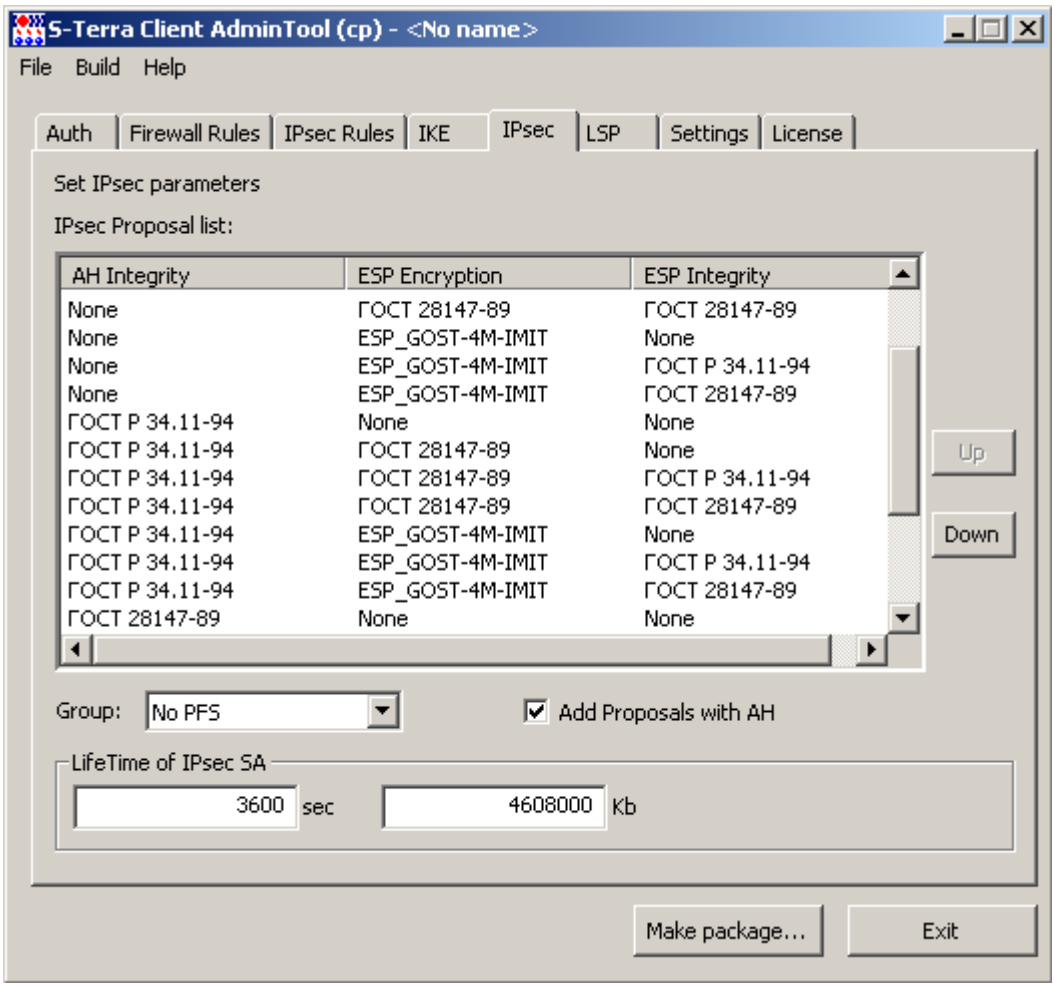

Рисунок 31

IPsec Proposal list – упорядоченный список наборов преобразований, высылаемых партнеру для согласования. При помощи кнопок *Up* и *Down* выполняется упорядочивание списка по приоритету. В верхней строчке находится набор преобразований с наивысшим приоритетом.

AH Integrity – предлагаемые алгоритмы проверки целостности пакета по протоколу AH: Имеется три значения:

- *None* – алгоритм проверки целостности не применяется.
- *ГОСТ Р 34.11-94* – российский криптографический алгоритм, представленный в конфигурации (во вкладке LSP) как GR341194CPRO1-H96-HMAC-254.
    - *ГОСТ 28147-89* (в режиме выработки имитовставки) – российский криптографический алгоритм, представленный в конфигурации (во вкладке LSP) как G2814789CPRO1-K256-MAC-255.

ESP Integrity – предлагаемые алгоритмы проверки целостности пакета по протоколу ESP:

- *None* – алгоритм проверки целостности не применяется.
- *ГОСТ Р 34.11-94* – российский криптографический алгоритм, представленный в конфигурации (во вкладке LSP) как GR341194CPRO1-H96-HMAC-65534.

- *ГОСТ 28147-89* (в режиме выработки имитовставки) – российский криптографический алгоритм, представленный в конфигурации (вкладке LSP) как G2814789CPRO1-K256-MAC-65535.

ESP Encryption – предлагаемые алгоритмы шифрования пакетов по протоколу ESP:

- *None* – алгоритм шифрования ESP не применяется.
- *Null* – алгоритм применять, но не шифровать.
- *ГОСТ 28147-89* (в режиме простой замены с зацеплением) – российский криптографический алгоритм, представленный в конфигурации (во вкладке LSP) как G2814789CPRO1-K256-CBC-254.
- *ESP_GOST-4M-IMIT* – российский криптографический алгоритм, представленный в конфигурации (во вкладке LSP) как G2814789CPRO1-K288-CNTMAC-253. Криптографический алгоритм ESP_GOST-4M-IMIT самостоятельно обеспечивает как защиту конфиденциальности (шифрование), так и контроль целостности данных (имитозащиту).

Имена алгоритмов шифрования пакетов и проверки целостности данных считываются из файла `admintool.ini`, размещенного в папке Продукта S-Terra Client AdminTool ср. Для изменения имен алгоритмов необходимо отредактировать этот файл, описанный в разделе «Формат задания имен алгоритмов в файле admintool.ini», и перезапустить графический интерфейс.

Add Proposals with AH – при установке этого флажка выводится сообщение:

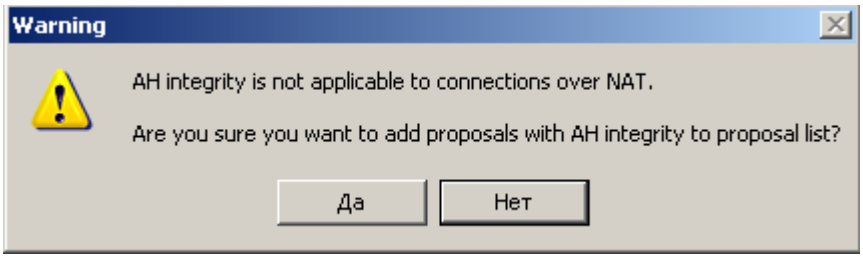

Рисунок 32

Оно означает, что протокол AH несовместим со средствами NAT, так как NAT изменяют IP-адрес в заголовке TCP/IP пакета. Протокол AH обеспечивает проверку аутентичности и целостности пакетов, а NAT нарушает данные аутентификации. После нажатия кнопки *Yes* добавляются российские криптографические алгоритмы ГОСТ Р 34.11-94 и ГОСТ 28147-89.

Group – параметры выработки ключевого материала, высылаемые партнеру для согласования:

- *No PFS* – опция PFS не включена и при согласовании новой SA новый обмен по алгоритму Диффи-Хеллмана или VKO для выработки общего сессионного ключа не выполняется. Ключевой материал заимствуется из первой фазы IKE.
- Выбранный параметр означает, что при согласовании новой SA выполняется новый обмен ключами по алгоритму Диффи-Хеллмана или VKO_1B в рамках IPsec. Может использоваться один из параметров:
  - *VKO_1B* – используется алгоритм VKO ГОСТ Р 34.10-2001 [RFC4357], (длина ключа 256 бит).
  - *MODP_768* – группа 1 (768-битовый вариант алгоритма Диффи-Хеллмана).
  - *MODP_1024* – группа 2 (1024-битовый вариант алгоритма Диффи-Хеллмана).
  - *MODP_1536* – группа 5 (1536-битовый вариант алгоритма Диффи-Хеллмана).

LifeTime of IPsec SA (sec) – время в секундах, в течение которого IPsec SA будет существовать. Возможное значение – целое число из диапазона 1..2147483647. Рекомендуемое значение – 3600, которое выставлено при открытии нового проекта. Пустая строка и значение 0, которое означает неограниченное время жизни IPsec SA, – недопустимы, при создании инсталляционного файла будет выдано сообщение об ошибке.

Рекомендуется указывать такое время SA жизни в секундах, что бы в основном удаление IPsec SA происходило по времени, а ограничение на объем трафика выбирать как дополнительную меру.

LifeTime of IPsec SA (Kb) – указывает объем данных в килобайтах, который могут передать стороны в рамках одной IPsec SA. Возможное значение – целое число из диапазона 0..2147483647. Рекомендуемое значение – 4608000, которое выставлено при открытии нового проекта. Значение 0 означает, что объем данных в килобайтах не ограничен. Пустая строка – недопустима, при создании инсталляционного файла будет выдано сообщение об ошибке.

Примечание

Если для проверки целостности и шифрования используются алгоритмы: ГОСТ 28147-89 (в режиме выработки имитовставки), ГОСТ 28147-89 (в режиме простой замены с зацеплением), то в этом случае максимальное допустимое значение LifeTime of IPsec SA (Kb) – 4032 Кб.

При превышении указанного значения для созданного SA, в журнал протоколирования будет выдано сообщение, что в созданном IPsec SA ограничение по трафику не соответствует допустимому ограничению для используемого криптографического алгоритма:

`"SA traffic limit exceeds limitations imposed by the cryptograghic algorithm".`

Кнопки **Up** и **Down** предназначены для упорядочивания списка предложений по приоритету.

7.3.7 Локальная политика безопасности

Во вкладке **LSP** (Рисунок 33) просматривается и редактируется локальная политика безопасности для пользователя, заданная в предыдущих вкладках.

Существует два режима работы с LSP:

- режим автоматического формирования LSP,
- режим ручного задания LSP.

7.3.7.1 Режим автоматического формирования LSP

В режиме автоматического формирования (флажок "Use custom LSP" не установлен) локальная политика безопасности формируется на основе данных вкладок **Auth**, **Firewall Rules**, **IPsec Rules**, **IKE**, **IPsec** и расширенных параметров, задаваемых в диалоговом окне, вызываемом кнопкой Advanced.

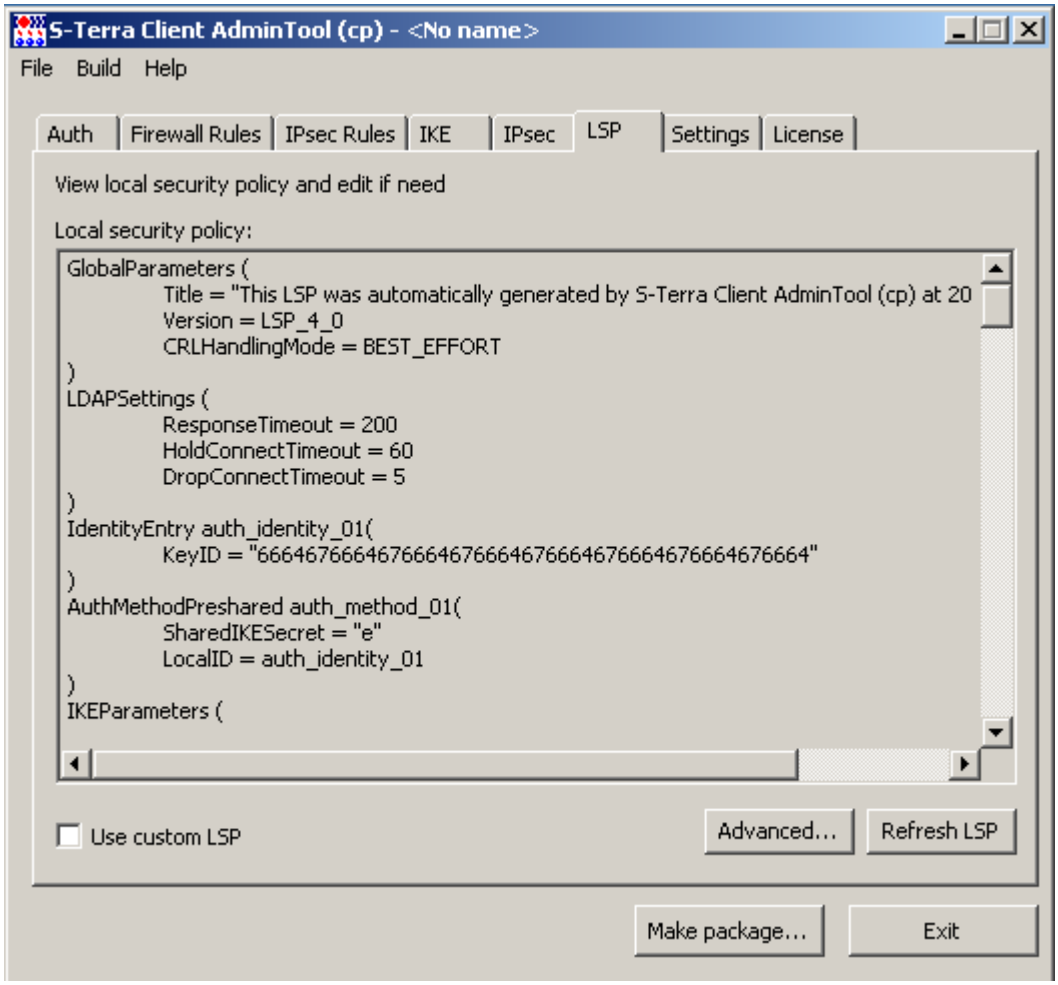

Рисунок 33

Local security policy – поле с LSP в текстовом формате.

Use custom LSP – установка этого флажка переключает в режим ручного формирования LSP.

Refresh LSP – кнопка для обновления LSP в окне **Local security policy** для отображения текущей конфигурации с изменениями.

Advanced – кнопка вызова окна Advanced LSP Settings для настройки расширенного списка параметров LSP.

Advanced LSP Settings

Это окно отображает расширенный список переменных LSP и их текущие значения, которые можно отредактировать и установить значения по умолчанию. Переменные объединены в шесть групп.

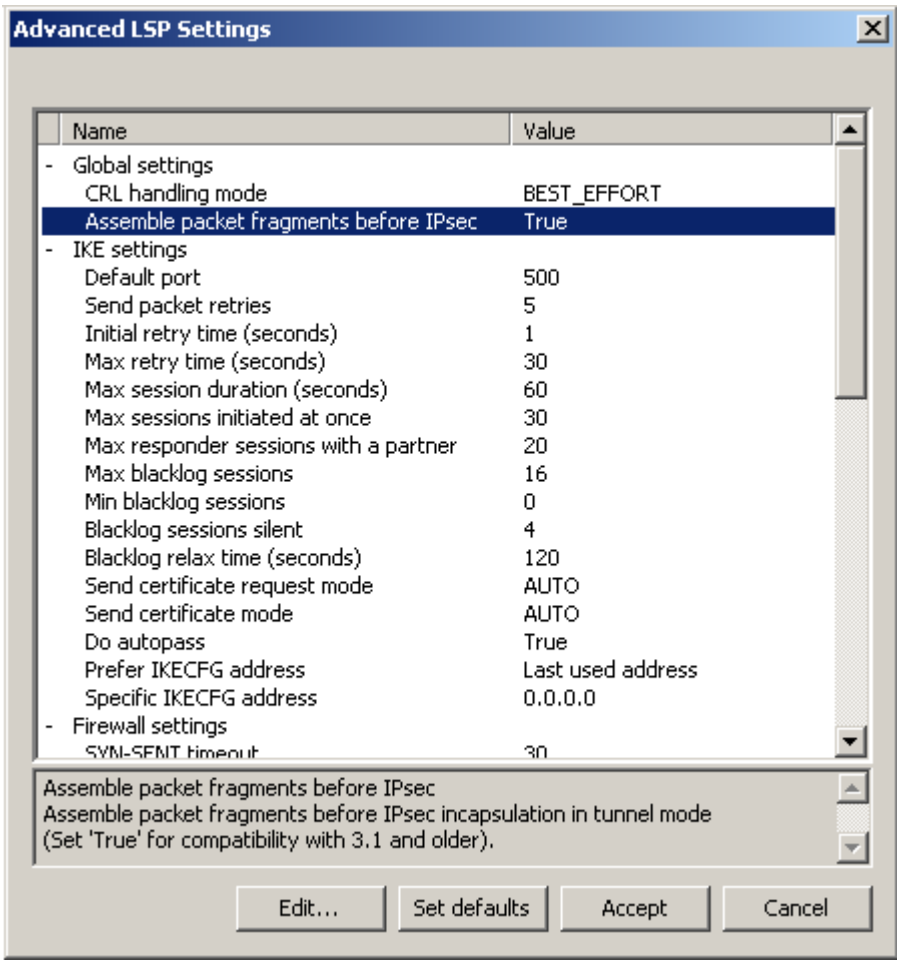

Рисунок 34

Окно содержит 4 функциональные кнопки:

- **Edit** – кнопка вызова окна для редактирования выделенной переменной. Окно редактирования открывается также при двойном клике левой кнопки "мыши" на выделенной строке.
- **Set defaults** – кнопка для установки значений по умолчанию для всех переменных.
- **Accept** – кнопка для закрытия окна с сохранением отредактированных значений переменных.
- **Cancel** – кнопка для закрытия окна без сохранения отредактированных значений переменных.

Global settings

CRL handling mode

Переменная задает режим использования списков отозванных сертификатов (CRL). При нажатии кнопки Edit появляется окно для выбора значений из выпадающего списка:

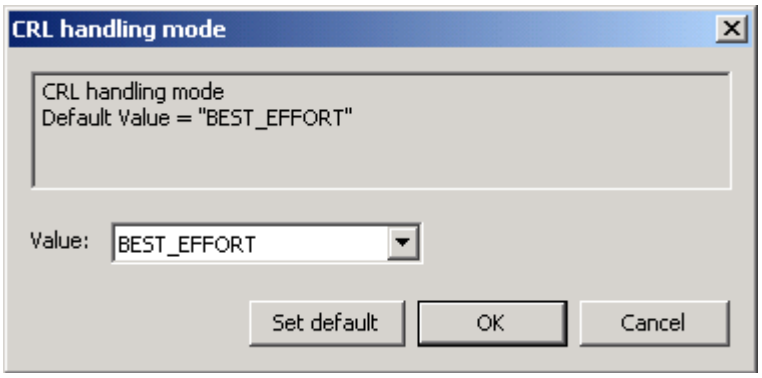

Рисунок 35

Возможные значения:

- *DISABLE* – при проверке сертификата список отозванных сертификатов не обрабатывается.
- *OPTIONAL* – список отозванных сертификатов используется только в случае, если он был предустановлен или получен (и обработан) в процессе IKE обмена и является действующим.
- *BEST_EFFORT* – список отозванных сертификатов используется при проверке сертификата только в том случае, если он является действующим. Этот режим отличается от режима OPTIONAL тем, что CRL может быть получен посредством протокола LDAP (если он настроен). Это значение используется по умолчанию.
- *ENABLE* – для успешной проверки сертификата обрабатывается список отозванных сертификатов.

Значение по умолчанию – *BEST_EFFORT*.

Все окна для редактирования переменных имеют три функциональные кнопки:

- **Set default** – кнопка для установления значения по умолчанию данной переменной.
- **OK** – кнопка для закрытия окна с сохранением выбранного значения переменной.
- **Cancel** – кнопка для закрытия окна без сохранения выбранного значения переменной.

Assemble packet fragments before IPsec

Переменная задает сборку IP-пакетов из фрагментов перед IPsec-инкапсуляцией. При нажатии кнопки **Edit** появляется окно с двумя положениями переключателя:

- *FALSE* – сборка пакета не производится.
- *TRUE* – производится сборка пакета из фрагментов.

Значение по умолчанию –*TRUE*.

Для совместимости с версией клиента 3.1 или 3.11 установите значение *TRUE*.

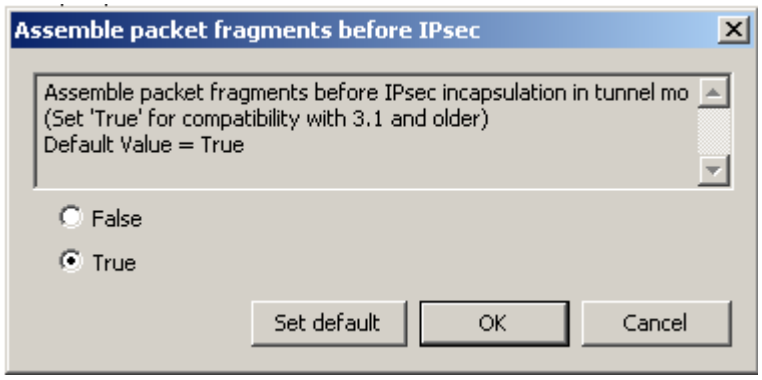

Рисунок 36

IKE settings

Переменные, в названии которых имеется слово blacklog, задают поведение механизма так называемого “черного списка”. "Черный список" предназначен для защиты от DoS-атак (Denial of Service –отказ от обслуживания). "Черный список" минимизирует обработку IKE-пакетов от партнеров, находящихся в "черном списке".

Default port

Порт для протокола IKE, который будет использован по умолчанию. Возможное значение – целое число из диапазона 1..65535. Значение по умолчанию – 500.

Окно для выбора значения порта:

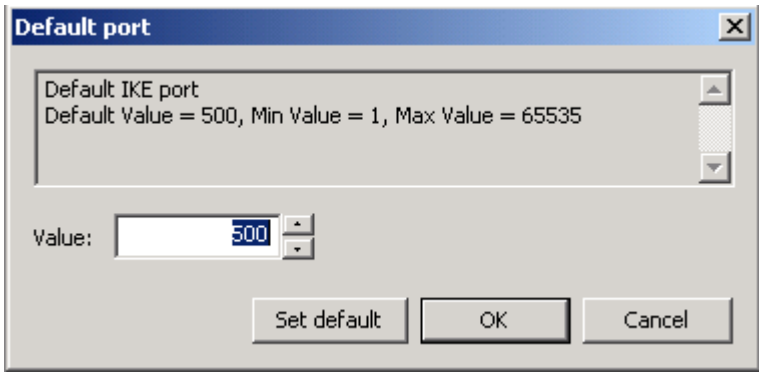

Рисунок 37

Send packet retries

Количество попыток посылки IKE-пакетов партнеру. Возможное значение – целое число из диапазона 1..30. Значение по умолчанию – 5.

Окно для установки значения:

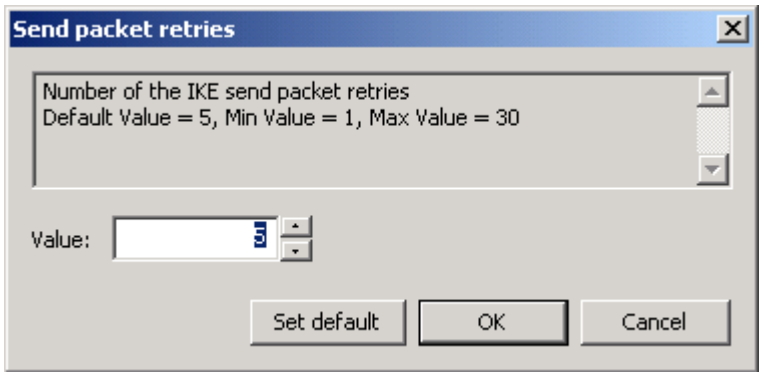

Рисунок 38

Initial retry time (seconds)

Начальный интервал времени между повторными попытками посылки IKE-пакетов партнеру (в секундах). Если ответ не получен в течение начального интервала, то запрос посылается повторно и интервал между повторными попытками увеличивается в два раза. Этот интервал увеличивается в два раза до тех пор, пока:

- не будет получен ответ или

- значение интервала Initial retry time не достигнет значения *Max retry time*, (повторные попытки будут продолжаться с интервалом *Max retry time*) и количество попыток не достигнет значения *Send packet retries*.

Возможное значение – целое число из диапазона 1..5. Значение по умолчанию – 1.

Окно для установки начального интервала времени:

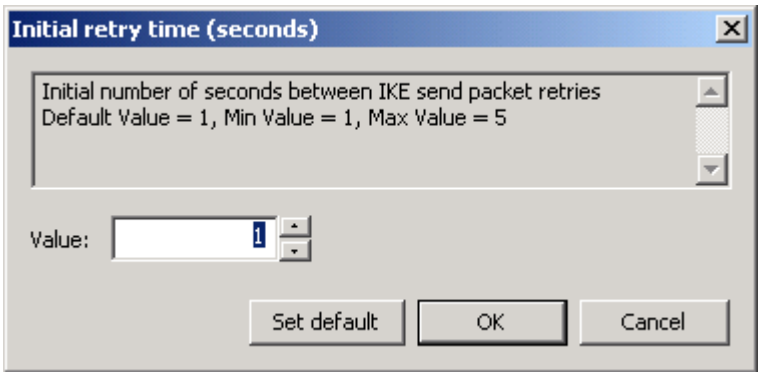

Рисунок 39

Max retry time (seconds)

Максимальный интервал времени между повторными попытками посылки IKE-пакетов партнеру (в секундах). Если выставленное значение *Max retry time* меньше, чем значение *Initial retry time*, то при загрузке конфигурации *Max retry time* присваивается значение *Initial retry time*. Возможное значение – целое число из диапазона 1..60. Значение по умолчанию – 30.

Окно для установки максимального интервала времени:

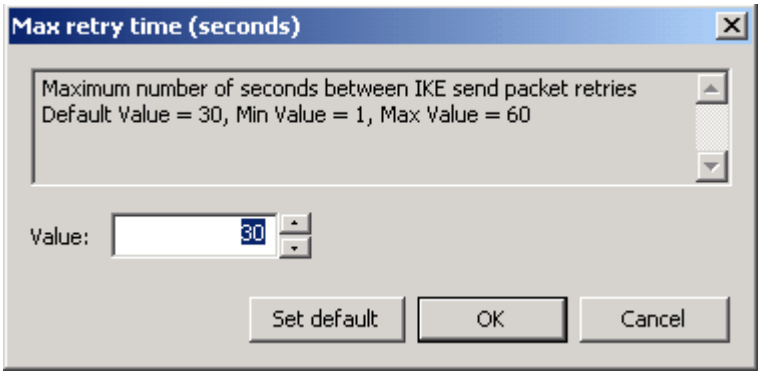

Рисунок 40

Max session duration (seconds)

Максимальный интервал времени на каждую сессию IKE (в секундах). Возможное значение – целое число из диапазона 10..300. Значение по умолчанию – 60. Окно для выбора значения:

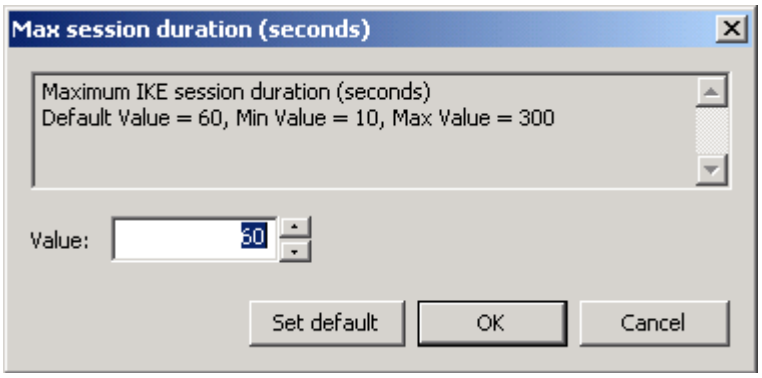

Рисунок 41

Max sessions initiated at once

Максимальное количество одновременно инициируемых IKE-сессий для всех партнёров. Возможное значение – целое число из диапазона 1..10000. Значение по умолчанию – 30. Окно для выбора значения:

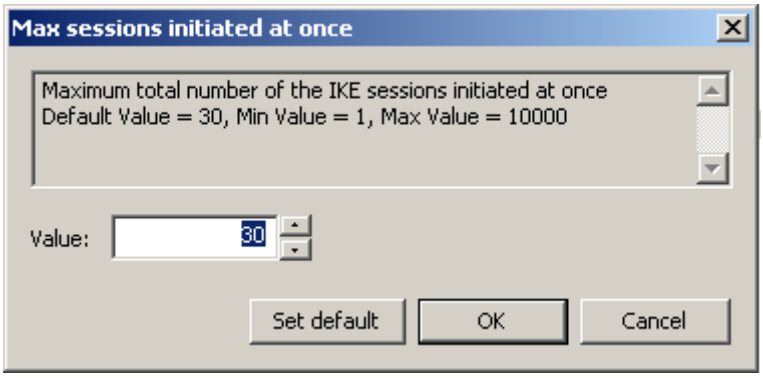

Рисунок 42

Max responder sessions with a partner

Максимально допустимое количество одновременных обменов, проводимых VPN-устройством со всеми партнерами, в качестве ответчика. Если локальное устройство имеет указанное количество незавершенных IKE-обменов в роли ответчика, то все входящие ISAKMP-пакеты, требующие установления новых обменов, игнорируются (без оповещения партнера).

Возможное значение – целое число из диапазона 1..10000. Значение по умолчанию – 20. Окно для установки значения:

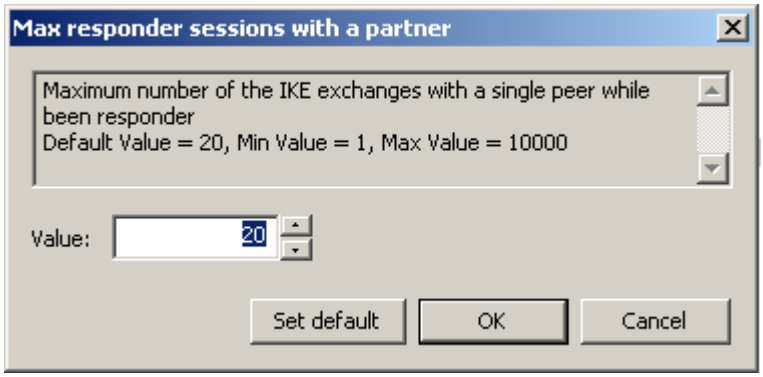

Рисунок 43

Max blacklog sessions

Max blacklog sessions устанавливает начальное число разрешенных одновременных IKE обменов, инициируемых одним партнером[1]. При каждом неудачном завершении IKE обмена число разрешенных одновременных IKE обменов для данного партнера снижается вдвое с округлением в меньшую сторону, вплоть до полного запрещения IKE трафика с данным партнером.

Примечание: как только партнер заносится в "черный список", для него текущее значение разрешенных одновременно проводимых IKE обменов не только начинает уменьшаться в два раза после каждого неуспешного завершения обмена, но и увеличиваться на единицу по истечении каждого интервала времени *Blacklog relax time* (описанного далее).

Возможное значение – целое число из диапазона – 0..2147483647.

Если значение равно 0, то "черный список" не используется.

Если значение *Max blacklog sessions* больше или равно значению *Max responder sessions with a partner,* то *Max blacklog sessions* присваивается значение *Max responder sessions with a partner*.

Значение по умолчанию – 16.

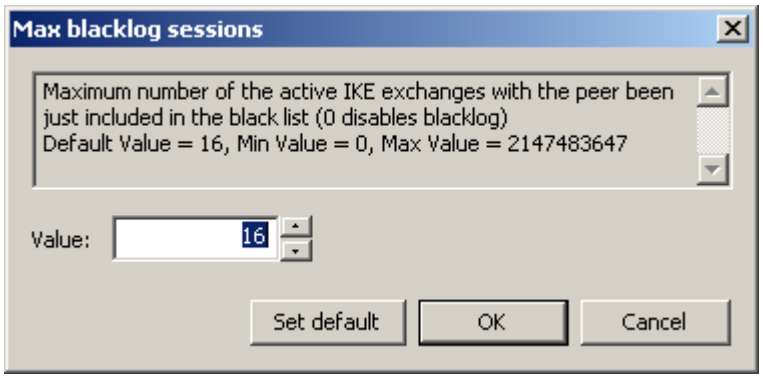

Рисунок 44

Min blacklog sessions

Минимальное число разрешенных одновременных IKE обменов, инициируемых неаутентифицированным партнером.

Возможное значение – целое число из диапазона – 0..2147483647.

Если значение *Min blacklog sessions* равно или больше, чем *Max blacklog sessions*, то число разрешенных одновременных IKE обменов, инициируемых неаутентифицированным партнером, не снижается (т.е. ”черный список“ отключен)[2].

Значение по умолчанию – 0 означает, что для партнера, поведение которого привело к понижению числа разрешенных инициируемых им одновременных IKE обменов до значения *Min blacklog sessions*, игнорируется весь IKE-трафик, а все имеющиеся с ним недостроенные IKE-сессии уничтожаются (ситуация “Access denied”).

Окно для выбора минимального количества обменов с партнером:

[1]В данном случае партнер идентифицируется по паре ip:port. Пока партнер не аутентифицирован (т.е. с таким партнером на данный момент нет ни одного ISAKMP-соединения – SA), допустимое количество IKE-обменов может снижаться в зависимости от того, насколько успешно завершаются IKE-обмены с этим партнером.

[2] При загрузке конфигурации с *отключенным* «черным списком» вся статистическая информация о «плохих» партнерах сбрасывается. Если же «черный список» *включен*, то к уже имеющейся накопленной статистике применяются новые параметры настроек «черного списка».

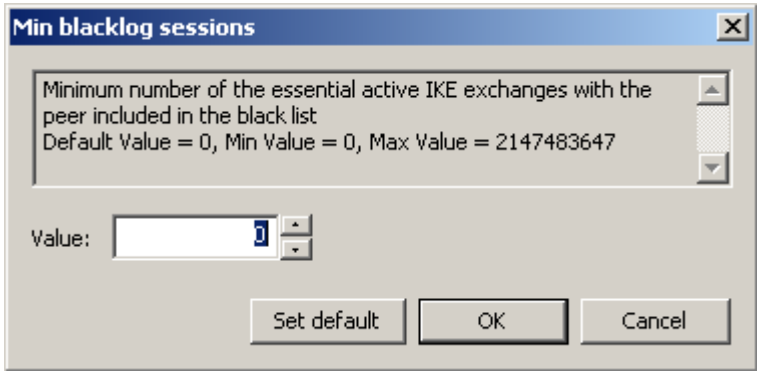

Рисунок 45

Blacklog sessions silent

Число активных обменов, инициированных неаутентифицированным партнером, по достижении которого VPN-устройство перестает информировать партнера о причине неуспешного завершения инициированного им IKE-обмена.

Возможное значение – целое число из диапазона – 0..2147483647.

Если значение *Blacklog sessions silent* больше, чем *Max blacklog sessions*, то *Blacklog sessions silent* присваивается значение *Max blacklog sessions*.
Если значение равно 0, то неаутентифицированный партнер никогда не оповещается о причинах ошибки инициированного им обмена.

Значение по умолчанию – 4.

Окно для выбора количества обменов:

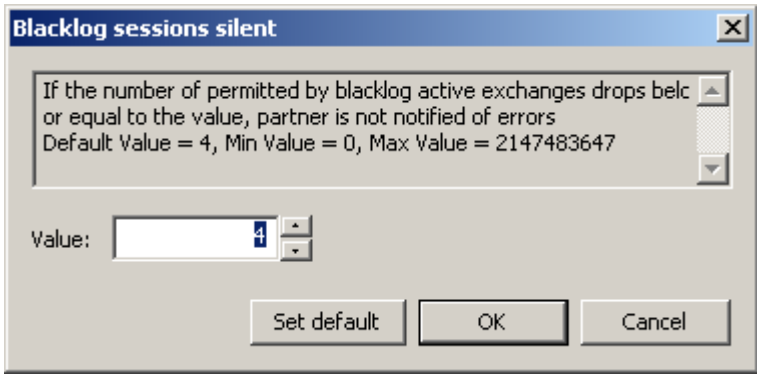

Рисунок 46

Blacklog relax time (seconds)

Устанавливает интервал времени (в секундах) релаксации "черного списка".

За указанный период времени число разрешенных одновременных IKE обменов для каждого партнера, находящегося в "черном списке", увеличивается на единицу. По истечении следующего такого же интервала времени, текущие значения разрешенных одновременно проводимых IKE обменов для каждого партнера опять увеличивается на единицу и т.д. Этот интервал времени отсчитывается с момента последней загрузки конфигурации.

Как только текущее значение разрешенных одновременно проводимых партнером IKE обменов начинает превышать значение *Max blacklog sessions*, такой партнер исключается из "черного списка".

Возможное значение – целое число из диапазона 0..2147483647. Значение 0 означает бесконечное время релаксации "черного списка" (партнер попадает в "черный список" навсегда). Значение по умолчанию – 120 секунд.

Примечание: помимо механизма релаксации, партнер также может быть исключен из "черного списка" в следующих случаях:

- при перезапуске сервиса
- при загрузке конфигурации с отключенным "черным списком"
- при инициации IKE обмена со стороны локального VPN устройства с целью установления ISAKMP (IPsec) соединения[3]
- если партнеру удалось установить ISAKMP (IPsec) соединение с локальным VPN устройством, и тем самым партнер был успешно аутентифицирован.

Окно для выбора интервала времени:

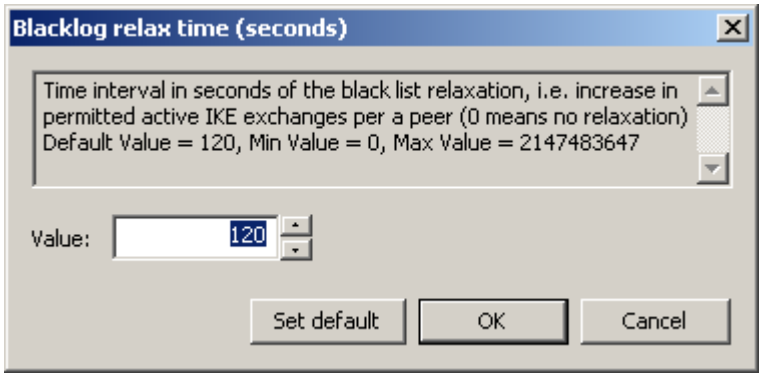

Рисунок 47

Send certificate request mode

Определяет логику отсылки запроса на сертификат партнера. Возможные значения:

- *AUTO* – запрос высылается, если возможный сертификат партнера отсутствует
- *NEVER* – запрос не высылается
- *ALWAYS* – запрос высылается всегда.

Значение по умолчанию – *AUTO*.

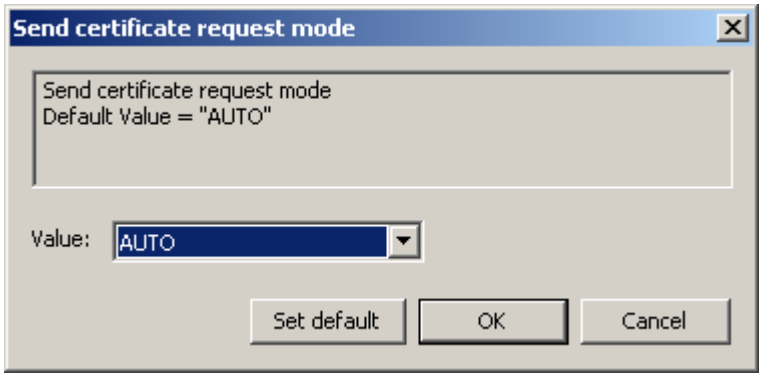

Рисунок 48

[3] В данном случае считается, что локальное VPN устройство потенциально доверяет партнеру, с которым оно хочет установить соединение, и информация, накопленная в "черном списке", для такого партнера сбрасывается.

Send certificate mode

Определяет логику отсылки локального сертификата в процессе первой фазы IKE на запрос партнера. В своем запросе партнер может указать, какому CA сертификату он доверяет. Если такой сертификат не найден, то он не отсылается. Возможные значения:

- *AUTO* – автоматически определяется, когда необходима отсылка локального сертификата партнеру.
- *NEVER* – сертификат не высылается.
- *ALWAYS* – сертификат высылается всегда.
- *CHAIN* – сертификат высылается всегда, причем в составе с цепочкой доверительных CA. Имеется ввиду цепочка сертификатов, построенная от локального сертификата до CA, который удовлетворяет описанию, присланному партнером в запросе. В общем случае это CA, удовлетворяющий запросу партнера, произвольное количество промежуточных CA и локальный сертификат.

Значение по умолчанию – *AUTO*.

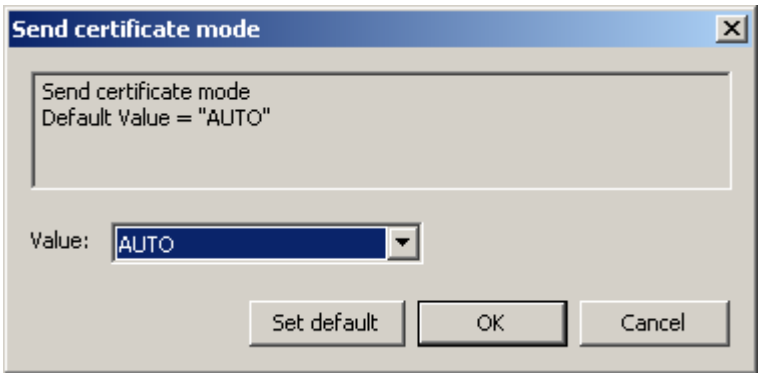

Рисунок 49

Do autopass

Задает режим автоматического пропускания ISAKMP-трафика. Возможные значения:

- *TRUE* – автоматически пропускать ISAKMP-пакеты по всем фильтрам, по которым защищается трафик.
- *FALSE* – не пропускать автоматически ISAKMP-пакеты. Правило фильтрации для пропускания ISAKMP-трафика должно быть задано явно (вручную) с действием PASS.

Значение по умолчанию – *TRUE*.

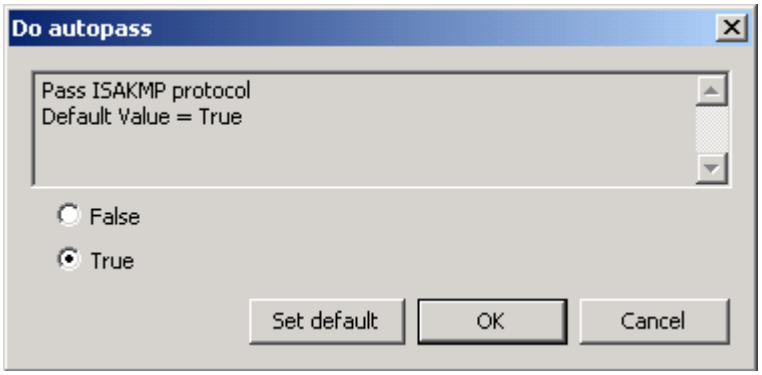

Рисунок 50

Prefer IKECFG address

Задает предпочитаемый запрашиваемый адрес по протоколу IKECFG при установлении соединения. Возможные значения:

- *Last used address* – запрашивается адрес, который был получен в предыдущий раз.
- *No preferred address* – предпочтений нет, запрашивается любой адрес.
- *Specific address* – запрашивается адрес, указанный в переменной *Specific IKECFG address.*

Значение по умолчанию – *Last used address*.

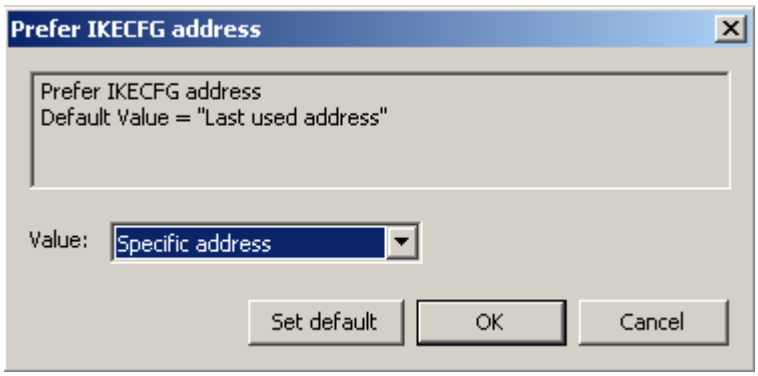

Рисунок 51

Specific IKECFG address

Задает адрес, который клиент предпочитает получить по протоколу IKECFG, если в переменной *Prefer IKECFG address* выбрано значение *Specific address*. Значение по умолчанию – 0.0.0.0, означает, что запрашивается произвольный адрес из пула.

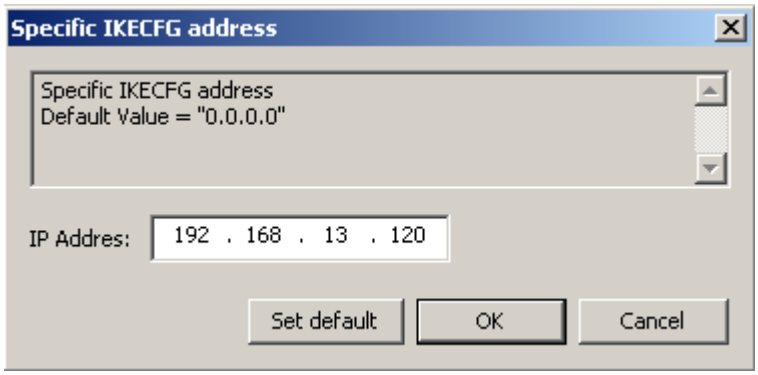

Рисунок 52

Firewall settings

SYN-SENT timeout

SYN-SENT timeout устанавливает период времени в секундах, в течение которого существует запись о TCP-соединении в состоянии SYN-SENT. Возможное значение – целое число из диапазона 1..65535. Значение по умолчанию – 30 секунд.

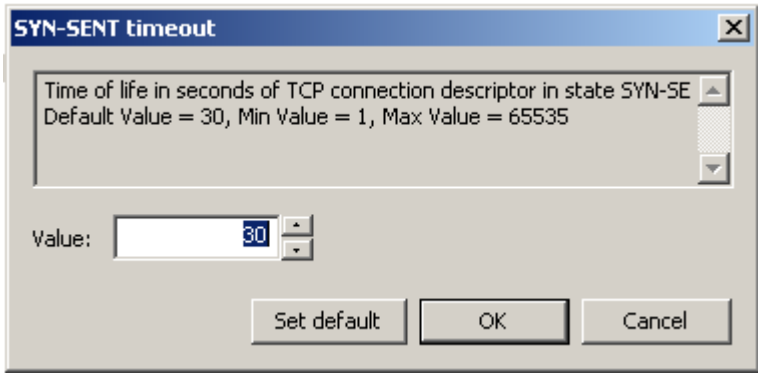

Рисунок 53

SYN-RECEIVED timeout

SYN-RECEIVED timeout устанавливает период времени в секундах, в течение которого существует запись о TCP-соединении в состоянии SYN-RECEIVED. Возможное значение – целое число из диапазона 1..65535. Значение по умолчанию – 30 секунд.

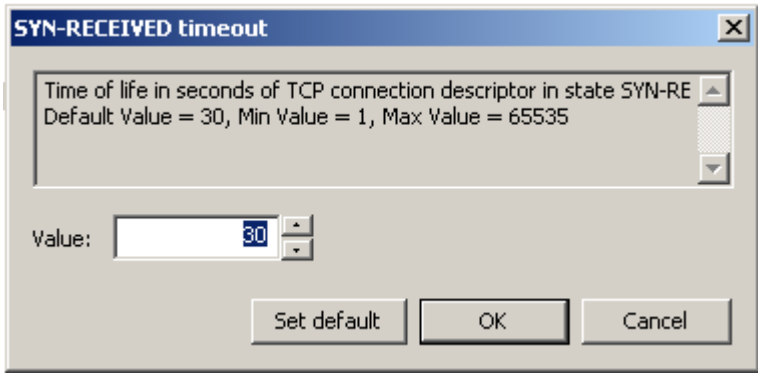

Рисунок 54

FIN timeout

FIN timeout устанавливает период времени в секундах, в течение которого существует запись о TCP-соединении в состоянии FIN-WAIT-1, CLOSE-WAIT, FIN-WAIT-2, LAST-ACK, TIME-WAIT или CLOSING. Возможное значение – целое число из диапазона 1..65535. Значение по умолчанию – 5 секунд.

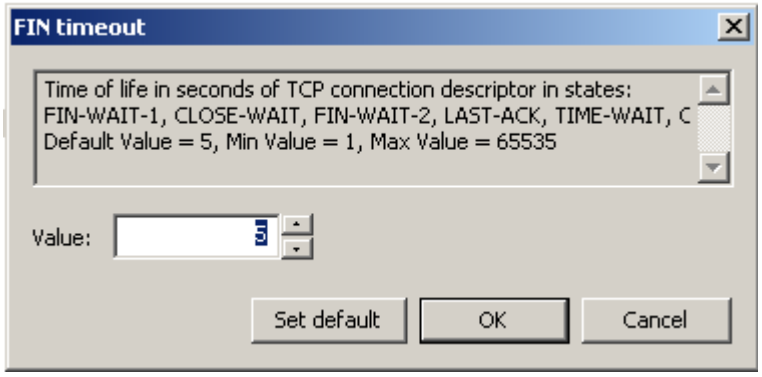

Рисунок 55

CLOSED timeout

CLOSED timeout устанавливает период времени в секундах, в течение которого существует запись о TCP-соединении в состоянии CLOSED или LISTEN. Возможное значение – целое число из диапазона 1..65535. Значение по умолчанию – 30 секунд.

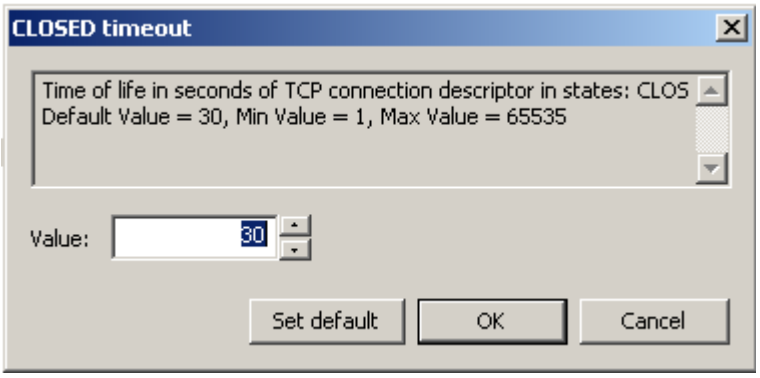

Рисунок 56

ESTABLISHED timeout

ESTABLISHED timeout устанавливает период времени в секундах, в течение которого существует запись о TCP-соединении в состоянии *ESTABLISHED*. Возможное значение – целое число из диапазона 1..65535. Значение по умолчанию – 3600 секунд.

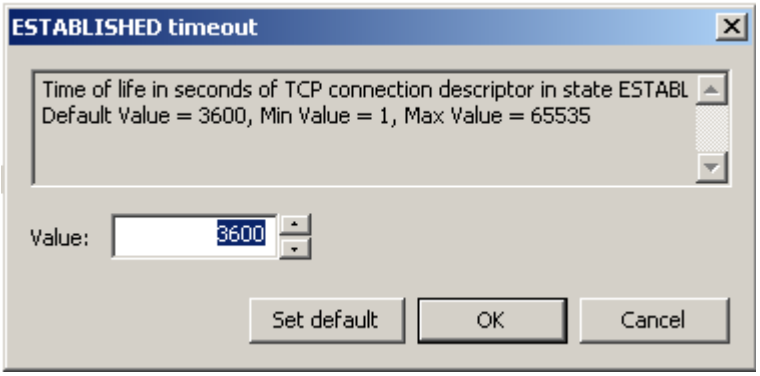

Рисунок 57

Maximum number of half open connections

Maximum number of half open connections устанавливает максимальное количество одновременно существующих полуоткрытых TCP-соединений (запросов соединения, оставшихся без ответов, или недостроенных соединений, не успевших перейти в установленное состояние). Когда количество полуоткрытых соединений превысит максимальное количество и как только появится новый запрос на соединение, одно полуоткрытое соединение будет удалено. Удаление будет происходить до тех пор, пока число полуоткрытых соединений не станет меньше значения *Lower bound of half open connections*. Далее вновь допускается увеличение TCP-соединений.

Возможное значение – целое число из диапазона 0..1000000. Значение по умолчанию – 500 полуоткрытых сеансов.

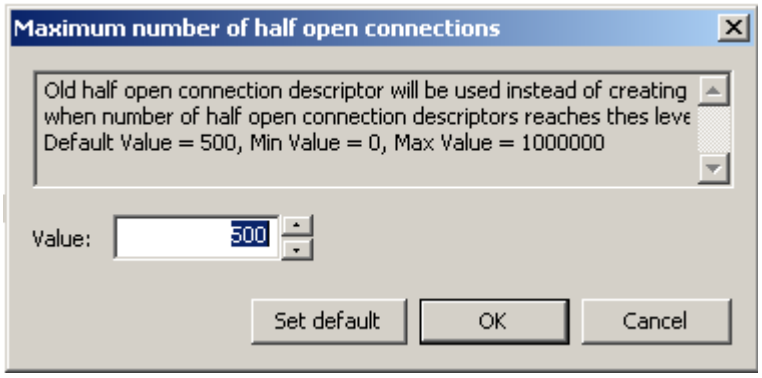

Рисунок 58

Lower bound of half open connections

Lower bound of half open connections устанавливает минимальное количество одновременно существующих полуоткрытых TCP-соединений (запросов соединения, оставшихся без ответов, или недостроенных соединений), по достижении которого прекращается их удаление. Возможное значение – целое число из диапазона 0..1000000. Значение по умолчанию – 400 полуоткрытых сеансов.

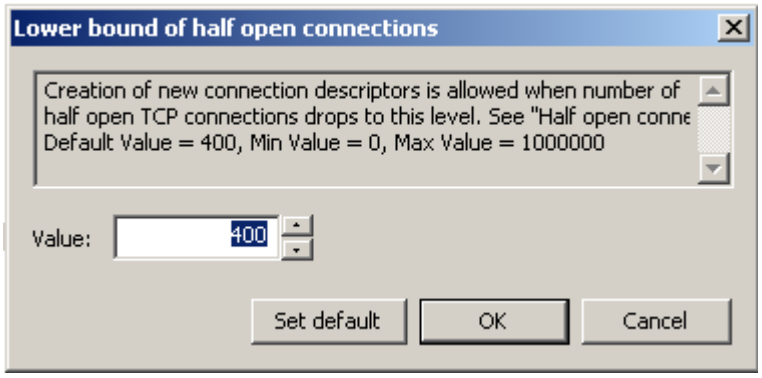

Рисунок 59

Maximum of new TCP connections rate

Maximum of new TCP connections rate устанавливает количество новых контекстов соединений, создаваемых в минуту, по достижении которого начинается их удаление, чтобы принимать новые запросы на соединение. Удаление будет продолжаться до тех пор, пока частота появления не будет совпадать с частотой, установленной переменной *Lower bound of TCP connections rate*. Возможное значение – целое число из диапазона 0..2147483647. Значение по умолчанию – 500 полуоткрытых сеансов в минуту.

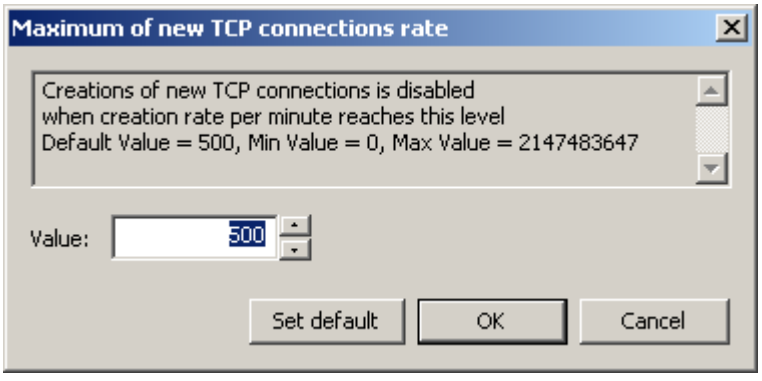

Рисунок 60

Lower bound of TCP connections rate

Lower bound of TCP connections rate устанавливает количество новых контекстов соединений, создаваемых в минуту, по достижении которого прекращается их удаление. Возможное значение – целое число из диапазона 0..2147483647. Значение по умолчанию – 400 полуоткрытых сеансов в минуту.

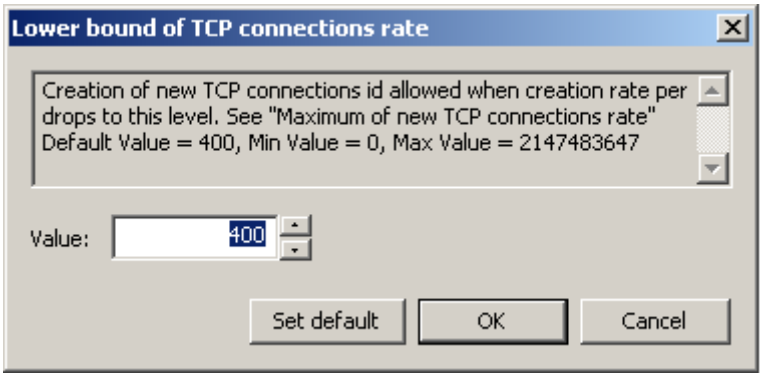

Рисунок 61

Maximum number of TCP connections

Maximum number of TCP connections устанавливает максимальное разрешенное количество TCP-соединений. При превышении данного предела новые TCP-соединения будут отвергаться. Возможное значение – целое число из диапазона 0..1000000. Значение по умолчанию – 65536.

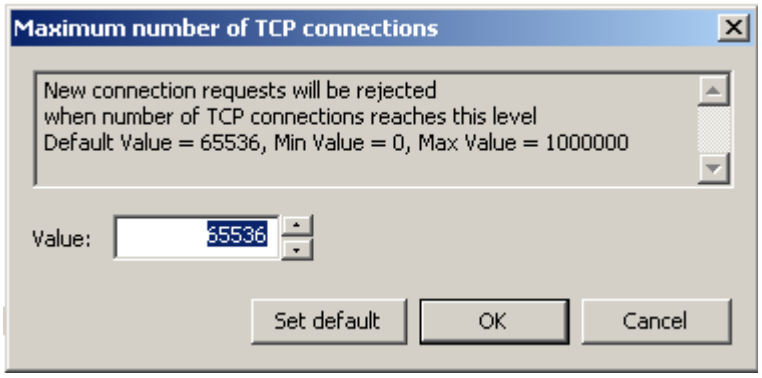

Рисунок 62

Strictness level

Strictness level задает уровень строгости обработки различных ошибочных ситуаций. В Таблица 5 приведены основные отличия в поведении при различных значениях *Strictness Level*. Возможное значение – целое число из диапазона 0..6. Значение по умолчанию – 3.

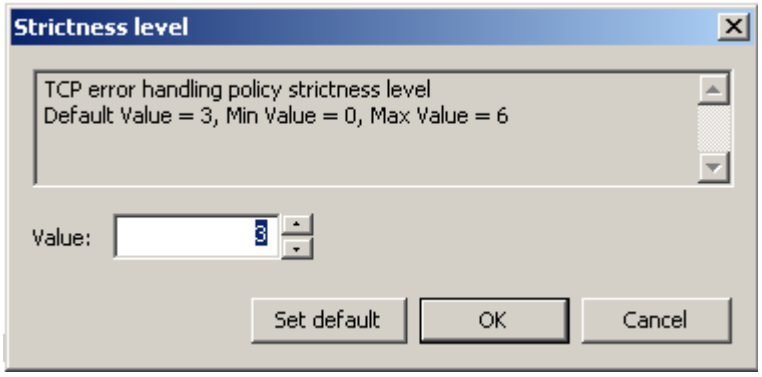

Рисунок 63

Таблица 5

Значение Strictness Level	Условие, при котором пакет уничтожается	Условие, при котором состояние соединения[4] не изменяется
0	Пакеты не уничтожаются firewall	При некорректном TCP заголовке (проверяется соответствие длины пакета, TCP заголовка, контрольной суммы)
1	При некорректном TCP заголовке	При некорректном TCP заголовке
2	При некорректном TCP заголовке	При некорректном TCP заголовке или при sequence, несоответствующем состоянию партнера
3	При некорректном TCP заголовке	При некорректном TCP заголовке или при несоответствии флагов заголовка состоянию партнера
4	При некорректном TCP заголовке или при несоответствии флагов заголовка состоянию партнера	При некорректном TCP заголовке или при несоответствии флагов заголовка состоянию партнера
5	При некорректном TCP заголовке или при sequence, несоответствующем состоянию партнера	При некорректном TCP заголовке или при несоответствии флагов заголовка состоянию партнера
6	При некорректном TCP заголовке или при приеме SYN для установившегося соединения	При некорректном TCP заголовке или при несоответствии флагов заголовка состоянию партнера, или при получении первым не SYN-пакета, или при приеме SYN-пакета для установившегося соединения.

LDAP settings

Server address

Задаваемые здесь параметры LDAP-сервера используются тогда, когда сертификат, для которого производится проверка подписи, не содержит расширение CDP (CRL Distribution Point) с адресом LDAP-сервера либо в этом поле прописанный путь к LDAP-серверу является неполным, и тогда добавляются данные из этой структуры.

В окне *Server address* задаются параметры LDAP-сервера. Возможные значения:

- LDAP-сервер не используется, когда флажок Use default LDAP Server не установлен.
- LDAP-сервер используется, когда флажок Use default LDAP Server установлен. При необходимости будет производиться поиск сертификатов и CRLs на заданном LDAP сервере. При этом нужно заполнить поля:
  - IP Address – сетевой адрес LDAP-сервера.
  - Port – сетевой порт LDAP-сервера, на который будут посылаться LDAP запросы. Значение по умолчанию – 389.

[4] Например, не пролонгируется существование записи о соединении.

- Search base – имя (Distinguished Name, DN) корневого X.500-объекта, в поддереве которого производится поиск сертификатов и CRL на LDAP-сервере. Если DN сертификата и DN X.500-объекта не совпадают, и если DN сертификата является частью имени DN X.500-объекта, то заполняется поле Searchbase, чтобы дополнить запрос, созданный на основе имени из сертификата или CRL, для нахождения соответствующего X.500-объекта. Для запроса на основе URL данное имя не используется. См. Пример в структуре LDAPSettings.

- Pass LDAP protocol with the LDAP Server – при установке этого флажка производится автоматическое создание сетевого фильтра для пропускания пакетов между агентом и LDAP-сервером.

Значение по умолчанию – LDAP-сервер не используется.

Сначала делается попытка установить соединение по LDAP версии 2. Если эта попытка завершается с ошибкой LDAP_PROTOCOL_ERROR (наиболее вероятная причина – не поддерживается версия 2), то повторяется попытка установить соединение по LDAP версии 3.

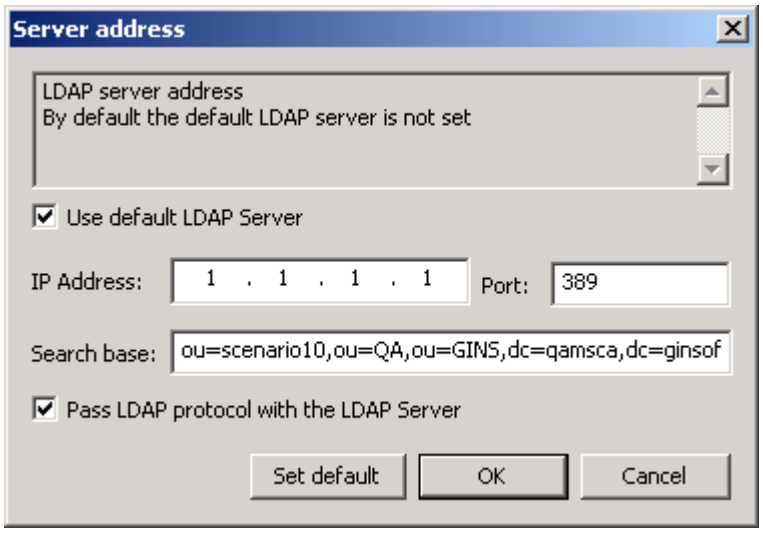

Рисунок 64

Connect timeout

Connect timeout позволяет ограничить время (в секундах) создания TCP-соединения с LDAP-сервером. Возможное значение – целое число из диапазона 1..6000. Значение по умолчанию – 0, которое означает, что время создания TCP-соединения с LDAP-сервером ограничивается установленным для ОС временем создания TCP-соединения.

Примечание: Если в момент обращения к LDAP-серверу устройство, на котором он установлен, недоступно, то процесс создания TCP-соединения может занимать продолжительное время (до 3 минут, зависит от ОС). По этой причине могут наблюдаться внешние признаки зависания агента, и это может служить причиной неудачной попытки создания соединения.

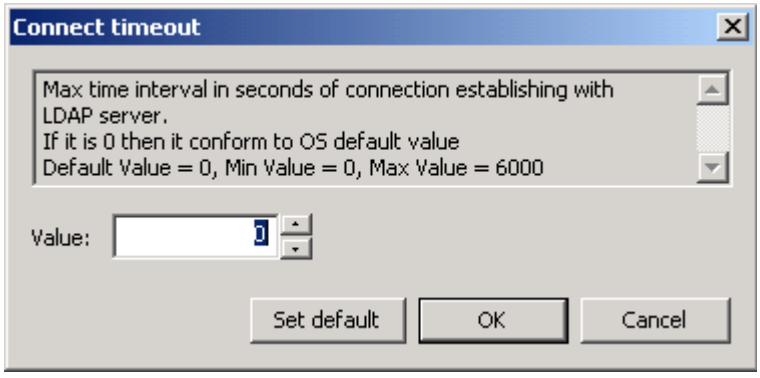

Рисунок 65

Response timeout

Поиск посредством протокола LDAP может занимать достаточно продолжительное время, оно зависит от многих факторов, в том числе от масштаба запроса и характеристик канала передачи данных. Response timeout позволяет ограничить время (в секундах), в течение которого ожидается ответ от LDAP-сервера на единичный запрос. Возможное значение – целое число из диапазона 2..6000. Значение по умолчанию – 200.

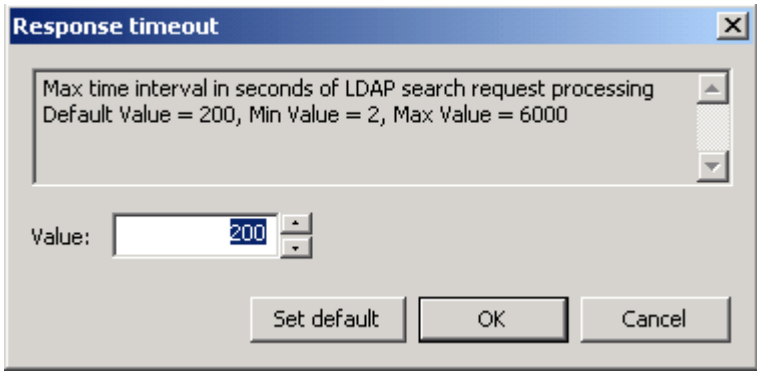

Рисунок 66

Hold connection timeout

Hold connection timeout устанавливает период времени, в течение которого держится установленное соединение к серверу на случай, если придет к нему повторный запрос. Возможное значение – целое число из диапазона 0..6000.

При значении 0 после обмена с LDAP-сервером соединение с ним сразу закрывается.

Не рекомендуется выставлять значение в 1 секунду в виду наличия погрешности в 1 секунду, поскольку это может привести в некоторых случаях к немедленному закрытию соединения и к избыточному открытию нового соединения.

Значение по умолчанию – 60.

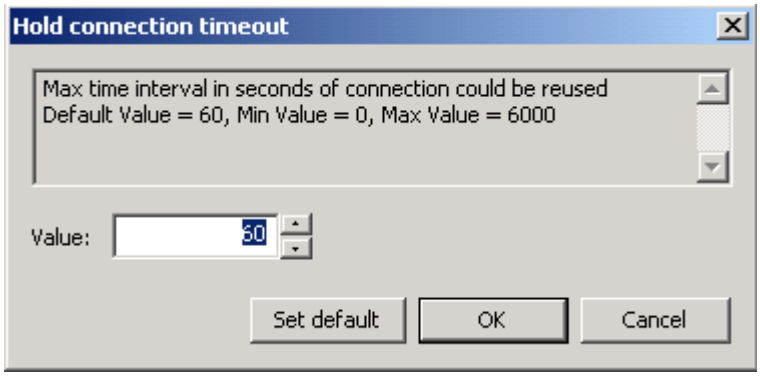

Рисунок 67

Drop connection timeout

Атрибут *Drop connection timeout* устанавливает период времени, начиная с первой неудачной попытки создания соединения с LDAP-сервером, в течение которого новые попытки создания соединения с ним игнорируются. Возможное значение – целое число из диапазона 0..6000.

При значении 0 в случае неудачной попытки установления соединения с LDAP-сервером новые попытки не игнорируются.

Не рекомендуется выставлять значение в 1 секунду в виду наличия погрешности в одну секунду, поскольку это может привести в некоторых случаях к избыточным попыткам создания соединения.

Значение по умолчанию – 5.

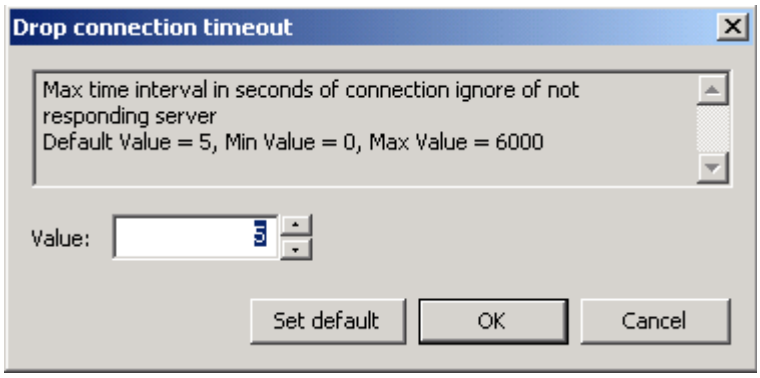

Рисунок 68

SNMP settings

SNMP polling

Задает настройки по выдаче информации по запросу SNMP-менеджера. Возможные значения:

- не принимаются и не обрабатываются запросы на выдачу SNMP статистики
- принимаются и обрабатываются запросы на выдачу SNMP статистики.

Значение по умолчанию – SNMP статистика не выдается.

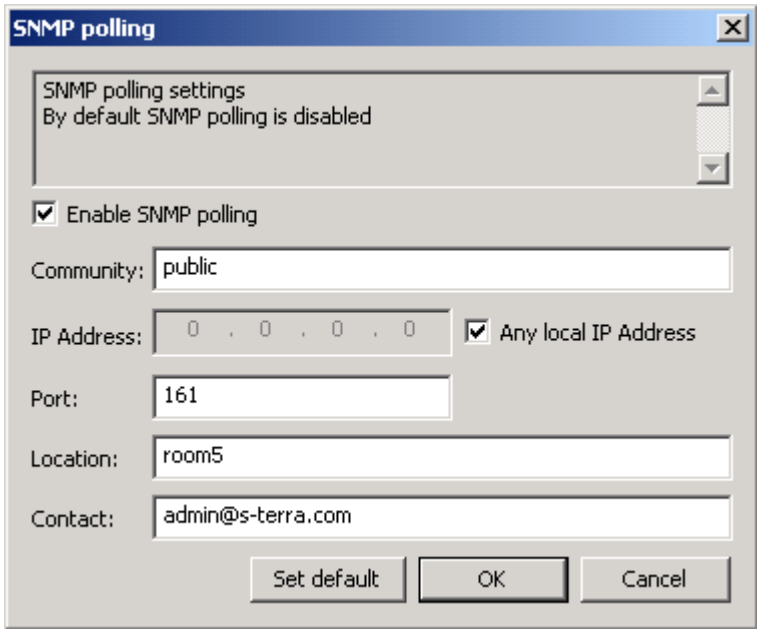

Рисунок 69

- *Enable SNMP polling* – при установке этого флажка задаются настройки для принятия запроса и выдачи статистики.
- *Community* – эта строка действует подобно паролю и разрешает доступ к чтению статистики SNMP-менеджеру.
- *IP Address* – локальный IPv4-адрес, на который можно получать запросы от SNMP-менеджера.
- *Any local IP Address* – установка этого флажка разрешает получение запроса от SNMP-менеджера на любой локальный IP-адрес.
- *Port* – задаёт порт, на который можно получать SNMP-запросы.

- *Location* – информация о физическом расположении SNMP-агента.
- *Contact* – информация о контактном лице, ответственном за работу SNMP-агента.

Trap receiver

Задает настройки получателя SNMP-трапов и дополнительные настройки для трапов, отсылаемых на него. Возможные значения:

- получатель SNMP-трапов не задан,
- получатель SNMP-трапов задан.

Значение по умолчанию – получатель SNMP-трапов не задан.

Можно задать до трех получателей SNMP-трапов.

- *Enable the trap receiver* – при установке этого флажка задаются настройки получателя SNMP-трапов.
- *Community* – текстовая строка, играющая роль идентификатора отправителя трап-сообщения.
- *Receiver's IP Address* – IP-адрес получателя SNMP-трапов.
- *Receiver's Port* – UDP-порт, на который менеджеру будут высылаться трап-сообщения.
- *SNMP Version* – версия SNMP, в которой формируются трап-сообщения.
- *Agent's IP Address* – IP-адрес отправителя трап-сообщения. Этот атрибут указывается только для Version = V1.

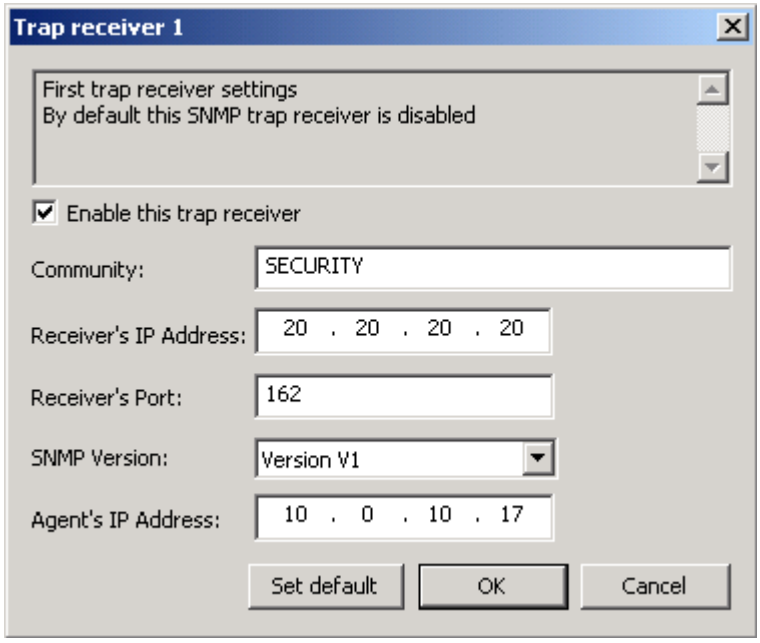

Рисунок 70

DPD settings

Use DPD

Задает режим использования протокола DPD (Dead-Peer-Detection). Возможные значения:

- *TRUE* – использовать протокол DPD.

- *FALSE* – не использовать протокол DPD. В этом случае другие переменные этого раздела не появляются.

Значение по умолчанию – *TRUE.*

Окно для установки переключателя:

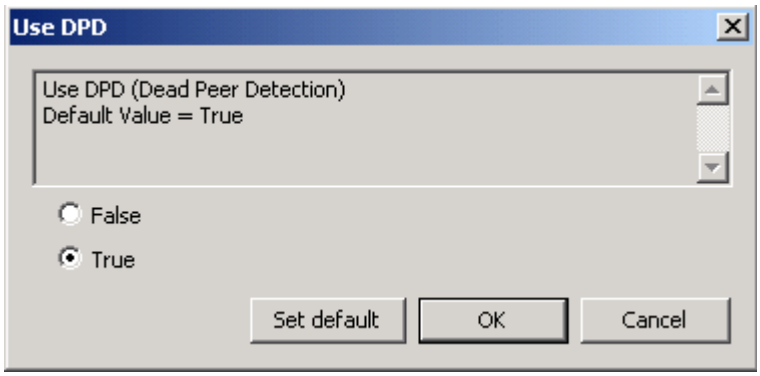

Рисунок 71

Idle duration (seconds)

Интервал времени отсутствия входящего трафика от партнера, по истечению которого, при наличии исходящего трафика, активируется DPD-сессия. Возможные значения – целое число из диапазона 1..32762. Значение по умолчанию – 60. Окно для установки значения:

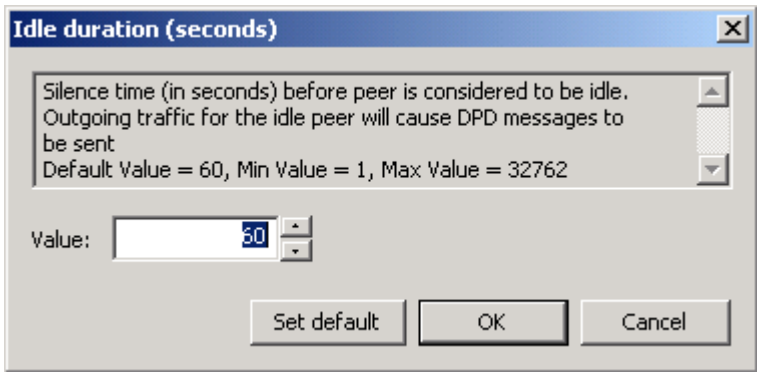

Рисунок 72

Response duration

Время ожидания ответа от партнера на DPD-запрос в секундах. Возможные значения – целое число из диапазона 1..300. Значение по умолчанию – 5. Окно для выбора значения:

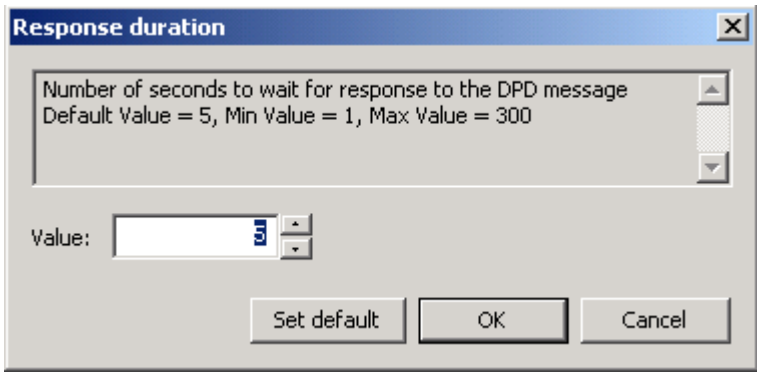

Рисунок 73

Retries

Количество попыток провести DPD-обмен. Если все попытки закончились неудачей, защищенное соединение (IKE-контекст) считается "мертвым", и производится попытка создать его заново. Возможные значения – целое число из диапазона 1..10. Значение по умолчанию – 3. Окно для выбора значения:

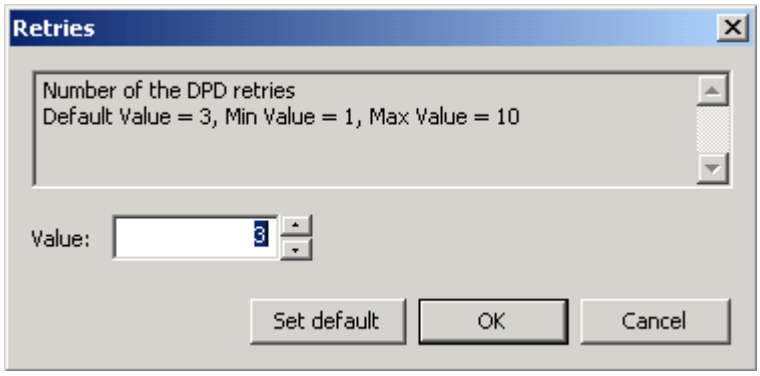

Рисунок 74

7.3.7.2 Режим ручного задания LSP

В режиме ручного задания локальная политика безопасности задается администратором (вкладки **Firewall Rules**, **IPsec Rules**, **IKE** и **IPsec** становятся невидимыми) (Рисунок 75).

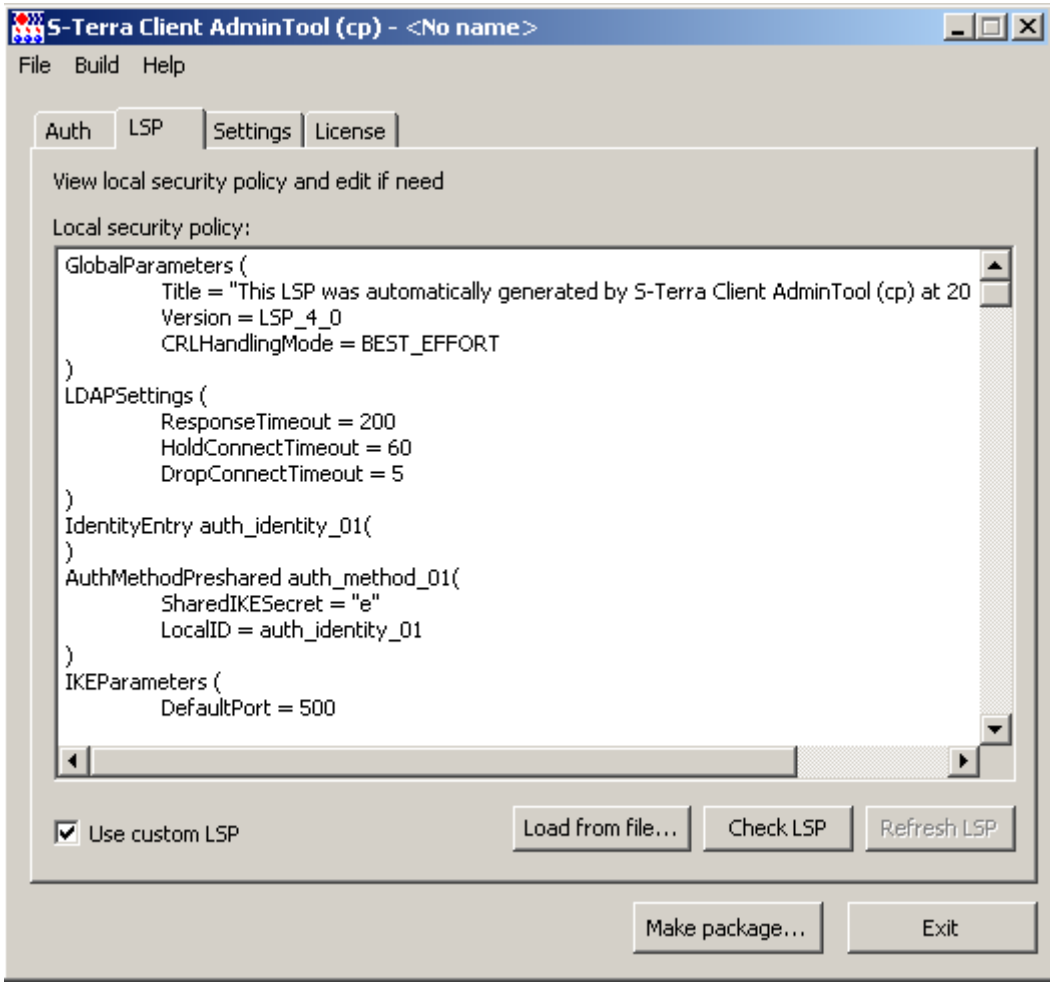

Рисунок 75

Local security policy – поле с текстовым представлением локальной политики безопасности. В этом поле можно создавать и редактировать LSP.

Use custom LSP – снятие этого флажка переводит в режим автоматического формирования LSP.

Load from file – при нажатии этой кнопки происходит загрузка LSP из файла и отображение в поле Local security policy.

Check LSP – при нажатии этой кнопки происходит проверка заданной LSP по выявлению синтаксических ошибок. При обнаружении ошибки выдается сообщение с описанием ошибки (если строка с ошибочными символами определена, то она выделяется и на эту строку автоматически переводится фокус). Если данная LSP не содержит синтаксических ошибок, то выдается сообщение, что синтаксических ошибок не найдено.

7.3.8 Вкладка Settings

Во вкладке **Settings** задаются настройки протоколирования событий, политика по умолчанию и дополнительные параметры инсталляции Продукта S-Terra Client.

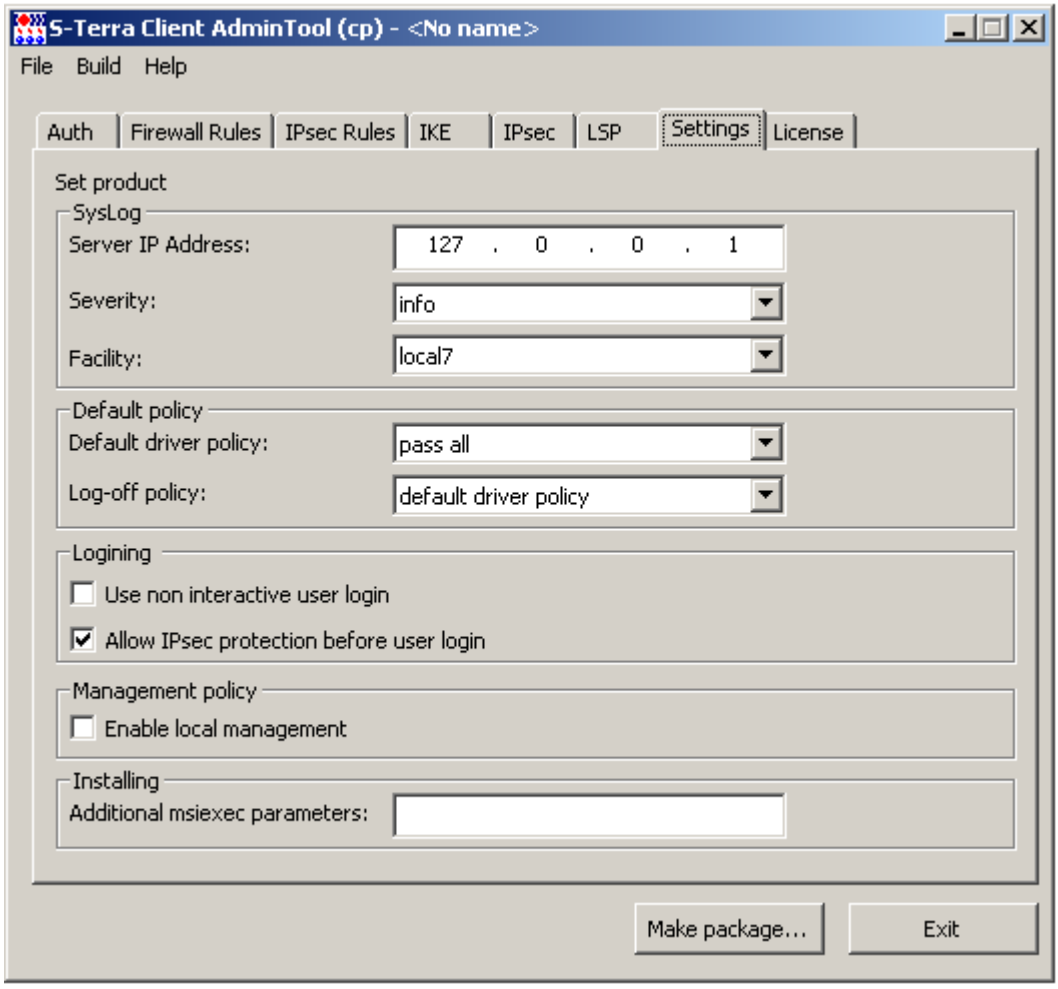

Рисунок 76

Для задания настроек Syslog-клиента заполняются следующие поля:

- **Server IP-Address** – IP-адрес компьютера, на который будут посылаться сообщения о протоколируемых событиях. Значение по умолчанию – *127.0.0.1* означает, что сообщения посылаются на локальный хост.
- **Severity** – задание общего уровня протоколирования. Содержит выпадающий список значений – *emerg*, *alert*, *crit*, *err*, *warning*, *notice*, *info*, *debug*. Значение по умолчанию – *info*.
- **Facility** – задание источника сообщений. Значение по умолчанию – *local7*.

Default Driver Policy (DDP) – политика драйвера по умолчанию. Выпадающий список содержит значения:

- *pass all* – пропускать все пакеты. Значение по умолчанию.
- *pass dhcp* – пропускать пакеты только по протоколу DHCP. Т.е. будут уничтожаться все пакеты, кроме исходящих UDP-пакетов на порт 67 и входящих UDP-пакетов на порт 68.
- *drop all* –не пропускать трафик (для релиза 14101 это значение недоступно).

Политика DDP, которая задается администратором, загружается в следующих случаях:

- при ошибке загрузки конфигурации,
- до старта VPN Service,
- при остановке VPN Service.

Log-off policy – специальная политика безопасности, которая задается администратором при подготовке инсталляционного пакета, и служит для безопасности работы пользователя, при которой клиент не может создавать защищенных соединений. Эта политика работает по одному из двух правил:

- *default driver policy (DDP)* – политика драйвера по умолчанию,
- *pass dhcp* – пропускать пакеты только по протоколу DHCP. Будут уничтожаться все пакеты, кроме исходящих UDP-пакетов на порт 67 и входящих UDP-пакетов на порт 68.

Политика Log-off policy загружается автоматически в следующих случаях:

- до тех пор, пока пользователь не ввел свой пароль,
- при вводе неверного пароля три раза,
- при отказе от регистрации (login), если нажать кнопку *Cancel,*
- при выходе пользователя из системы,
- при смене пользователя,
- если при загрузке конфигурации обнаружены ошибки (если была ранее загружена Log-off policy).

Use non interactive user login – при установке этого флажка S-Terra Client будет использовать неинтерактивный режим логина, а при снятии – интерактивный режим логина:

- Неинтерактивный режим – при входе пользователя в систему производится попытка логина в продукт S-Terra Client с пустым паролем (в качестве пароля используется пустая строка). При таком успешном логине окно с запросом пароля не выводится. При неуспешном логине – Продукт ведет себя как при интерактивном режиме.
- Интерактивный режим – выдается окно запроса пароля для регистрации в Продукте S-Terra Client. Этот режим используется по умолчанию.

В случае неинтерактивного логина Log-off policy при старте не загружается.

Allow IPsec protecrion before user login – установка флажка включает функциональность по защите до логина в ОС. По умолчанию защита включена. (Включить и отключить эту опцию можно и для установленного Продукта. Для этого в "Установка и удаление программ" для Продукта выбрать "Изменить". Далее "Modify". И затем установить соответствующую функцию для "Login Protection".)

Enable local management – при установке этого флажка включается возможность изменять настройки Продукта конечным пользователем. По умолчанию эта возможность отключена. Пользователь может менять только пароль, уровень логирования, добавлять CRL и сертификаты партнеров, перезагружать локальную политику безопасности.

Additional msiexec parameters – в этом поле можно установить дополнительные параметры запуска WinInstaller.

Например, альтернативный каталог, в который будет установлен Продукт, настройки лога Windows Installer и т.п. Эти параметры можно посмотреть по ссылке http://msdn.microsoft.com/library/default.asp?url=/library/en-us/msi/setup/command_line_options.asp

`/l* C:\Client\install_log_file.txt` – протоколирование событий в файл C:\Client\install_log_file.txt при инсталляции S-Terra Client (рекомендуется при режиме silent).

`INSTALLDIR=каталог установки продукта` – переопределение каталога. Указание каталога, недоступного на компьютере пользователя, приведет к ошибке инсталляции.

Можно часть текста заключить в символы % (процент) и она будет рассматриваться как имя переменной окружения. И эта часть текста будет заменяться на значение переменной окружения (значения переменных окружения можно просмотреть командой set). Если переменная окружения отсутствует, в командке остается исходный текст.

Поддерживается специальная переменная окружения SfxDir – полный путь к папке, в которую распакованы данные. Таким образом, последовательность символов %SfxDir% заменяется на полный путь к папке, в которую распакованы данные.

`REBOOT=F` – обязательно запрашивать перезагрузку системы в конце инсталляции, даже если он не инициируется инсталлятором.

REBOOT=S – отключить запрос на перезагрузку системы в конце инсталляции. Не блокировать рестарт в случае ForceReboot action.

REBOOT=R – полностью отключить все запросы на перезагрузку системы, включая ForceReboot action. Используется для установки нескольких продуктов и/или выполнения дополнительных действий после инсталляции. После этого перезапустить систему вручную или с помощью сторонних инструментальных средств.

MAX_SERVICE_START_TIMEOUT= … – время (в секундах) ожидания старта VPN сервиса (vpnsvc). Максимальное значение – 600 секунд. Значение по умолчанию – 30. Можно использовать для предотвращения появления сообщений об ошибке связи с сервисом на этапе логина для медленных и/или находящихся под сильной нагрузкой систем.

AGENT_DB_REMOVE=1 – автоматически (без дополнительных запросов) будет удаляться база локальных настроек при установке или при удалении продукта. Рекомендуется использовать для режима инсталляции silent.

AGENT_DB_REMOVE=0 – база локальных настроек удаляться не будет, запросы пользователю выдаваться не будут. По умолчанию (параметр пустой) – пользователю выдается запрос на удаление базы локальных настроек.

DISABLE_ANTIVIRUS_WARNING=1 – при инсталляции не будет показываться предупреждение 25036 (о необходимости отключения антивирусных программ).

> **Примечание:** пользователь должен знать о необходимости отключения антивируса, иначе данный параметр использовать не следует.

REBOOT_REQUIRED=1 – принудительно инициировать запрос на рестарт системы в конце инсталляции. Параметр обычно выставляется автоматически (при необходимости).

DISABLE_CALL_LOGIN=1 – в конце инсталляции логин не запустится. Устанавливать параметр имеет смысл только для интерактивного логина (NON_INTERACTIVE_LOGIN=0) на Windows Vista и более поздних версиях.

7.3.9 Вкладка License

Во вкладке **License** задаются регистрационные данные Лицензии на Продукт S-Terra Client:

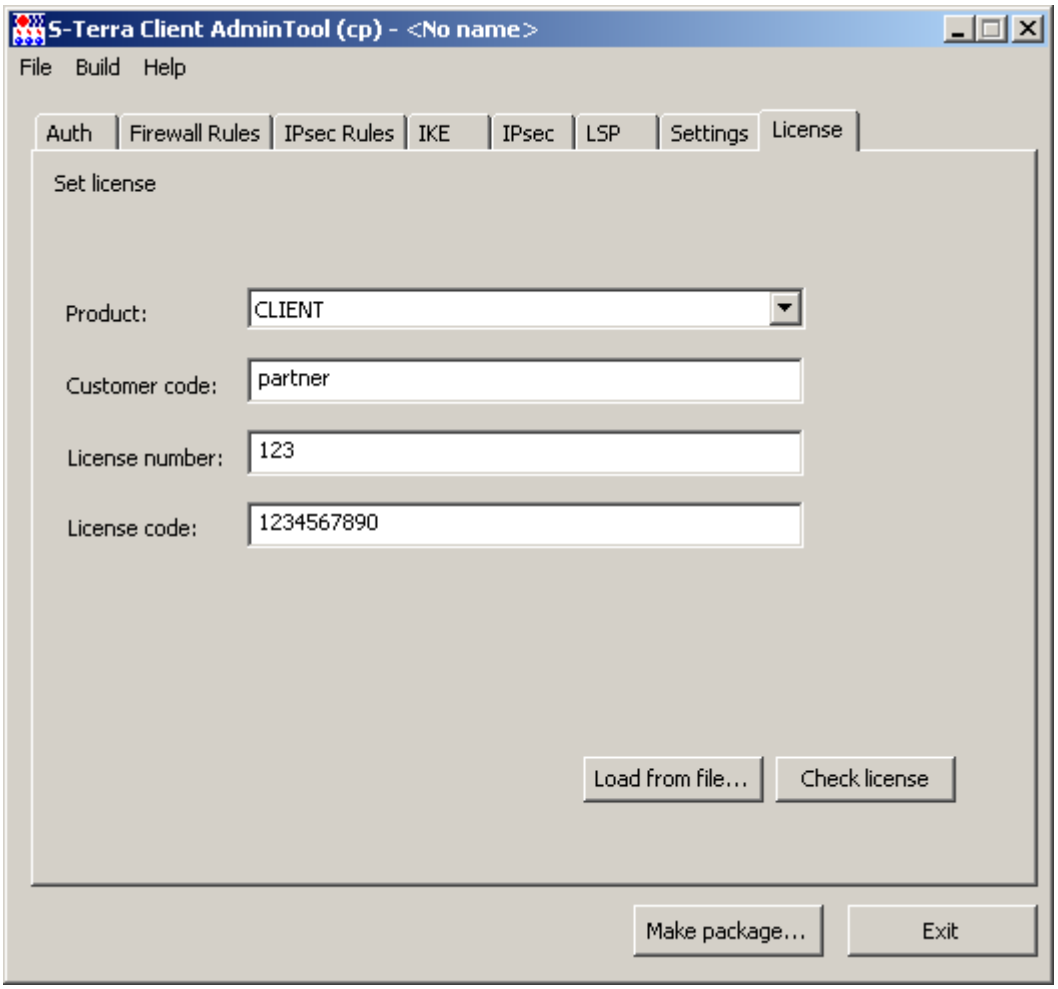

Рисунок 77

Данные Лицензии:

- **Product** – поле для задания типа Продукта.
- **Customer code** – код пользователя.
- **License number** – номер лицензии.
- **License code** – код лицензии.

Кнопки управления:

- **Load from file** – при нажатии этой кнопки происходит загрузка данных Лицензии из указанного текстового файла. В файле данные Лицензии должны быть записаны в виде:

```
[license]
CustomerCode=NNNN
ProductCode=CLIENTB/CLIENT
LicenseNumber=NNNN
LicenseCode=NNNNNNN
```

- **Check License** – проверка правильности введенных данных Лицензии.

7.3.10 Задание сертификатов партнеров

Сертификат партнера можно получить либо по протоколу IKE, либо по протоколу LDAP при создании IKE соединения.

Сначала шлюз безопасности пытается получить сертификат партнера по IKE. Если партнер не прислал сертификат, а прислал свой идентификатор, то шлюз безопасности по этому идентификатору ищет сертификат партнера сначала в своей базе Продукта, если не нашел, то продолжает поиск на LDAP-сервере.

Но не всегда удается получить сертификаты от удаленных партнеров: могут быть проблемы, связанные с фрагментацией UDP пакетов.

Поэтому появилась возможность положить в базу Продукта S-Terra Client имеющиеся сертификаты партнеров.

В меню GUI выберите раздел **File**, а затем предложение ***Advanced Project Settings***. В одноименном окне (Рисунок 78) с одной вкладкой *Partner certificates* создайте список сертификатов ваших партнеров. Сертификаты партнеров будут актуальны только при аутентификации с использованием сертификатов.

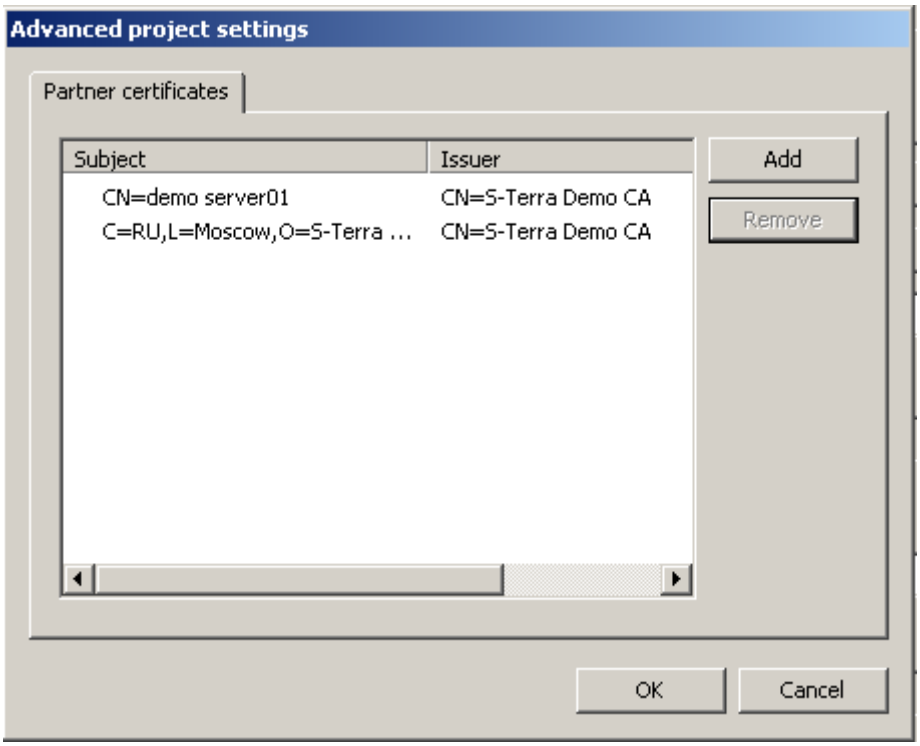

Рисунок 78

Кнопки управления:

- **Add** – добавляет сертификат партнера в список, при этом появляется стандартное окно открытия файла.
- **Remove** – удаляет выделенный сертификат партнера из списка.

Если добавление сертификата происходит из контейнера формата PKCS#12 (.pfx), в котором размещено более одного сертификата, появляется окно для выбора сертификата для добавления в список (Рисунок 79):

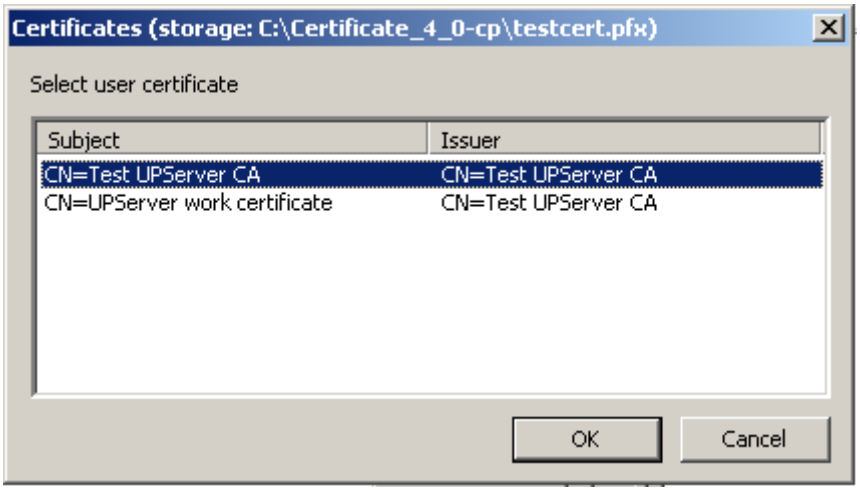

Рисунок 79

Добавление сертификата в список может быть неуспешным с выводом соответствующих сообщений по следующим причинам:

- прочитанные данные не являются корректным сертификатом: «Certificate storage <путь к файлу> is incorrect»;
- список уже содержит такой сертификат, при этом пути к файлам могут быть разными (сравниваются сами сертификаты): «This certificate already in list Subject: <subject сертификата> Issuer: <issuer сертификата>».

В случае, когда некорректные данные прочитаны из файла проекта (созданного в S-Terra Client Admintool версии 3.0 или 3.1 и если были отредактированы вручную ссылки на сертификаты), выполняется проверка дублирования сертификатов в списке, отображаемых в окне (Рисунок 80). Напротив проблемной строки отображается восклицательный знак красного цвета, таким образом помечаются второй и последующие одинаковые сертификаты.

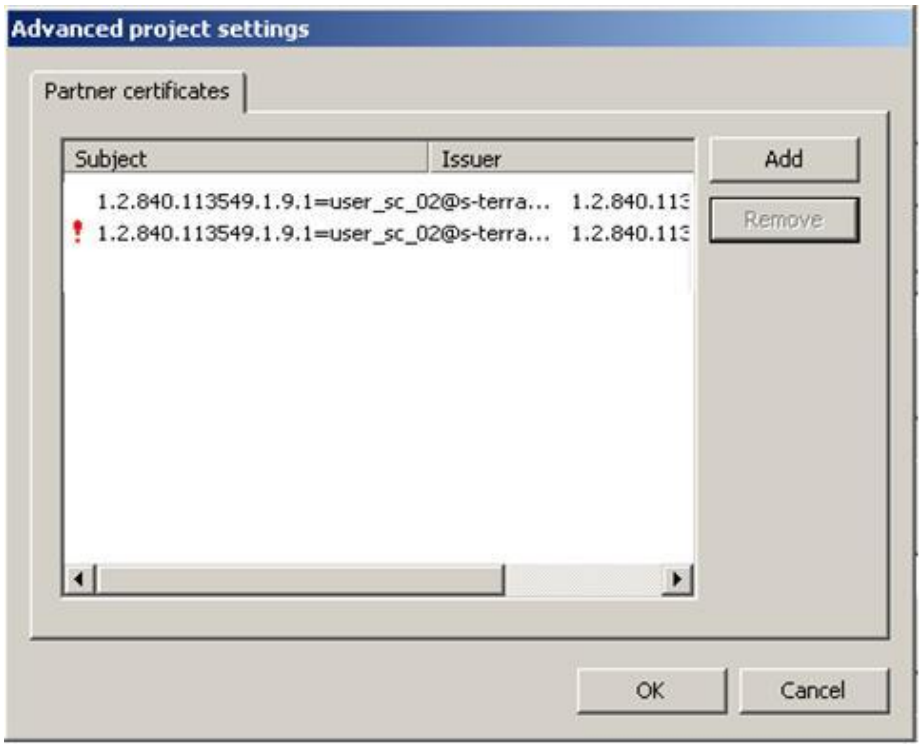

Рисунок 80

7.3.11 Создание инсталляционного файла

Создание инсталляционного файла S-Terra Client происходит при нажатии кнопки **Make package** в главной форме. При этом происходит проверка корректности введенных данных и при обнаружении ошибки выводится сообщение о возможных причинах, и переключение на вкладку с некорректными данными. Если ошибки не обнаружено, то появляется окно **Package parameters** (Рисунок 81):

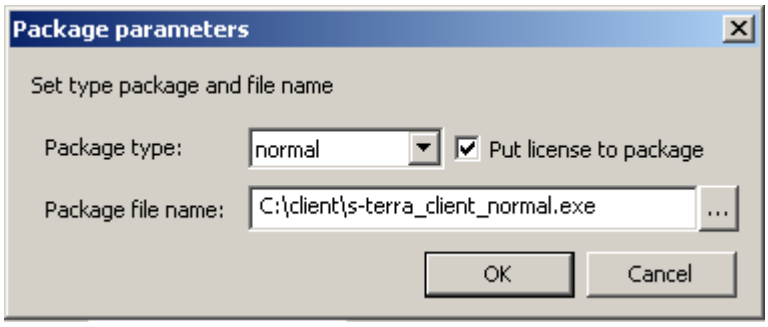

Рисунок 81

В этом окне необходимо задать:

- **Package type** – поле для выбора режима инсталляции. Возможные значения:
  - *basic* – неинтерактивная установка с запросом на инсталляцию. Вариант по умолчанию,
  - *normal* – интерактивная установка (в диалоговом режиме) с демонстрацией Лицензионного Соглашения и другими окнами,
  - *silent* – неинтерактивная установка без запросов.
- **Package file name** – поле для ввода имени инсталляционного файла на компьютере администратора.
- **Put license to package** – при установке этого флажка введенные данные Лицензии будут включены в инсталляционный файл. При этом вкладка **License** должна содержать корректные данные Лицензии.

При нажатии кнопки **ОК** вызывается утилита `make_inst.exe` с соответствующими опциями, которая и создает инсталляционный файл. На время работы утилиты появляется окно с просьбой подождать (Рисунок 82):

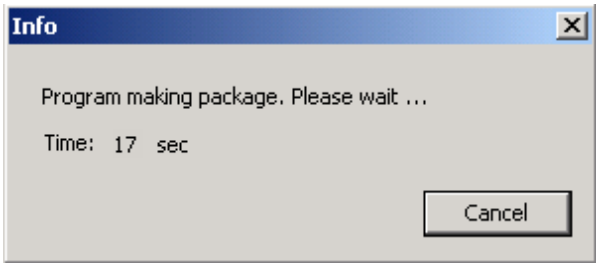

Рисунок 82

В случае выявления ошибки выдается сообщение о коде ошибки.

При нажатии на кнопку **Cancel** работа утилиты `make_inst.exe` прерывается (инсталляционный файл не создается). В случае успешного завершения работы утилиты выдается сообщение о создании инсталляционного файла:

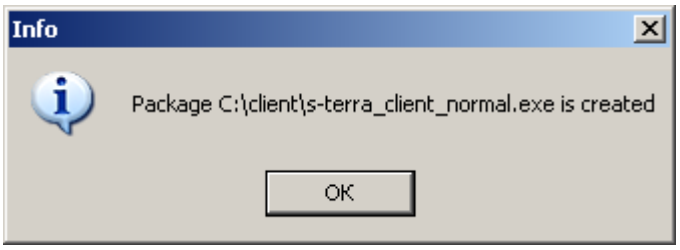

Рисунок 83

Все сообщения, выдаваемые программой утилитой `make_inst` в процессе ее работы, выводятся в файл `make_inst_log.txt` (при каждом создании инсталляционного файла `make_inst_log.txt` переписывается).

7.3.12 Сохранение данных проекта

В процессе сохранения проекта – `Save Project as` (`Save Project`) – сохраняются данные тех вкладок, которые на данный момент являются активными. Данные вкладок, которые являются невидимыми, не сохраняются. Исключение составляет ситуация: при переключении на ручное задание LSP (вкладка **LSP**, выставлен флажок *Use custom LSP*) данные вкладок **Firewall Rules**, **IPsec Rules**, **IKE** и **IPsec** сохраняются в проекте, не смотря на то, что после переключения эти вкладки являются неактивными и не показываются пользователю. При повторном открытии сохраненного проекта, при переходе к режиму автоматического формирования LSP (снятие флажка *Use custom LSP*), все введенные ранее пользователем данные во вкладках **Firewall Rules**, **IPsec Rules**, **IKE** и **IPsec** будут доступны для дальнейшего редактирования. Данная особенность реализована для облегчения редактирования LSP при ее автоматическом формировании.

7.4 GUI. Режим пользовательского токена

Режим пользовательского токена в GUI – это режим, в котором можно создать пользовательский токен. Подготовленный Продукт S-Terra Client в таком режиме будет работать только при наличии пользовательского токена, подключенного к компьютеру пользователя перед началом работы и требующего введения PIN-кода. В таком режиме Продукт может работать с пользовательскими токенами разных пользователей.

На пользовательском токене должен лежать CA сертификат, сертификат пользователя, контейнер с секретным ключом и локальная политика безопасности, предписанная данному пользователю. При отсутствии на токене любого из этих объектов Продукт работать не будет. Кроме того, Продукт не сохраняет объекты с токена в базе Продукта, как и PIN-код к токену.

В этом режиме Продукт может использовать из базы Продукта только партнерские сертификаты и список отозванных сертификатов.

Для подготовки пользовательского токена:

На компьютер с установленным административным пакетом S-Terra Client AdminTool и СКЗИ «КриптоПро CSP 3.6/3.6R2/3.6R4» установите набор драйверов и утилит для работы с токеном. Например, для eToken PRO, eToken NG-OTP, eToken NG-FLASH, eToken PRO 72K (Java) установите пакет «eToken PKI Client 5.1 SP1 для Microsoft Windows», который можно взять с web-страницы по адресу: http://www.aladdin-rd.ru/support/downloads/etoken/.

Подключите токен к этому компьютеру и введите PIN-код.

В S-Terra Client AdminTool перейдите в режим пользовательского токена, выбрав в меню предложение **Build – Token pattern** (Рисунок 84).

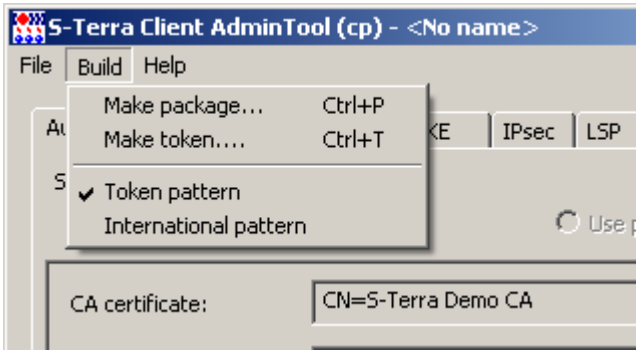

Рисунок 84

В режиме пользовательского токена изменятся только две вкладки – **Auth** (Рисунок 85) и **Settings** (Рисунок 97) GUI, остальные вкладки останутся без изменений.

7.4.1 Вкладка Auth

В режиме пользовательского токена аутентификация осуществляется только с использованием сертификатов, поэтому на вкладке **Auth** переключатель может стоять только в положении *Use certificate* (Рисунок 85).

Во вкладке **Auth** доступны для заполнения следующие поля:

CA certificate – здесь отражается поле Subject корневого сертификата Удостоверяющего Центра (Trusted CA Certificate). Для этого разместите на компьютере администратора файл с Trusted CA сертификатом и в конце поля нажмите кнопку `[...]`, в открывшемся окне выберите файл с CA сертификатом. Обязательный параметр.

Token reader name – имя считывателя для токена. Нажмите кнопку `[...]` в конце поля и в открывшемся окне **Token reader list** (Рисунок 86) выберите считыватель. *Обязательный параметр*.

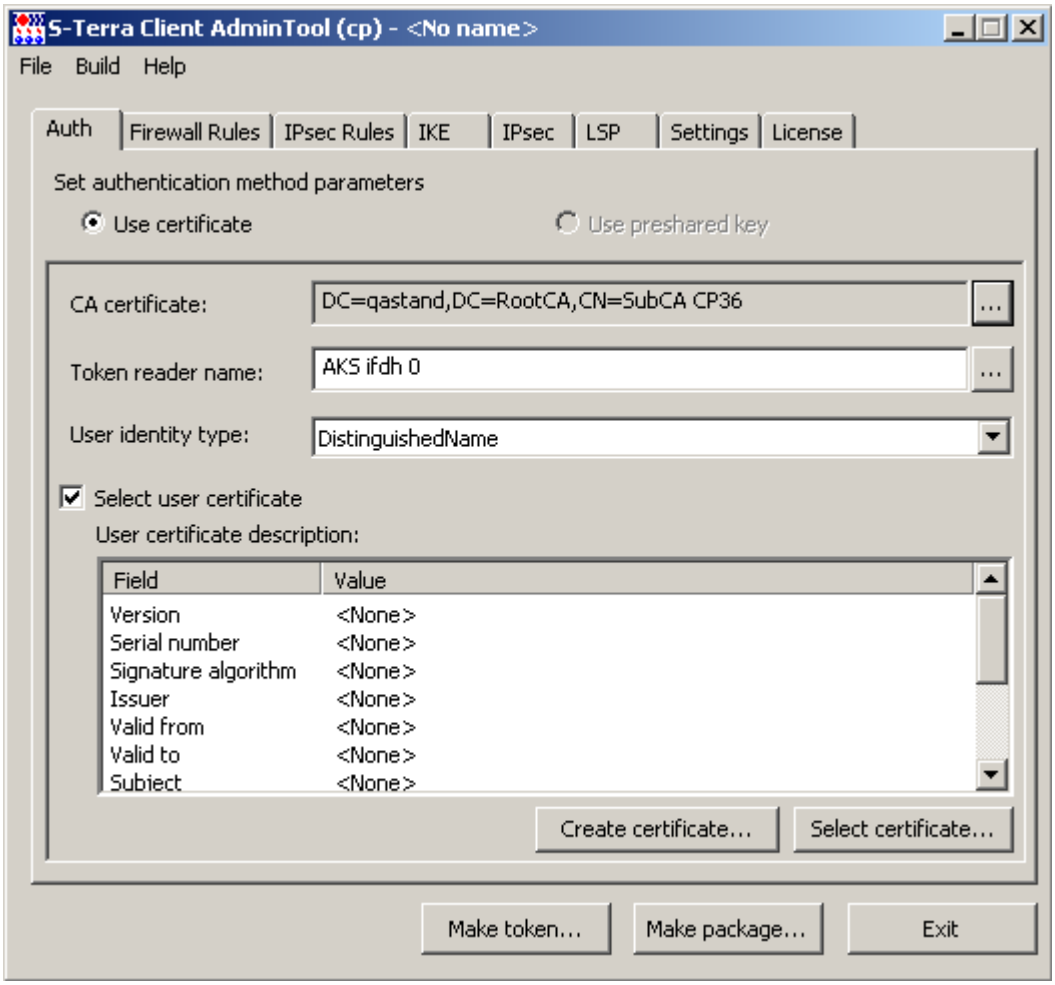

Рисунок 85

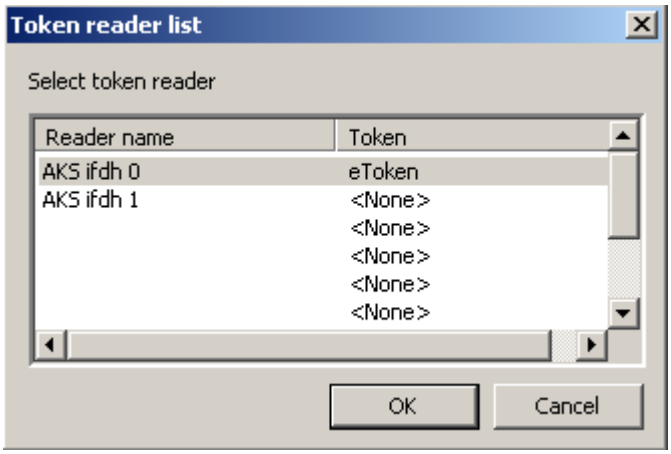

Рисунок 86

User identity type – тип идентификационной информации, пересылаемой партнеру при создании защищенного соединения. Обязательный параметр. Поле содержит выпадающий список со следующими значениями:

- *Distinguished Name* – в качестве идентификатора партнеру будет высылаться значение Subject из сертификата пользователя. Значение по умолчанию.
- *Email* – в качестве идентификатора партнеру будет высылаться значение поля E-mail расширения сертификата пользователя.

- *FQDN* – в качестве идентификатора партнеру будет высылаться значение доменного имени хоста, считываемое из поля DNS расширения сертификата.
- *IPV4Addr* – в качестве идентификатора партнеру будет высылаться первый IP-адрес, указанный в расширении сертификата.
- *Local IP address* – в качестве идентификатора партнеру будет высылаться действительный IP-адрес хоста, на котором будет установлен S-Terra Client.

Select user certificate – при установке этого флажка нужно выбрать сертификат пользователя, расположенный на токене. Если на токене сертификата нет, то его нужно создать. Если флажок не устанавливать, то при создании соединения будет использоваться первый попавшийся сертификат пользователя, размещенный на токене.

User certificate description – в этом поле отображаются поля выбранного сертификата пользователя.

Кнопки управления:

Create certificate – кнопка для вызова окна визарда, который помогает создать ключевую пару и запрос на сертификат пользователя, а также записать сертификат на токен (Рисунок 87).

Select certificate – кнопка для вызова окна со списком пользовательских сертификатов, размещенных на токене, из которых нужно выбрать сертификат.

Создание сертификата

При отсутствии сертификата пользователя на токене нажмите кнопку **Create certificate**. В открывшемся окне **Cerificate creation wizard** (Рисунок 87) имеется переключатель с двумя положениями:

- *Step 1* – создать ключевую пару, записав ее в контейнер на токене, и запрос на локальный сертификат.
- *Step 2* – записать сертификат пользователя в контейнер с ключевой парой на токене.

Если на токене записан контейнер с ключевой парой и имеется сертификат пользователя, то запишите его в контейнер, перейдя к разделу «Запись сертификата на токен».

Если сертификат пользователя отсутствует, то выберите положение *Step 1*.

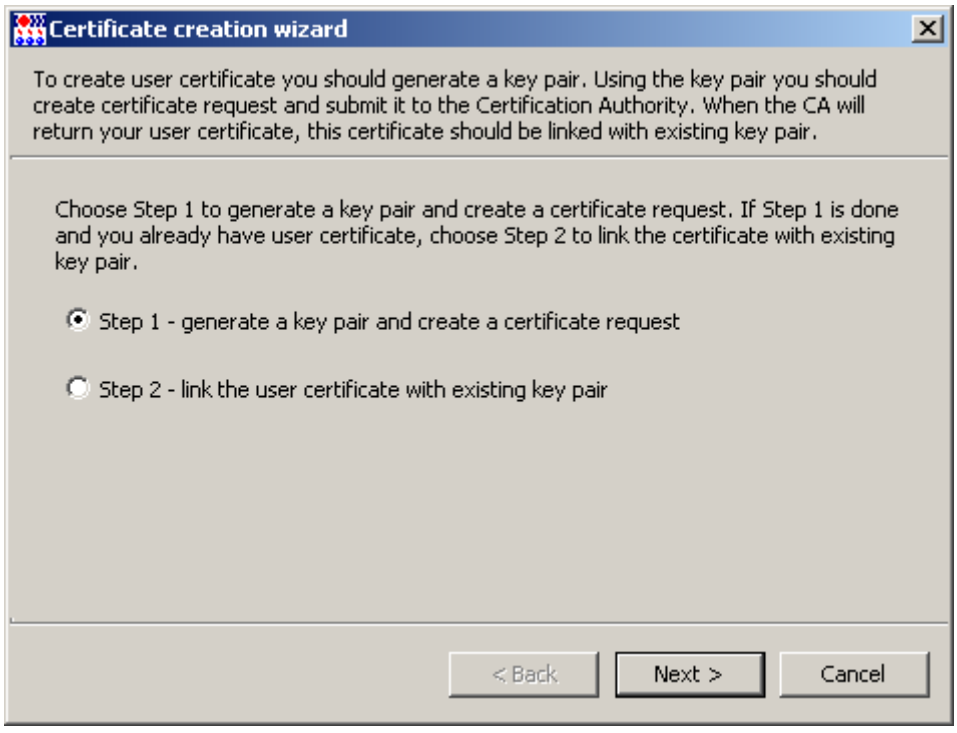

Рисунок 87

В открывшемся окне (Рисунок 88) заполните поля, составляющие поле Subject сертификата пользователя. В качестве алгоритма формирования и проверки ЭЦП, которым будет подписан запрос на сертификат будет использоваться алгоритм GOST_R3410EL (ГОСТ Р 34.10-2001). После заполнения нажмите кнопку *Next*.

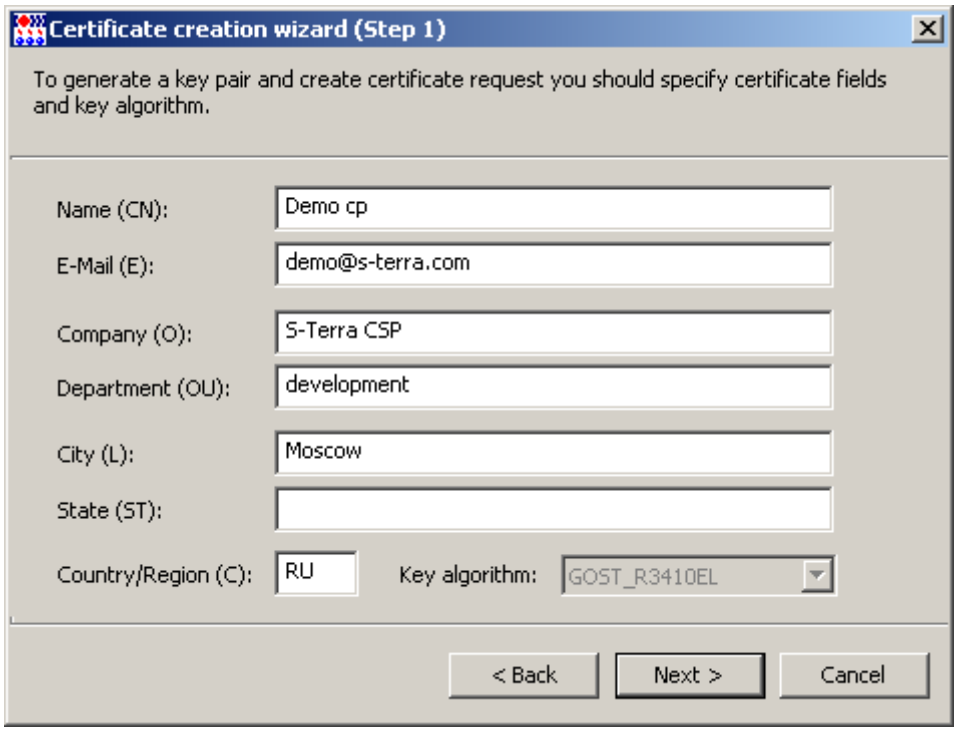

Рисунок 88

Введите PIN-код к токену, на который будет записан контейнер с созданной ключевой парой (Рисунок 89), и нажмите *OK*.

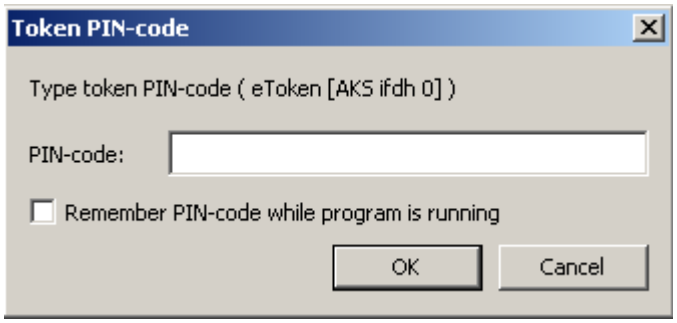

Рисунок 89

Для создания ключевой пары датчик случайных чисел просит нажать любую клавишу или подвигать мышкой (Рисунок 90).

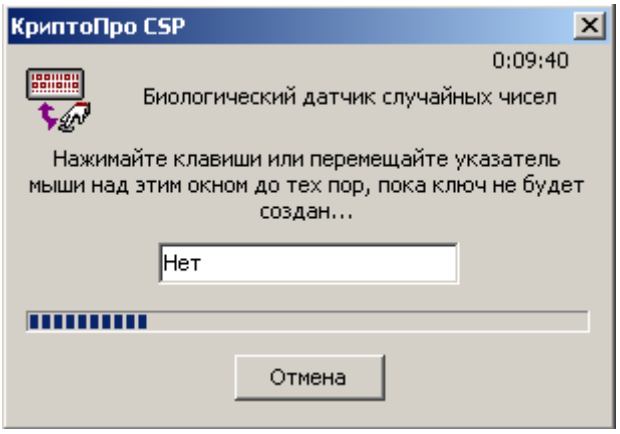

Рисунок 90

В следующем окне появится созданный запрос на сертификат пользователя, который можно скопировать в буфер или сохранить в файле, после этого нажмите кнопку *Finish* (Рисунок 91).

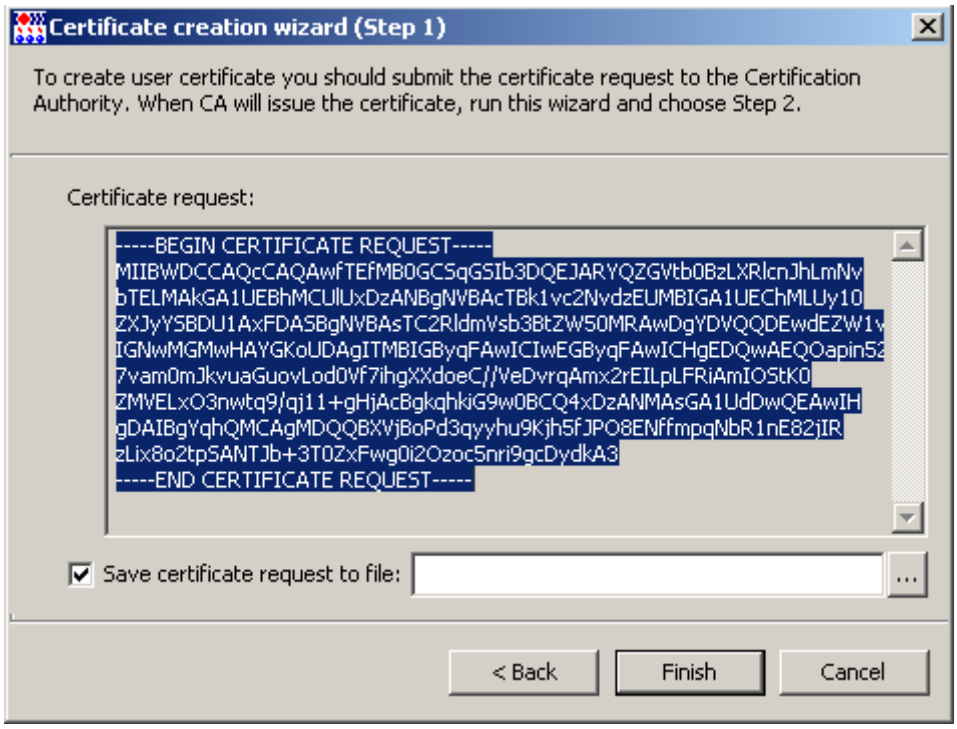

Рисунок 91

Созданный запрос отошлите в Центр сертификации, где будет создан сертификат пользователя на основе запроса и открытого ключа пользователя.

Запись сертификата на токен

Полученный сертификат пользователя из Центра сертификации надо записать на токен в контейнер с ключевой парой. В окне **Certificate creation wizard** поставьте переключатель в положение *Step 2* (Рисунок 92) и нажмите *Next*.

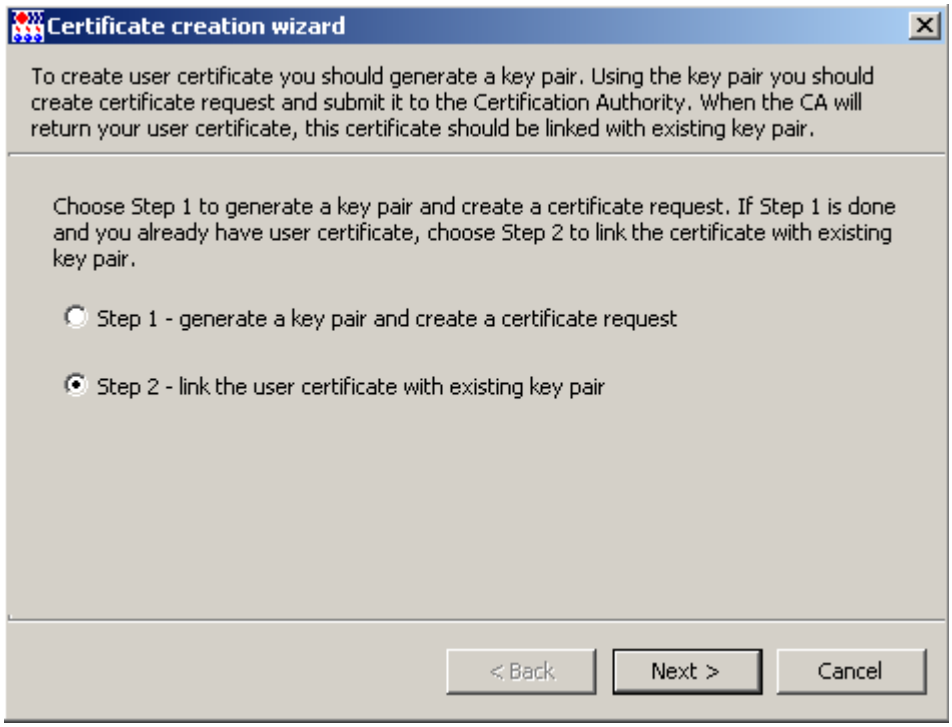

Рисунок 92

В окне **Certificate creation wizard (Step 2)** укажите сертификат пользователя (Рисунок 93):

- в положении *From file* – укажите имя файла в Base64-кодировке или
- в положении *From keyboard* – в поле укажите тело сертификата.

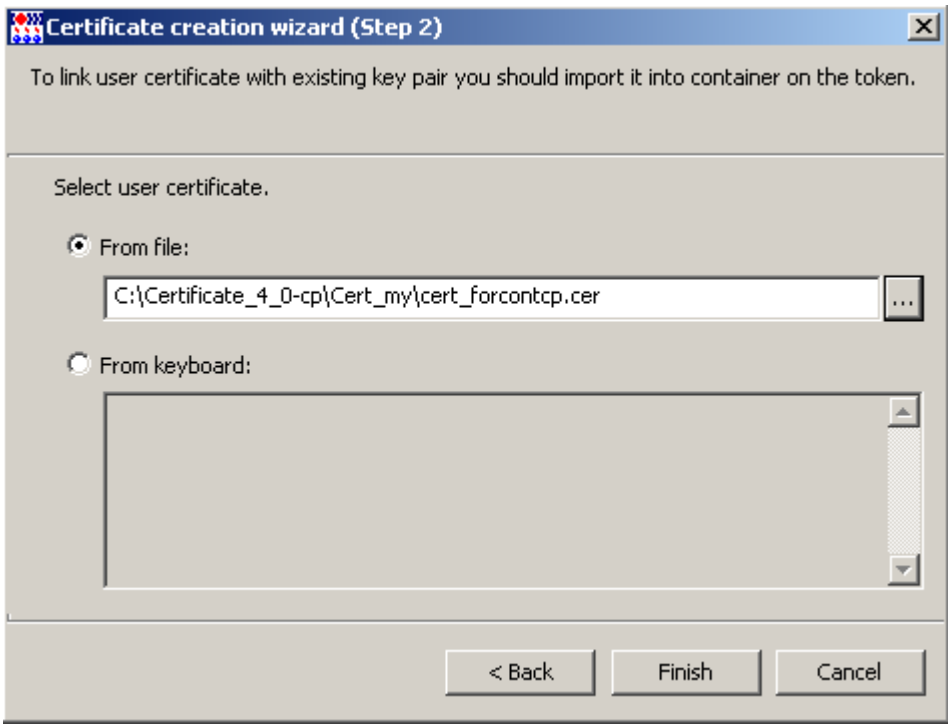

Рисунок 93

Для записи сертификата в контейнер на токене укажите PIN-код к токену (Рисунок 94).

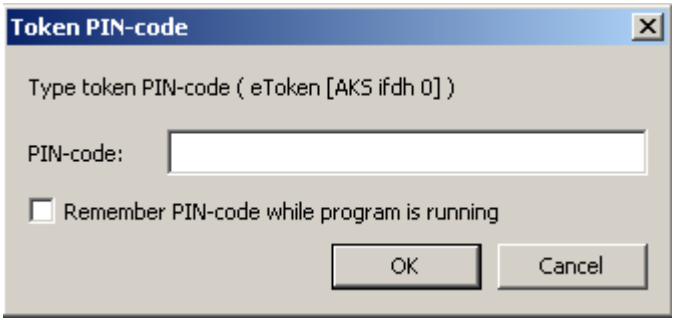

Рисунок 94

После этого во вкладке **Auth** выберите сертификат пользователя, записанный на токене (Рисунок 95). Для этого нажмите кнопку *Select certificate*.

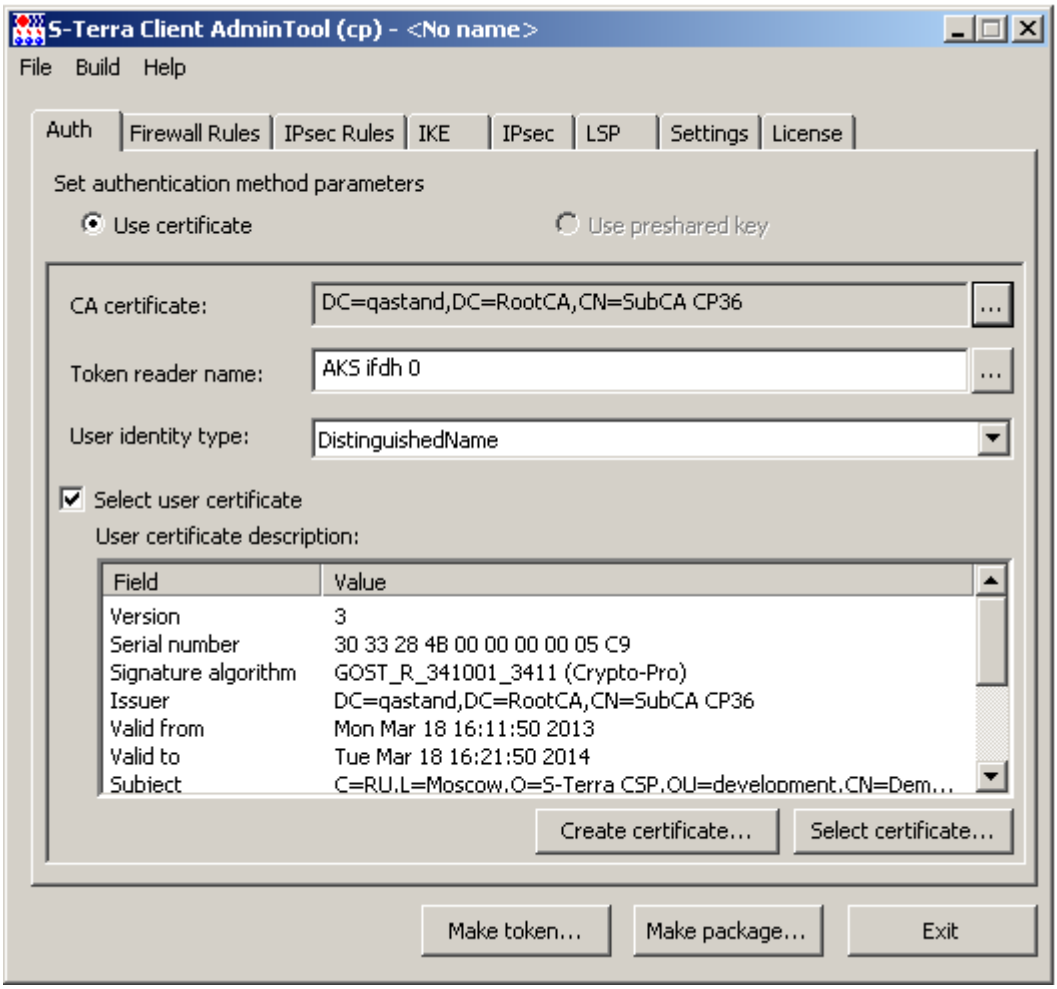

Рисунок 95

В появившемся окне с сертификатами на токене (Рисунок 96) выберите сертификат и нажмите *OK*. Заполнение вкладки **Auth** закончено.

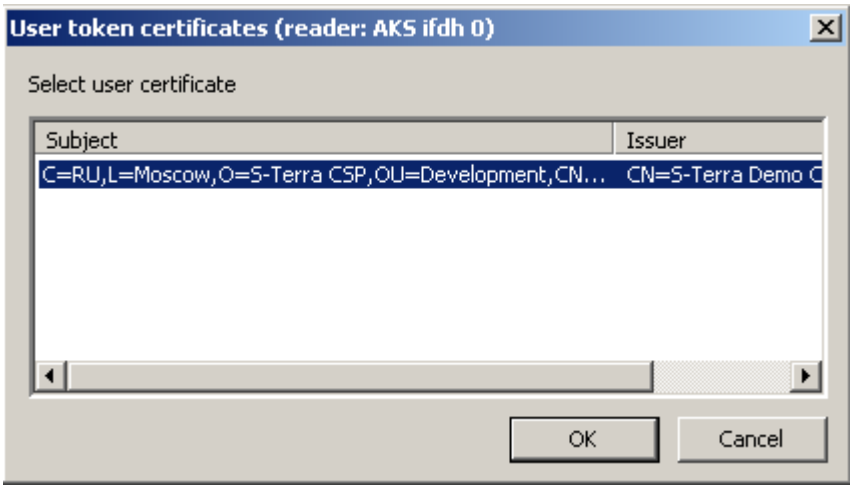

Рисунок 96

Во вкладке **Auth** необходимые сертификаты выбраны, перейдите в следующие вкладки.

7.4.2 Вкладки для создания политики безопасности

Все остальные вкладки для создания политики безопасности, предписанной для данного пользователя, заполняются также, как было описано ранее в разделах 7.3.3 – 7.3.9. Исключение составляет вкладка **Settings**.

7.4.3 Вкладка Settings

Вкладка **Settings** имеет такие же поля как и раньше, но флажок **Use non interactive user login** стал недоступен (Рисунок 97). В режиме пользовательского токена всегда используется интерактивный режим логина в Продукт S-Terra Client с выдачей окна запроса PIN-кода к токену.

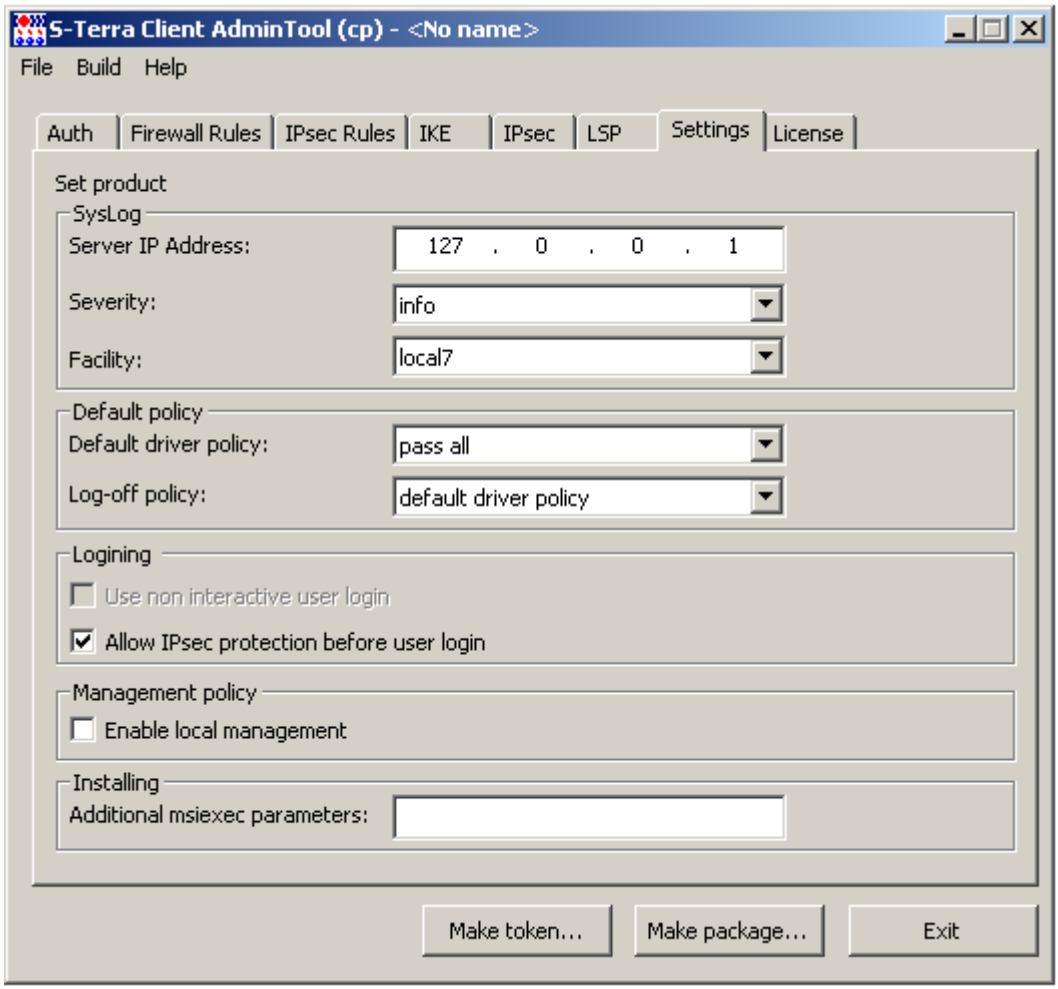

Рисунок 97

7.4.4 Создание инсталляционного файла

Для создания инсталляционного файла нажмите кнопку *Make package* в главной форме. В окне **Package parameters** введите такие же настройки как и в обычном режиме (см.раздел «Создание инсталляционного файла»). Но в режиме пользовательского токена создается универсальный инсталляционный файл, который не содержит никакой аутентификационной информации и политики безопасности, и может использоваться для разных пользователей. После установки такого файла, Продукт ориентирован на работу только с пользовательским токеном. И у каждого пользователя должен быть свой пользовательский токен.

7.4.5 Создание пользовательского токена

Для создания пользовательского токена нажмите кнопку *Make token* в главной форме. Появляется окно **Make token** (Рисунок 98) для запуска и отображения процесса записи на токен локальной политики безопасности и СА сертификата. Нажмите кнопку *OK*.

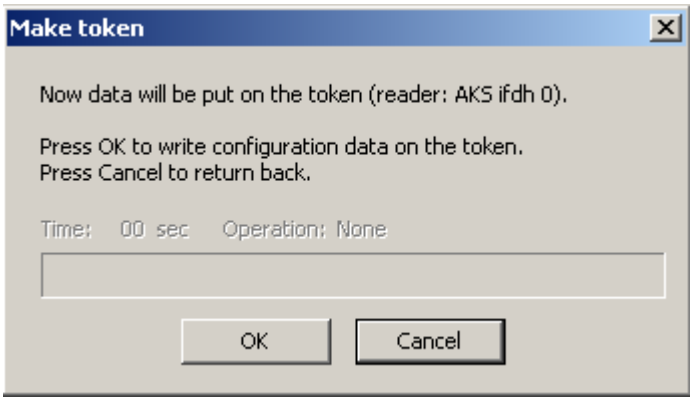

Рисунок 98

Перед обращением и записью на токен запрашивается PIN-код токена (Рисунок 99).

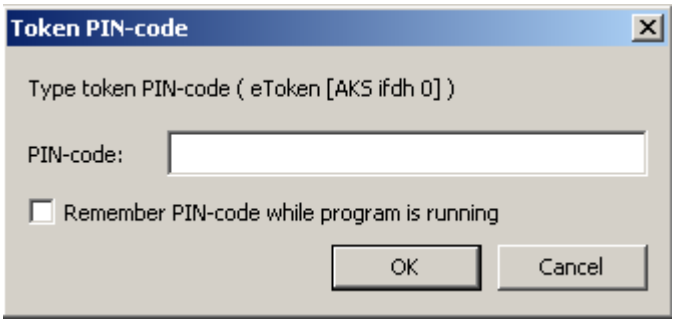

Рисунок 99

В строке состояния записи на токен могут отображаться следующие операции (Рисунок 100):

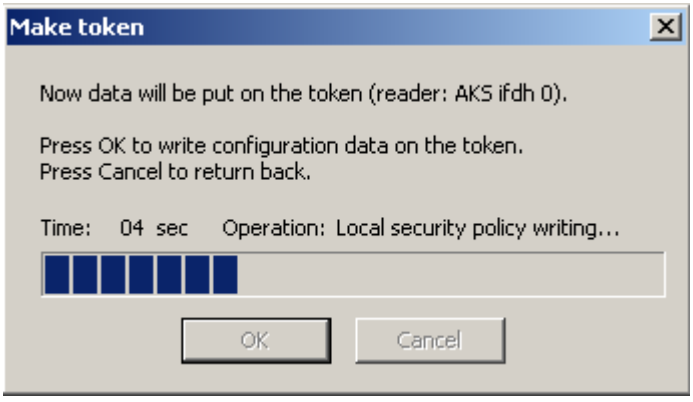

Рисунок 100

- None – процесс записи не запущен.
- Token system initialization... – процесс инициализации токена.
- Token login... – процесс логина на токен.
- Check user certificate... – проверка пользовательского сертификата (если во вкладке **Auth** пользовательский сертификат был указан явно – флажок Select user certificate).
- Previous objects removal... – удаление предыдущих данных с токена.

- Local security policy writing... – запись локальной политики безопасности на токен.
- CA certificates writing...— запись CA-сертификата на токен.
- Completed – процесс записи успешно завершен.

При окончании успешной записи на токен выдается об этом сообщение (Рисунок 101).

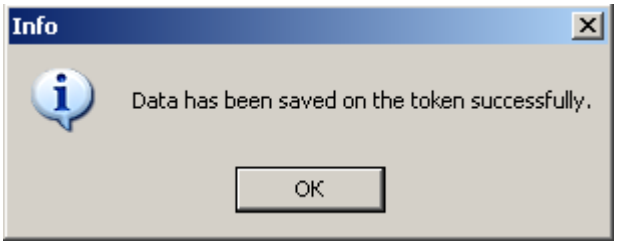

Рисунок 101

Таким образом, на пользовательский токен записаны локальная политика безопасности и CA сертификат в формате PKCS#11. Просмотреть эти данные можно с помощью сторонних утилит, например, eTExplorer.exe, компании Аладдин.

Кроме того, на токене лежит контейнер с ключевой парой и сертификатом пользователя.

На компьютер пользователя следует установить подготовленный инсталляционный файл и только после этого подключить пользовательский токен (см. раздел «Подготовка к инсталляции S-Terra Client»).

7.5 GUI. Режим международных алгоритмов

Режим международных алгоритмов в GUI – это режим создания инсталляционного файла Продукта С-Терра Клиент, работающего с использованием международных алгоритмов шифрования и проверки целостности пакетов.

В GUI переключение в этот режим осуществляется выбором в меню предложения **Build**, а в выпавшем списке – пункта **International pattern** (Рисунок 102). СКЗИ «КриптоПро CSP» в этом режиме не используется.

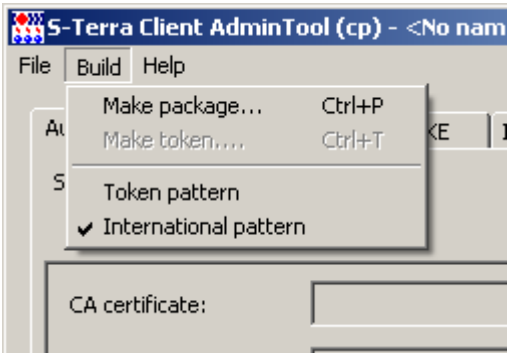

Рисунок 102

В режиме международных алгоритмов изменятся только три вкладки **Auth**, **IKE** и **IPsec** административного пакета, остальные вкладки останутся без изменений.

Этот режим отличается от других тем, что невозможно выполнить копирование контейнера на компьютере пользователя во время инсталляции S-Terra Client. И во-вторых, в инсталляционном файле можно перенести контейнер с секретным ключом с компьютера администратора на компьютер пользователя.

7.5.1 Вкладка Auth

Аутентификация сторон в этом режиме может осуществляться как с использованием сертификатов, так и с использованием предопределенных ключей. Вкладка **Auth** при использовании предопределенного ключа не изменилась, а при использовании сертификатов – приняла следующий вид (Рисунок 103).

Во вкладке доступны для заполнения следующие поля:

- **CA certificate** – здесь отражается поле Subject корневого сертификата Удостоверяющего Центра (Trusted CA Certificate), которому доверяет Продукт. Для этого разместите на компьютере администратора файл с Trusted CA сертификатом и в конце поля нажмите кнопку `[...]`, в открывшемся окне выберите файл с CA сертификатом. Обязательный параметр.
- **User certificate** – здесь отражается поле Subject сертификата пользователя. Для этого разместите на компьютере администратора файл с сертификатом пользователя и в конце поля нажмите кнопку `[...]`, в открывшемся окне выберите файл с этим сертификатом. Обязательный параметр.
- **User container name** – уникальное имя контейнера, размещенного на компьютере пользователя, на который будет установлен S-Terra Client. Обязательный параметр. Формат имени контейнера такой же, как и в основном режиме ГОСТ.
- **User container password** – пароль к контейнеру, размещенному на компьютере пользователя. Обязательный параметр.

- **User identity type** – тип идентификационной информации, пересылаемой партнеру при создании защищенного соединения. Обязательный параметр. Поле содержит выпадающий список со следующими значениями:
  - *Distinguished Name* – в качестве идентификатора партнеру будет высылаться значение Subject из сертификата пользователя, показываемое в поле **User identity value**, если оно там задано. Значение по умолчанию.
  - *Email* – в качестве идентификатора партнеру будет высылаться значение поля E-mail расширения сертификата пользователя, показываемое в поле **User identity value**, если оно там задано.
  - *FQDN* – в качестве идентификатора партнеру будет высылаться значение доменного имени хоста, на который будет установлен S-Terra Client, считываемое из поля DNS расширения сертификата, показываемое в поле **User identity value**, если оно там задано.
  - *IPV4Addr* – в качестве идентификатора партнеру будет высылаться первый IP-адрес, указанный в расширении сертификата, и показываемый в поле User identity value, если он там задан.
  - *Local IP address* – в качестве идентификатора партнеру будет высылаться действительный IP-адрес хоста, на котором будет установлен S-Terra Client.
- **User identity value** – значение идентификационной информации, пересылаемой партнеру. Поле доступно только для чтения и заполняется автоматически соответствующим типу идентификатора значением, считываемым из сертификата пользователя. Заполнение происходит в момент выбора типа идентификатора или изменения имени файла с сертификатом пользователя. Параметр обязательный.
- **Check consistency now** – установка этого флажка означает, что при создании инсталляционного файла будет проведена проверка соответствия сертификата пользователя и секретного ключа в контейнере. Для этого контейнер с секретным ключом разместите на компьютере администратора. Имя контейнера указывается в поле **Container name**, а пароль к нему – в поле **Container password**.
- **Container name** – имя контейнера на компьютере администратора для проведения проверки. При нажатии кнопки […] появится стандартное окно для открытия файлов в формате PKCS#12. Выберите нужный контейнер для проверки и нажмите *OK*. В поле **Container name** появится имя контейнера.
- **Container password** – пароль к контейнеру с секретным ключом.
- **Use container from admin computer** – установка этого флажка означает, что контейнер с секретным ключом будет взят с компьютера администратора и скопирован в инсталляционный пакет, а при инсталляции перенесен на компьютер пользователя (рекомендуется не выставлять это флажок, если канал доставки инсталляционного пакета не защищен). Параметры контейнера задаются полями **Container name** и **Container password**. При этом задание места расположения контейнера на пользовательском компьютере не требуется.

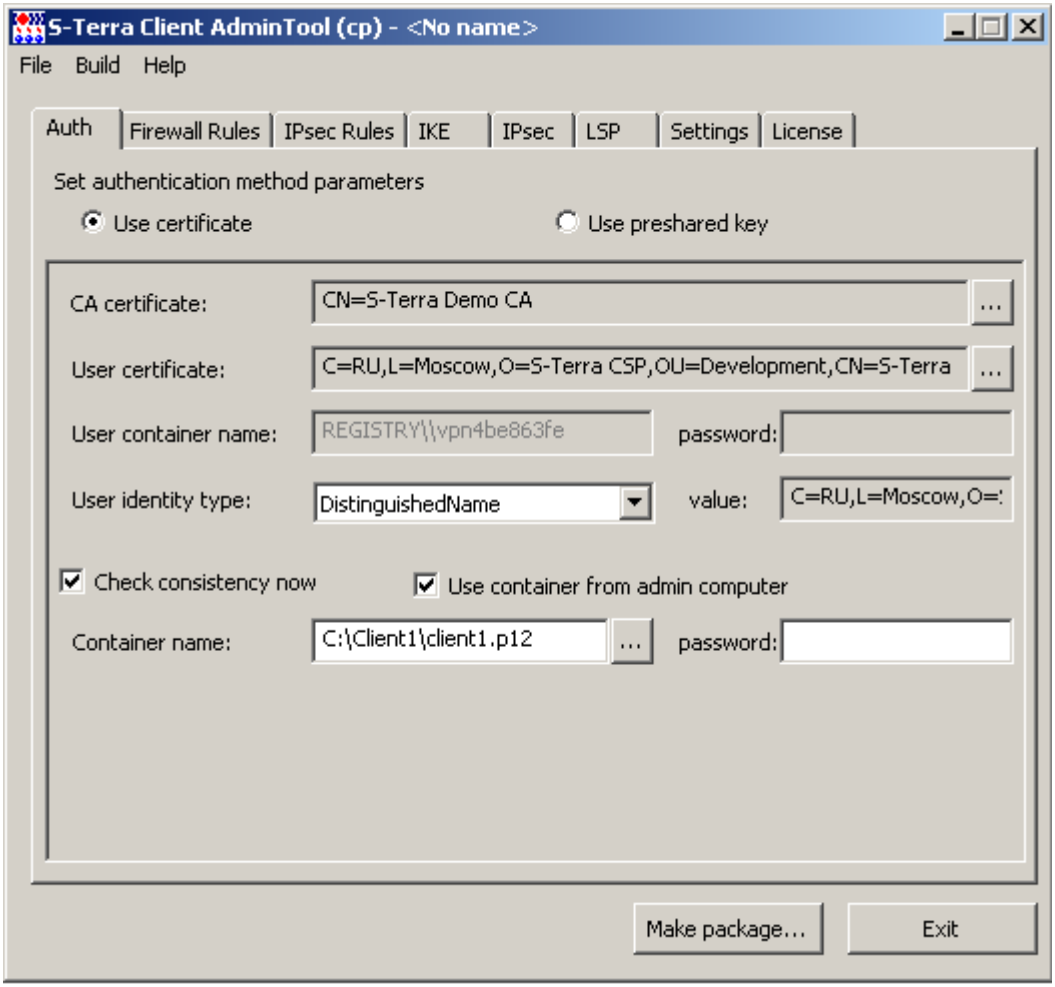

Рисунок 103

7.5.2 Вкладка IKE

Во вкладке **IKE** алгоритмы шифрования и проверки целостности ГОСТ заменены на международные алгоритмы – AES-256 и SHA1 (Рисунок 104). Общий сессионный ключ разной длины вырабатывается только по алгоритму Диффи-Хеллмана, алгоритм VKO ГОСТ Р 34.10-2001 – отсутствует. Остальные параметры остались без изменения.

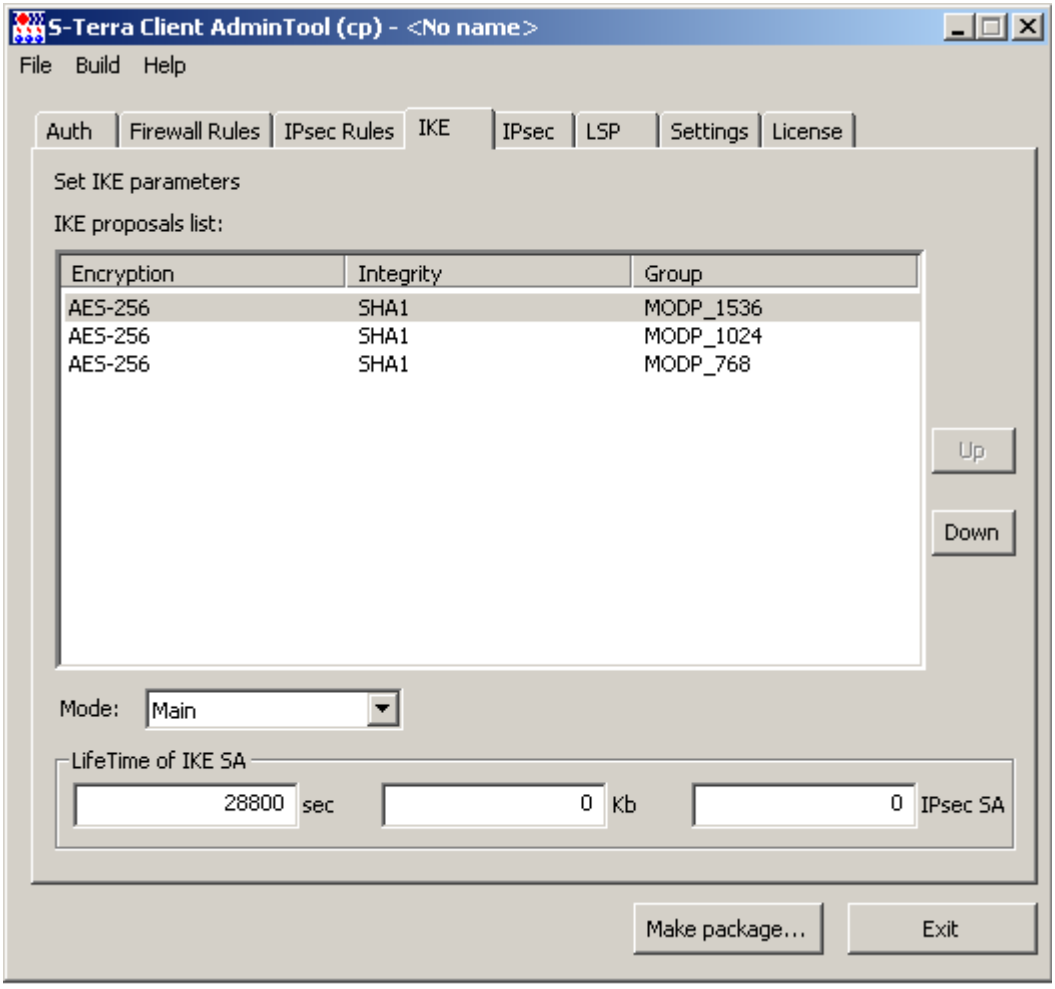

Рисунок 104

7.5.3 Вкладка IPsec

Во вкладке **IPsec** алгоритмы шифрования и проверки целостности ГОСТ заменены на международные алгоритмы – AES-256 и SHA1 (Рисунок 105).

Общий сессионный ключ разной длины вырабатывается только по алгоритму Диффи-Хеллмана, алгоритм VKO ГОСТ Р 34.10-2001 – удален.

Остальные параметры остались без изменения.

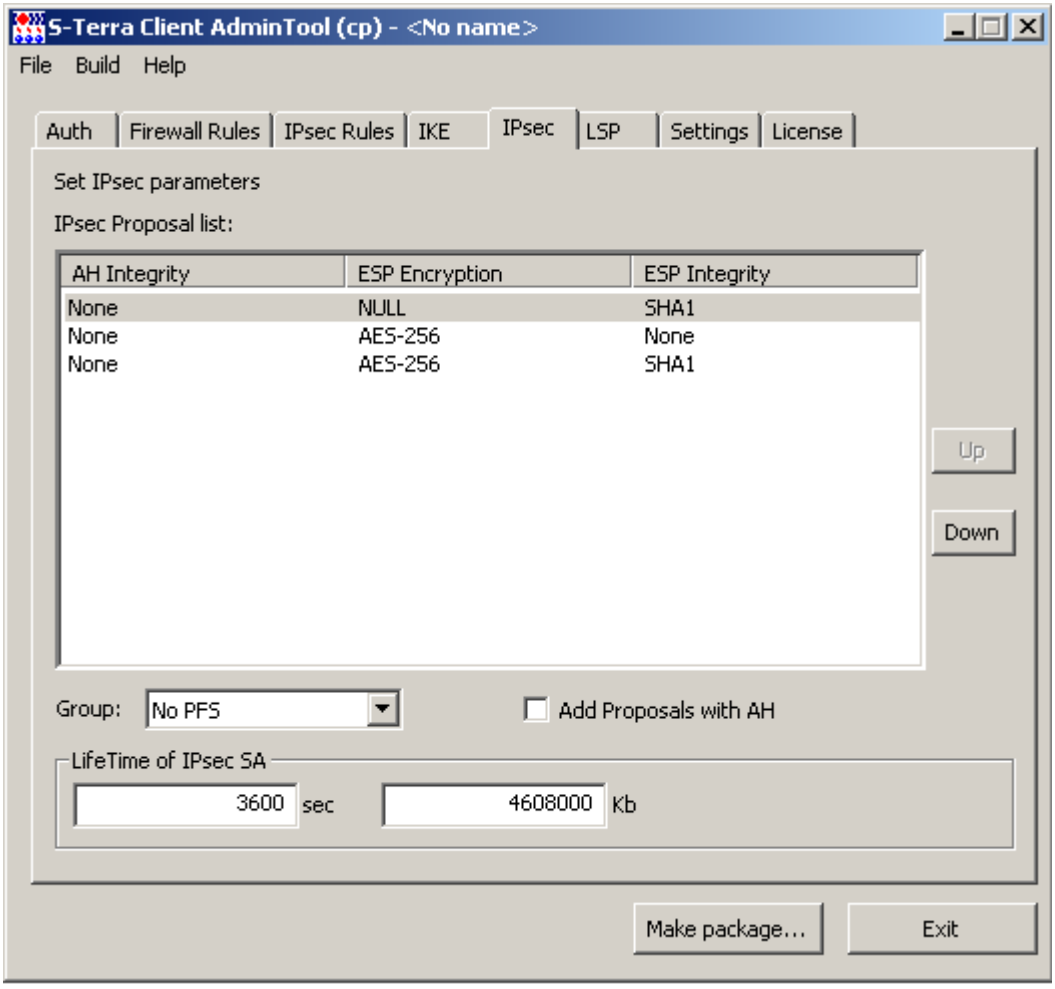

Рисунок 105

Все вкладки заполняются как в обычном режиме, аналогично создается и инсталляционный файл S-Terra Client.

7.6 Формат задания имен алгоритмов в файле admintool.ini

Имена алгоритмов, используемые во вкладках **IKE**, **IPsec** и **LSP**, задаются в файле `admintool.ini` в секции `[algorithm_names]`:

```
[algorithm_names]
ike-hash=GR341194CPRO1-65534 (ГОСТ Р 34.11-94)
ike-cipher=G2814789CPRO1-K256-CBC-65534 (ГОСТ 28147-89)
ah-integrity1=GR341194CPRO1-H96-HMAC-254 (ГОСТ Р 34.11-94)
ah-integrity2=G2814789CPRO1-K256-MAC-255 (ГОСТ 28147-89)
esp-integrity1=GR341194CPRO1-H96-HMAC-65534 (ГОСТ Р 34.11-94)
esp-integrity2=G2814789CPRO1-K256-MAC-65535 (ГОСТ 28147-89)
esp-cipher1=G2814789CPRO1-K256-CBC-254 (ГОСТ 28147-89)
esp-cipher2=G2814789CPRO1-K288-CNTMAC-253 (ESP_GOST-4M-IMIT)

[algorithm_names_international]
ike-hash=SHA1 (SHA1)
ike-cipher=AES-K256-CBC (AES-256)
ah-integrity=SHA1-H96-HMAC (SHA1)
esp-integrity=SHA1-H96-HMAC (SHA1)
esp-cipher=AES-K256-CBC (AES-256)
```

Для большей наглядности разрешается назначать алгоритмам пользовательские псевдонимы (в этом случае во вкладках **IKE** и **IPsec** будут отображаться не реальные имена, а назначенные псевдонимы). Для задания псевдонима необходимо дополнить строку имени алгоритма именем псевдонима, заключенного в круглые скобки:

Имеется возможность задать список алгоритмов для каждого семейства алгоритмов. При этом к имени семейства алгоритмов добавляется номер, начиная с 1. Порядок строк в файле не важен. Например:

```
ah-integrity1=GR341194CPRO1-H96-HMAC-254 (ГОСТ Р 34.11-94)
ah-integrity2=G2814789CPRO1-K256-MAC-255 (ГОСТ 28147-89)
```

Разрешение спорных ситуаций происходит по следующим правилам:

- Если в файле присутствует описание алгоритма без номера в имени семейства, то будет считан только он, даже не смотря на то, что в файле могут быть строки с описанием алгоритмов для этого же семейства с альтернативным синтаксисом для списков. Например, из файла содержащего следующие строки будет прочитана только первая строка:

  ```
  ah-integrity=GR341194CPRO1-H96-HMAC-254 (ГОСТ Р 34.11-94)
  ah-integrity1=GR341194CPRO1-H96-HMAC-254 (ГОСТ Р 34.11-94)
  ah-integrity2=G2814789CPRO1-K256-MAC-255 (ГОСТ 28147-89)
  ```

- Если при задании списка один или несколько номеров будет пропущен, то описания после пропуска прочитаны не будут. Например, из файла, содержащего следующие строки, будет прочитана только первая строка:

  ```
  ah-integrity1=GR341194CPRO1-H96-HMAC-254 (ГОСТ Р 34.11-94)
  ah-integrity3=G2814789CPRO1-K256-MAC-255 (ГОСТ 28147-89)
  ```

Если в файле присутствует две или более строки с одинаковым именем семейства алгоритмов, включая номер (текст перед знаком «=»), то будет прочитана только последняя.

8. Утилита make_inst

8.1 Сценарии подготовки инсталляционного пакета с помощью утилиты make_inst

Утилита `make_inst` предоставляет администратору безопасности интерфейс командной строки для задания локальных настроек Продукта S-Terra Client и создания инсталляционного файла Продукта S-Terra Client для пользователя.

При использовании ***предопределенного ключа*** для аутентификации сторон предоставляется возможность считывать созданный ключ из файла либо задать его значение в командной строке.

При подготовке ***сертификатов*** возможны два сценария, которые отличаются тем, кто создает ключевую пару для локального сертификата пользователя и на каком ключевом носителе размещен контейнер с секретным ключом локального сертификата, имеет ли администратор на своем рабочем месте доступ к этому контейнеру (см. описание в разделе «Атрибуты аутентификации»). Контейнер с секретным ключом должен быть уровня компьютера.

Первый сценарий

Шаг 1: Администратор безопасности получает от администратора CA Корневой сертификат Удостоверяющего Центра (Trusted CA Certificate) и сертификат пользователя, импортированные в файлы, и также контейнер на внешнем носителе.

> Поэтому в данном сценарии возможно на компьютере администратора провести проверку соответствия сертификата пользователя и секретного ключа в контейнере при создании инсталляционного файла.

Шаг 2: Администратор безопасности задает локальную политику безопасности (LSP) для данного пользователя в виде текстового файла (см. раздел «Создание локальной политики безопасности. Конфигурационный файл»).

Шаг 3: Администратор безопасности на своем рабочем месте запускает команду `make_inst`, в опциях задает файл с LSP для данного пользователя, путь к локальному и CA сертификату, имя контейнера с секретным ключом – где он будет размещен на компьютере пользователя, локальные настройки, создает инсталляционный файл S-Terra Client.

Шаг 4: Администратор безопасности передает пользователю подготовленный инсталляционный пакет, состоящий из:

- инсталляционного файла S-Terra Client
- контейнера с секретным ключом на внешнем ключевом носителе
- утилиты `integr_mgr`
- файла с контрольной суммой инсталляционного файла S-Terra Client.

Контейнер и файл с контрольной суммой должны быть переданы пользователю по заслуживающему доверия каналу связи. Инсталляционный файл S-Terra Client содержит базовый инсталляционный файл, локальную политику безопасности, сертификат пользователя и CA сертификат, персональные настройки.

Если администратор подготовил пользовательский токен, то пользователю передается подготовленный инсталляционный пакет, состоящий из:

- инсталляционного файла S-Terra Client
- пользовательского токена с записанным на нем CA сертификатом, локальным сертификатом, контейнером с секретным ключом и локальной политикой безопасности
- утилиты `integr_mgr` для вычисления контрольной суммы

- файла с х контрольной суммой инсталляционного файла S-Terra Client.

Пользовательский токен и файл с контрольной суммой должны быть переданы пользователю по заслуживающему доверия каналу связи.

Второй сценарий

Шаг 1: На компьютере пользователя создается ключевая пара и запрос на сертификат пользователя, который отсылается в Удостоверяющий Центр. Контейнер с секретным ключом размещается на компьютере пользователя в локальном хранилище (Реестре). Администратор безопасности получает Корневой сертификат Удостоверяющего Центра (Trusted CA Certificate) и сертификат пользователя, импортированные в файлы.

Администратор безопасности в результате на своем рабочем месте не имеет доступа к контейнеру, поэтому в данном сценарии невозможно на компьютере администратора провести проверку соответствия сертификата пользователя и секретного ключа в контейнере при создании инсталляционного файла.

Шаг 2: Администратор безопасности задает локальную политику безопасности (LSP) для данного пользователя в виде текстового файла (см. раздел «Создание локальной политики безопасности. Конфигурационный файл»).

Шаг 3: Администратор безопасности на своем рабочем месте запускает команду `make_inst`, в опциях задает файл с LSP для данного пользователя, путь к локальному и CA сертификату, имя контейнера на компьютере пользователя, локальные настройки, и создает инсталляционный файл S-Terra Client.

Шаг 4: Администратор безопасности передает пользователю подготовленный инсталляционный пакет, состоящий из:

- инсталляционного файла S-Terra Client
- утилиты `integr_mgr` для вычисления контрольной суммы
- файла с контрольной суммой инсталляционного файла S-Terra Client.

Файл с контрольной суммой должен быть передан пользователю по заслуживающему доверия каналу связи. Инсталляционный файл S-Terra Client содержит базовый инсталляционный файл, локальную политику безопасности, CA сертификат, локальный сертификат, ссылку местоположения контейнера на компьютере пользователя и персональные настройки.

Три режима

Подготовить инсталляционный пакет можно в одном из трех режимов:

- основной режим, в котором при создании инсталляционного файла S-Terra Client используются алгоритмы ГОСТ при шифровании, проверки целостности пакетов, формировании и проверки ЭЦП;
- режим международных алгоритмов, в котором при создании инсталляционного файла S-Terra Client используются международные алгоритмы шифрования, проверки целостности пакетов и ЭЦП;
- режим пользовательского токена.

С помощью утилиты `make_inst` можно создать одновременно инсталляционные файлы S-Terra Client для большого количества пользователей (см. раздел «Создание нескольких инсталляционных пакетов одновременно»).

Все сообщения, выдаваемые утилитой `make_inst` в процессе ее работы, выводятся в файл `make_inst_log.txt` (при каждом создании инсталляционного файла `make_inst_log.txt` переписывается).

8.2 Описание утилиты make_inst

Вызов утилиты `make_inst.exe` должен происходить из каталога административного пакета. В противном случае будет выдано сообщение об ошибке. Утилита имеет обязательные опции и необязательные, которые заключены в квадратные скобки.

```
make_inst.exe -o SFX_file_path -l LSP_file_path
```

при использовании Preshared_Key указываются опции:

```
-kn Preshared_key_name
{-kv Preshared_key_val|-kvf file_path_Preshared_key_val}
```

при использовании сертификатов указываются опции:

для CA сертификата:

```
-c CA_file_path
[-capwd CA_storage_password]|[-capwdf file_path_CA_storage_password]
[-caidx CA_object_index]
```

для локального сертификата:

```
-u USER_cert_file_path
{[-certpwd USER_cert_storage_password]|[-certpwdf
file_path_USER_cert_storage_password]}
[-certidx USER_cert_object_index]
[-uc USER_cert_container_name]
[-skt {signature | exchange}]
{[-up USER_cert_container_password]|[-ufp
file_path_USER_cert_container_password]}
```

для сертификатов партнеров или CRL:

```
[-p PARTNER_cert_1_file_path [-p PARTNER_cert_2_file_path] ...]
```

для копирования контейнера с одного носителя на другой, на компьютере пользователя при инсталляции S-Terra Client указываются опции:

```
{[-cs Source_USER_cert_container_name
 [-cp Source_USER_cert_container_password]|
 [-cfp file_path_Source_USER_cert_container_password]]}
```

для проверки соответствия сертификата пользователя и секретного ключа на компьютере администратора, для копирования контейнера в инсталляционный файл указываются опции:

```
[-chksecret {on | off}]
[-uac USER_cert_container_name_ADMIN]
{[-uap USER_cert_container_password_ADMIN]|[-uafp
file_path_USER_cert_container_password_ADMIN]}
[-ucpkgcopy {on | off}]
```

режим международных сертификатов:

```
[-intern {on | off}] (default: off)
```


локальные настройки:

```
[-q {basic | normal | silent}] (default: basic)
[-d {passall | passdhcp | dropall}] (default: passall)
```
(для релиза 14101 значение `dropall` недоступно)
```
[-f {ddp | passdhcp}] (default: ddp)
[-s {emerg |alert |crit |err |warning |notice |info |debug}]
(default: notice)
[-t <SYSLOG_server_IP>] (default: 127.0.0.1)
[-y <log_facility>] (default: log_local7)
[-a "<Additional_cmd_msiexec_params>"]
[-lic <license_file_path>]
[-nilogin {on | off}] (default: off)
[-login_protection { on | off }] (default: on)
[-local_mgmt { on | off }] (default: off)
[-token {on | off}] (default: off)
```

где:

-o SFX_file_path
`SFX_file_path` – имя создаваемого инсталляционного SFX-файла. Обязательная опция. Имя файла подразумевает и путь к этому файлу.
-l LSP_file_path
`LSP_file_path` – имя файла, содержащего LSP. Имеет текстовый формат. Обязательная опция, если не задан режим логина с использованием пользовательского токена `(-token on)`.
Использование предопределенного ключа
-kn Preshared_key_name
`Preshared_key_name` – имя предопределенного ключа. Обязательная опция, если используются предопределенные ключи. Может быть задано несколько таких ключей (см. Примечание 2). Предопределенные ключи или сертификаты обязательно должны быть заданы. Можно задавать и то, и другое.
-kv Preshared_key_val
`Preshared_key_val` – значение предопределенного ключа. Например, `-kv 12345` или `-kv "Test preshared key"` (кавычки в ключ не входят). Может быть задано несколько таких ключей (см. Примечание 2).
-kvf file_path_Preshared_key_val
`file_path_Preshared_key_val` – имя файла, содержащего значение предопределенного ключа на компьютере администратора. Если используется предопределенный ключ, то обязательно должна быть задана опция `-kv` либо `-kvf`. Может быть задано несколько таких ключей (см. Примечание 2).
Использование CA сертификата
-c CA_file_path
`CA_file_path` – имя файла с CA-сертификатом на компьютере администратора. Обязательная опция, если используются сертификаты.
-capwd CA_storage_password

CA_storage_password – пароль к хранилищу с CA сертификатом (если требуется, например, для файла в формате PKCS#12). Ситуации – отсутствие пароля (пароль не задан) и пустой пароль ("") – не различаются.

-capwdf file_path_CA_storage_password

file_path_CA_storage_password – имя файла, содержащего пароль к хранилищу с CA сертификатом. В этой и других подобных опциях пароль читается из файла как текстовая строка. Если файл содержит несколько строк, читается только первая из них и воспринимается как пароль. Нельзя указывать одновременно с опцией capwd.

-caidx CA_object_index

CA_object_index – порядковый номер CA сертификата в данном хранилище. Нумерация начинается с 1. Значение по умолчанию – 1.

Использование локального сертификата

-u USER_cert_file_path

USER_cert_file_path – имя файла с локальным сертификатом пользователя на компьютере администратора. Если опция не задана – берется хранилище с CA сертификатом. В этом случае порядковые номера CA и локального сертификатов в хранилище (caidx и certidx) должны различаться.

-certpwd USER_cert_storage_password

USER_cert_storage_password – пароль к хранилищу с локальным сертификатом (если требуется, например, для файла в формате PKCS#12). Ситуации - отсутствие пароля (пароль не задан) и пустой пароль ("") – не различаются. Используется, если задана опция -u.

-certpwdf file_path_USER_cert_storage_password

file_path_USER_cert_storage_password – имя файла, содержащего пароль к хранилищу с локальным сертификатом. Пароль читается из файла как текстовая строка. Если файл содержит несколько строк, читается только первая из них и воспринимается как пароль. Нельзя указывать одновременно с опцией certpwd.

-certidx USER_cert_object_index

USER_cert_object_index – порядковый номер локального сертификата в хранилище. Нумерация начинается с 1. Значение по умолчанию – 1.

Если не задана опция -u, используется одно и тоже хранилище для CA и локального сертификатов. В этом случае порядковые номера CA и локального сертификатов в хранилище должны различаться.

-uc USER_cert_container_name

USER_cert_container_name – имя контейнера с секретным ключом на компьютере пользователя. Здесь же указывается и носитель информации, на котором хранится контейнер. Не больше 60 символов.

Например, "REGISTRY\\container" (см. Примечание 3 об именах контейнеров).

Опция не указывается, если используется комбинация intern on и ucpkgcopy on. В противном случае, опция является обязательной, если используются сертификаты.

Если задана опция intern on и не задана, ucpkgcopy on то данная опция **-uc** будет указывать на файл в формате PKCS#12 с секретным ключом на компьютере

пользователя.
-up USER_cert_container_password
USER_cert_container_password – пароль к контейнеру с секретным ключом. Не больше 40 символов. Параметр актуален, если не задана опция -ufp. Ситуации – отсутствие пароля (пароль не задан) и пустой пароль ("") – не различаются. По умолчанию – пароль пустой.
-ufp file_path_USER_cert_container_password
file_path_USER_cert_container_password – имя файла, содержащего пароль к контейнеру, на компьютере администратора,. Не больше 40 символов. Пароль читается из файла как текстовая строка. Если файл содержит несколько строк, то читается только первая строка и воспринимается как пароль. Нельзя задавать вместе с опцией -up.
Указание сертификатов партнеров, промежуточного CA сертификата
-p PARTNER_cert_i_file_path
PARTNER_cert_i_file_path – путь к сертификату партнера или промежуточному CA-сертификату, который будет положен в базу локальных настроек продукта S-Terra Client при инсталляции. Можно задать несколько таких опций (в базу сертификаты будут положены в порядке перечисления данных опций). Необязательный параметр. Рекомендуется использовать в случаях, когда присутствуют проблемы с передачей сертификатов по протоколу IKE и LDAP.
Проверка соответствия сертификата пользователя и секретного ключа на компьютере администратора
-chksecret {on | off}
включение/выключение проверки соответствия сертификата пользователя и секретного ключа. По умолчанию – значение off. Такая проверка осуществляется на компьютере администратора и возможна только при наличии на нем контейнера с секретным ключом. Проверяется контейнер, указанный в опции -uac.
-uac USER_cert_container_name_ADMIN
USER_cert_container_name_ADMIN – имя контейнера на компьютере администратора. Эта опция используется только при включенной опции chksecret. В случае указанной опции -intern on данный параметр указывает на файл в формате PKCS#12, содержащему секретный ключ, на компьютере администратора
-uap USER_cert_container_password_ADMIN
USER_cert_container_password_ADMIN – пароль к контейнеру, указанному в опции -uac. По умолчанию – пароль пустой.
-uafp file_path_USER_cert_container_password_ADMIN
file_path_USER_cert_container_password_ADMIN – имя файла на компьютере администратора, в котором записан пароль к контейнеру, указанному в опции -uac. Нельзя задавать вместе с опцией **-uap.**
Режим международных сертификатов
-intern {on | off}

Включение/выключение режима использования международных (не ГОСТовых) сертификатов. При использовании международных сертификатов логика работы опций, связанных с сертификатами, может существенно меняться (см. описание соответствующих опций).
-ucpkgcopy {on | off}
Включен/выключен режим копирования сертификатного контейнера в инсталляционный файл. Используется только в режиме международных алгоритмов (опция `intern on`). В этом случае секретный ключ из контейнера, заданного в опции –uac, копируется в инсталляционный файл. При установке на компьютере пользователя данный секретный ключ переносится в базу локальных настроек S-Terra Client. Значение по умолчанию – `off`.
Копирование контейнера с одного носителя на другой при инсталляции S-Terra Client
-skt { exchange | signature }
задание типа секретного ключа. Данная опция игнорируется, если задана опция `-chksecret on`: в этом случае тип секретного ключа берется из проверяемого контейнера. Если создание ключевой пары и запроса на сертификат пользователя производились средствами MSCA и был выбран тип ключа `both` или `exchange`, то в этой опции нужно установить параметр `exchange`. Если был выбран тип ключа `signature`, то и в этой опции следует указать `signature`. Значение по умолчанию – `signature`. Эта опция указывается, если на компьютере пользователя происходит копирование контейнера. Но задавать тип секретного ключа необязательно, так как при его отсутствии будут последовательно перебираться все типы ключей при копировании контейнера. См. подробно в разделе "Копирование контейнера при инсталляции".
-cs Source_USER_cert_container_name
при указании этой опции будет произведено копирование контейнера с именем **Source_USER_cert_container_name**, размещенного на компьютере пользователя, в контейнер с именем `USER_cert_container_name`, которое указано в опции `-uc`, при инсталляции S-Terra Client на компьютере пользователя. Опция `-cs` задается, если используется сертификат. Если опция не задана, то копирование контейнера не производится. Копирование контейнера с точки зрения пользователя описанов разделе "Копирование контейнера при инсталляции".
-cp Source_USER_cert_container_password
`Source_USER_cert_container_password` – пароль к контейнеру с именем, указанным в опции `-cs`, который будет копироваться при инсталляции. Если пароль отсутствует или пустой, то опция `-cp` не задается. По умолчанию – пароль пустой.
-cfp file_path_Source_USER_cert_container_password
`file_path_Source_USER_cert_container_password` – имя файла, в котором записан пароль к контейнеру, указанному в опции `-cs`. Нельзя задавать одновременно с опцией `-cp`.
Локальные настройки
-q {basic | normal | silent}
тип инсталляции Продукта S-Terra Client: • `basic` – неинтерактивная установка с запросом на инсталляцию. Вариант по

умолчанию.

- `normal` – интерактивная установка (в диалоговом режиме) с демонстрацией Лицензии и другими окнами.
- `silent` – неинтерактивная установка без запросов. Стартует сразу после запуска EXE-файла без дополнительных запросов.

`-d {passall | passdhcp}`

`Default Driver Policy:`

- `passall` – пропускать все пакеты. Вариант по умолчанию
- `passdhcp` – ничего не пропускать, кроме DHCP-пакетов
- `dropall` – не пропускать трафик (для релиза 14101 это значение недоступно).

`-f {ddp | passdhcp}`

`Logoff policy -` специальная политика:

- `ddp - Default Driver Policy`. Вариант по умолчанию
- `passdhcp` – ничего не пропускать, кроме DHCP-пакетов.

`-s log_severity`

`log_severity ={EMERG|ALERT|CRIT|ERR|WARNING|NOTICE|INFO|DEBUG}`

По умолчанию – `NOTICE`. Опция задает общий уровень важности протоколируемых событий, ее использование описано в главе "Протоколирование событий".

`-t SYSLOG_server_IP`

`SYSLOG_server_IP` – IP-адрес SYSLOG сервера, на который будут посылаться сообщения о протоколируемых событиях. По умолчанию – 127.0.0.1 (сообщения будут присылаться на локальный хост).

`-y log_facility`

`log_facility =log_local 0-7`. По умолчанию `-log_local7`.

`-a "Additional_cmd_msiexec_params"`

`"Additional_cmd_msiexec_params"` – дополнительные параметры запуска WinInstaller. Например, альтернативная инсталляционная папка, настройки лога Windows Installer и т.п. Эти параметры можно посмотреть по ссылке http://msdn.microsoft.com/library/default.asp?url=/library/en-us/msi/setup/command_line_options.asp

`/l* C:\Client\install_log_file.txt` - протоколирование событий в файл C:\Client\install_log_file.txt при инсталляции S-Terra Client (рекомендуется при режиме silent).

`INSTALLDIR=каталог установки продукта` – переопределение каталога. Указание каталога, недоступного на компьютере пользователя, приведет к ошибке инсталляции.

Можно часть текста заключить в символы % (процент) и она будет рассматриваться как имя переменной окружения. И эта часть текста будет заменяться на значение переменной окружения (значения переменных окружения можно просмотреть командой set). Если переменная окружения отсутствует, в командке остается исходный текст.

Поддерживается специальная переменная окружения SfxDir – полный путь к папке, в которую распакованы данные. Таким образом, последовательность символов

%SfxDir% заменяется на полный путь к папке, в которую распакованы данные.

`REBOOT=F` –- обязательно запрашивать перезагрузку системы в конце инсталляции, даже если он не инициируется инсталлятором.

`REBOOT=S` – отключить запрос на перезагрузку системы в конце инсталляции. Не блокировать рестарт в случае ForceReboot action.

`REBOOT=R` – полностью отключить все запросы на перезагрузку системы, включая ForceReboot action. Используется для установки нескольких продуктов и/или выполнения дополнительных действий после инсталляции. После этого перезапустить систему вручную или с помощью сторонних инструментальных средств.

`MAX_SERVICE_START_TIMEOUT= ….` – время (в секундах) ожидания старта VPN сервиса (vpnsvc). Максимальное значение – 600 секунд. Значение по умолчанию – 30. Можно использовать для предотвращения появления сообщений об ошибке связи с сервисом на этапе логина для медленных и/или находящихся под сильной нагрузкой систем.

`AGENT_DB_REMOVE=1` – автоматически (без дополнительных запросов) будет удаляться база локальных настроек при установке или при удалении продукта (рекомендуется использовать для режима инсталляции `silent`)

`AGENT_DB_REMOVE=0` – база локальных настроек удаляться не будет, запросы пользователю выдаваться не будут. По умолчанию (параметр пустой) – пользователю выдается запрос на удаление базы локальных настроек.

`DISABLE_ANTIVIRUS_WARNING=1` – при инсталляции не будет показываться предупреждение 25036 (о необходимости отключения антивирусных программ).

Примечание 1: пользователь должен знать о необходимости отключения антивируса, иначе данный параметр использовать не следует.

`REBOOT_REQUIRED=1` – принудительно инициировать запрос на рестарт системы в конце инсталляции. Параметр обычно выставляется автоматически (при необходимости).

`DISABLE_CALL_LOGIN=1` – в конце инсталляции логин зе запустится. Устанавливать параметр имеет смысл только для интерактивного логина (NON_INTERACTIVE_LOGIN=0) на Windows Vista и более поздних версиях.

-lic license_file_path

`License_file_path` – имя файла с Лицензией на S-Terra Client на компьютере администратора. Эта опция обязательна для режимов инсталляции `basic` и `silent`. Для режима `normal` эта опция необязательна:

- если ее задать, то при установке Продукта вопросы о Лицензии задаваться не будут
- если ее не задать, то при установке Продукта появится стандартное окно для ввода Лицензии.

В текстовом файле данные Лицензии должны быть записаны в виде:

```
[license]
CustomerCode=NNNN
ProductCode=CLIENT/CLIENTB
LicenseNumber=NNNN
LicenseCode=NNNNNNN
```

-nilogin {on | off}

Включение/выключение неинтерактивного режима логина пользователя в Продукт S-Terra Client:

- `on` – включен неинтерактивный режим
- `off` – выключен неинтерактивный режим логина, работает интерактивный режим.

При неинтерактивном режиме логина при входе пользователя в систему производится попытка логина пользователя в Продукт с пустым паролем (в качестве пароля

используется пустая строка). При таком успешном логине в Продукт окно с запросом пароля не выводится. При неуспешном логине – Продукт ведет себя как при интерактивном логине (будет выдано окно запроса пароля).
-login_protection {on | off}
Включение/выключение функциональности по защите до логина в ОС. Значение по умолчанию – `on`.
-local_mgmt {on | off}
Включение/выключение возможности изменять настройки Продукта конечным пользователем. Значение по умолчанию – `off`.
-token {on | off}
Использование/неиспользование пользовательского токена для логина в Продукт S-Terra Client. Значение по умолчанию – `off`.

Примечание 2:

Если задается несколько предопределенных ключей, то опции с именем ключа и самим ключом (`-kn` и `-kv` или `-kvf`) должны следовать одна за другой, т.е. опции `-kn` и `-kv (-kvf)`, расположенные рядом относятся к одному и тому же предопределенному ключу.

Пример правильного задания ключей:

```
-kn key1 -kv value1 -kn key2 -kvf file_with_value2
```

Пример неправильного задания ключей (два имени и два значения расположены подряд):

`-kn key1 -kn key2 -kv value1 -kvf value2` – НЕПРАВИЛЬНО!!!

Примечание 3:

Имя контейнера имеет следующий формат

`\\.\READER\CONTAINER` или `READER\\CONTAINER` или `MEDIA\CONTAINER`,

где

`READER` – название считывателя ключевой информации

`MEDIA` – носитель ключевой информации

`CONTAINER` – имя контейнера.

Например:

`\\.\REGISTRY\cont_1` *или* `REGISTRY\\cont_2` (для реестра)

`\\.\FAT12_A\cont_3` или `FAT12\cont_4` (для дискеты)

`SCARD\ETOKEN_PRO32_4f22aa14\CC03\0FDA` (для eToken)

Если контейнер находится на внешнем ключевом носителе, то для подключения и инсталляции ключевых считывателей смотрите в «Приложении А» разделы «Подключение внешних ключевых считывателей», «Инсталляция внешнего считывателя и ключевого носителя информации в «КриптоПро CSP». А для указания уникального имени контейнера воспользуйтесь полем Container name GUI.

8.3 Сообщения об ошибках утилиты make_inst

	Сообщение	Пояснение
1	Error: SFX file path is missing	Не задан путь к SFX-файлу
2	Error: CA file path is missing	Не задан путь к CA-сертификату
3	Error: Local certificate file path is missing	Не задан путь к локальному сертификату
4	Error: Container name is missing	Не задано имя контейнера
5	Error: LSP file path is missing	Не задан путь к LSP
6	Error: Wrong install type	Неправильно задан тип инсталляции
7	Error: Wrong Default Driver Policy	Неправильно задана DDP
8	Error: Wrong Logoff Policy	Неправильно задана Logoff policy
9	Error: Wrong Log Severity	Неправильно задана Log Severity
10	Error: Wrong Log Facility	Неправильно задана Log Facility
11	Error: Wrong parameter: "…"	Неподдерживаемый параметр
12	Error: temporary directory creation failed	Не удалось создать временную директорию для работы утилиты
13	Error: Key creation failed	Не удалось создать описание контейнера ключа (наиболее вероятная причина – не удалось прочитать файл с паролем).
14	Error: installer files copy failed	Не удалось скопировать файлы инсталлятора
15	Error: CA not found	Не удалось найти файл с CA сертификатом
16	Error: Local certificate not found	Не удалось найти файл с локальным сертификатом
17	Error: LSP not found	Не удалось найти файл с LSP
18	Error: User preferences write failed	Не удалось создать пользовательские настройки
19	Error: Log settings write failed	Не удалось создать настройки лога
20	Error: SFX archive creation failed	Не удалось сформировать SFX-архив
21	Error: Preshared key value not found	Не удалось найти файл со значением Preshared ключа
22	Error: Source container is not applicable	Попытка задать исходный контейнер, когда не используются сертификаты или при использовании опции –intern on
23	Error: Partner certificate is not applicable	Партнерский сертификат неприменим (попытка задать опцию –p при отсутствии других опций, связанных с сертификатами)
24	Error: Source and destination containers have the same name	Исходный и рабочий контейнеры имеют одинаковые имена, что не допустимо
25	Error: Certificates or Preshared key should be set	Сертификаты или Preshared ключ должны быть заданы
26	Error: Preshared key name is missing	Не задано имя Preshared ключа
27	Error: Preshared key value is missing	Не задано значение Preshared ключа
28	Error: Preshared key name or value missed	Не задано имя или значение Preshared ключа

	Сообщение	Пояснение
29	Error: Preshared key names should be different	Имена Preshared ключей должны различаться
30	Error: Cannot run utility outside product dir	Не удается запустить утилиту вне директории продукта
31	Error: License should be set for non-interactive installation	Лицензия должна быть задана для неинтерактивной инсталляции
32	Error: Product incorrectly installed or damaged	Продукт некорректно установлен или поврежден
33	Error: Container name on the administrator computer is missing	Отсутствует контейнер на компьютере администратора
34	Error: Secret key copy is only usable with the container copy	Задано копирование секретного ключа, но не задано копирование сертификатного контейнера
35	Error: Secret key type should not be set due to secret key check	Тип секретного ключа не должен указываться, если задана проверка секретного ключа
36	Error: Unknown Driver Signing ignore mode	Неизвестный режим отключения Driver Signing сообщений
37	Error: Insertion of certficate container into package failed	Не удалось вставить сертификатный контейнер в инсталляционный пакет
38	Error: Container password reading from file failed	Не удалось прочитать из файла пароль на контейнер
39	Error: Source container password reading from file failed	Не удалось прочитать из файла пароль на исходный контейнер
40	Error: Container on administrator computer password reading from file failed	Не удалось прочитать из файла пароль на контейнер на компьютере администратора
41	Error: Secret key password reading from file failed	Не удалось прочитать из файла пароль на секретный ключ
42	Error: Cannot load CA: Internal error	Не удалось загрузить CA по невыясненной причине
43	Error: Cannot load CA: file not found or access denied	Не удалось загрузить CA: файл не найден или отказ в доступе
44	Error: Cannot load CA: Unknown storage format	Не удалось загрузить CA: неизвестный формат хранилища
45	Error: Password should be set to unlock the CA storage	Для доступа к хранилищу CA требуется пароль
46	Error: Cannot unlock the CA storage: Possibly incorrect password	Не удалось раскрыть хранилище CA: вероятно введен неправильный пароль
47	Error: Cannot get CA from the storage	Не удалось получить CA из хранилища
48	Error: Cannot get CA from the storage: object index exceeds number of objects in the storage	Не удалось извлечь CA из хранилища: порядковый номер объекта больше, чем количество объектов в хранилище
49	Error: Given certificate cannot be used as CA	Данный сертификат не может использоваться в качестве CA
50	Error: Cannot load local certificate: Internal error	Не удалось загрузить локальный сертификат по невыясненной причине
51	Error: Cannot load local certificate: file not found or access denied	Не удалось загрузить локальный сертификат: файл не найден или отказ в

	Сообщение	Пояснение
		доступе
52	Error: Cannot load local certificate: Unknown storage format	Не удалось загрузить локальный сертификат: неизвестный формат хранилища
53	Error: Password should be set to unlock the local certificate storage	Для доступа к хранилищу локального сертификата требуется пароль
54	Error: Cannot unlock the local certificate storage: Possibly incorrect password	Не удалось раскрыть хранилище локального сертификата: вероятно введен неправильный пароль
55	Error: Cannot get local certificate from the storage	Не удалось получить локальный сертификат из хранилища
56	Error: Cannot get local certificate from the storage: object index exceeds number of objects in the storage	Не удалось извлечь локальный сертификат из хранилища: порядковый номер объекта больше, чем количество объектов в хранилище
57	Error: Given certificate cannot be used as local certificate	Данный сертификат не может использоваться в качестве локального
58	Error: CA storage password reading from file failed	Не удалось прочитать из файла пароль на хранилище CA
59	Error: Local certificate storage password reading from file failed	Не удалось прочитать из файла пароль на хранилище локального сертификата
60	Error: Container password is set by two options: file and clear text. It is prohibited.	Пароль на контейнер задан двумя способами: через файл и открытым текстом. Это запрещено.
61	Error: Source container password is set by two options: file and clear text. It is prohibited.	Пароль на исходный контейнер задан двумя способами: через файл и открытым текстом. Это запрещено.
62	Error: Container on administrative computer password is set by two options: file and clear text. It is prohibited.	Пароль на контейнер на машине администратора задан двумя способами: через файл и открытым текстом. Это запрещено.
63	Error: Secret key password is set by two options: file and clear text. It is prohibited.	Пароль на секретный ключ задан двумя способами: через файл и открытым текстом. Это запрещено.
64	Error: CA storage password is set by two options: file and clear text. It is prohibited.	Пароль на хранилище CA задан двумя способами: через файл и открытым текстом. Это запрещено.
65	Error: Local certificate storage password is set by two options: file and clear text. It is prohibited.	Пароль на хранилище локального сертификата задан двумя способами: через файл и открытым текстом. Это запрещено.
66	Error: Wrong CA object index value	Неправильный порядковый номер объекта CA
67	Error: Wrong local certificate object index value	Неправильный порядковый номер объекта локального сертификата
68	Error: Local certificate storage password is set while local certificate storage is not set	Задан пароль хранилища локального сертификата, но хранилище локального сертификата не задано
69	Error: Container name is not applicable with -intern on and -ucpkgcopy on	Имя контейнера на пользовательской машине несовместимо с опциями –intern on и -ucpkgcopy on

	Сообщение	Пояснение
70	Error: Partner certificate '<file_path>' load failed	Не удалось загрузить партнерский сертификат <file_path> (наиболее вероятная причина – отсутствие или неправильный формат файла, заданного в опции –p)
71	Error: Partner certificate processing failed	Не удалось обработать партнерский сертификат
72	Error: Cannot load CA: Internal error	Не удалось загрузить CA: внутренняя ошибка
73	Error: Cannot load CA: file not found or access denied	Не удалось загрузить CA: файл не найден или к нему нет доступа
74	Error: Cannot load CA: Unknown storage format	Не удалось загрузить CA: неизвестный формат хранилища
75	Error: Password should be set to unlock the CA storage	Требуется задать пароль для разблокирования хранилища CA
76	Error: Cannot unlock the CA storage: Possibly incorrect password	Не удается разблокировать хранилище CA: возможно задан неверный пароль
77	Error: Cannot get CA from the storage	Не удается извлечь CA из хранилища
78	Error: Cannot get CA from the storage: object index exceeds number of objects in the storage	Не удается получить CA из хранилища: индекс объекта превышает количество объектов в хранилище
79	Error: Cannot load local certificate: Internal error	Не удалось загрузить локальный сертификат: внутренняя ошибка
80	Error: Cannot load local certificate: file not found or access denied	Не удалось загрузить локальный сертификат: файл не найден или к нему нет доступа
81	Error: Cannot load local certificate: Unknown storage format	Не удалось загрузить локальный сертификат: неизвестный формат хранилища
82	Error: Password should be set to unlock the local certificate storage	Требуется задать пароль для разблокирования хранилища с локальным сертификатом
83	Error: Cannot unlock the local certificate storage: Possibly incorrect password	Не удается разблокировать хранилище с локальным сертификатом: возможно задан неверный пароль
84	Error: Cannot get local certificate from the storage	Не удается извлечь локальный сертификат из хранилища
85	Error: Cannot get local certificate from the storage: object index exceeds number of objects in the storage	Не удается получить локальный сертификат из хранилища: индекс объекта превышает количество объектов в хранилище
86	Error: Cannot load partner certificate ‘<cert_file_path>’: Internal error	Не удалось загрузить партнерский сертификат ‘<cert_file_path>’: внутренняя ошибка
87	Error: Cannot load partner certificate ‘<cert_file_path>’: file not found or access denied	Не удалось загрузить партнерский сертификат ‘<cert_file_path>’: файл не найден или к нему нет доступа
88	Error: Cannot load partner certificate ‘<cert_file_path>’: Unknown storage format	Не удалось загрузить партнерский сертификат ‘<cert_file_path>’: неизвестный формат хранилища
89	Error: Password should be set to unlock the partner certificate ‘<cert_file_path>’ storage	Требуется задать пароль для разблокирования хранилища с партнерским

	Сообщение	Пояснение
		сертификатом ‘<cert_file_path>’ Примечание: в текущей версии продукта хранилища партнерских сертификатов, защищенные паролем, не поддерживаются.
90	Error: Cannot get partner certificate ‘<cert_file_path>’ from the storage	Не удалось извлечь партнерский сертификат ‘<cert_file_path>’ из хранилища
91	Error: Cannot load private key of the local certificate: Internal error	Не удалось загрузить секретный ключ локального сертификата: внутренняя ошибка
92	Error: Cannot load private key of the local certificate: file not found or access denied	Не удалось загрузить секретный ключ локального сертификата: файл не найден или к нему нет доступа
93	Error: Cannot load private key of the local certificate: Unknown storage format	Не удалось загрузить секретный ключ локального сертификата: неизвестный формат хранилища
94	Error: Password should be set to unlock the private key of the local certificate storage	Требуется задать пароль для разблокирования хранилища секретный ключ локального сертификата
95	Error: Cannot unlock the private key of the local certificate storage: Possibly incorrect password	Не удается разблокировать хранилище секретного ключа локального сертификата: возможно задан неверный пароль
96	Error: Cannot get private key of the local certificate from the storage	Не удается извлечь секретный ключ локального сертификата из хранилища
97	Error: Cannot get private key of the local certificate from the storage: certificate not found	Не удается извлечь секретный ключ локального сертификата из хранилища: не найден сертификат

8.4 Создание нескольких инсталляционных пакетов одновременно

Для создания инсталляционных пакетов для большого числа пользователей одновременно предлагается использовать BAT-файлы, вызывающие в цикле утилиту `make_inst.exe`. Далее описаны несколько BAT-файлов типичных сценариев. На компьютере администратора должна быть создана специальная папка для файлов пользователей. В этой папке создаются подпапки, которые называются по имени пользователей. Например, папка `c:\vpn_client`, в ней подпапки `c:\vpn_client\alice` и `c:\vpn_client\bob` (важно, чтобы не было посторонних подпапок). В этих подпапках лежит файл `localcert.crt`, а также для некоторых сценариев могут лежать файлы `ca.crt`, `lsp.txt` и `pwd.txt` (пароль на контейнер).

Сценарий 1. В этом сценарии контейнеры с секретными ключами пользователей имеют пустой пароль. Получаемые SFX-файлы кладутся в папки пользователей под именем `vpnclient.exe`. В папках пользователей лежат локальные сертификаты. Используется один CA сертификат и одна LSP для всех пользователей:

```
@echo off

SET TEMPLATE_DIR=c:\vpn_client
SET MAKE_INST_PATH=D:\S-Terra Client\make_inst.exe
SET CONTAINER_NAME=REGISTRY\\container
SET LSP_PATH=c:\vpn_client\lsp.txt
SET CA_PATH=c:\vpn_client\ca.crt

for /r %TEMPLATE_DIR% /d %%i in (*) do (%MAKE_INST_PATH% -o
%%i\vpnclient.exe -c %CA_PATH% -u %%i\localcert.crt -uc
%CONTAINER_NAME% -l %LSP_PATH%) & (if errorlevel 1 goto err)

goto :end

:err

echo An error occured
exit

:end

echo Make installations complete
```

Используются следующие настройки:

`TEMPLATE_DIR`	папка, в которой лежат подпапки пользователей. Путь должен быть без пробелов.
`MAKE_INST_PATH`	путь к утилите make_inst.exe.
`CONTAINER_NAME`	имя контейнера.
`LSP_PATH`	путь к общей LSP.
`CA_PATH`	путь к общему CA сертификату.

Здесь и далее фраза в конце "Make installations complete" обозначает успешное завершение, а "An error occured" – что произошла ошибка.

Сценарий 2. Используется общий пароль для всех контейнеров с секретными ключами всех пользователей. Получаемые SFX-файлы кладутся в папки пользователей под именем `vpnclient.exe`. Каждый пользователь имеет свой CA сертификат и свою LSP:

```
@echo off

SET TEMPLATE_DIR=c:\vpn_client
SET MAKE_INST_PATH=D:\S-Terra Client\make_inst.exe
```



```
SET CONTAINER_NAME=REGISTRY\\container
SET CONTAINER_PASSWORD=somepwd

for /r %TEMPLATE_DIR% /d %%i in (*) do (%MAKE_INST_PATH% -o
%%i\vpnclient.exe -c %%i\ca.crt -u %%i\localcert.crt -uc
%CONTAINER_NAME% -up %CONTAINER_PASSWORD% -l %%i\lsp.txt) & (if
errorlevel 1 goto err)

goto :end

:err

echo An error occured
exit

:end

echo Make installations complete
```

Новые настройки:

```
CONTAINER_PASSWORD – общий пароль.
```

Сценарий 3. Все условия аналогичны сценарию 2, но получаемые файлы кладутся в одну папку с именами username.exe (где username совпадает с именем пользовательской подпапки, например `alice.exe или bob.exe`):

```
@echo off

SET TEMPLATE_DIR=c:\vpn_client
SET MAKE_INST_PATH=D:\S-Terra Client\make_inst.exe
SET CONTAINER_NAME=REGISTRY\\container
SET CONTAINER_PASSWORD=somepwd
SET SFX_DIR=c:\sfx

cd %TEMPLATE_DIR%
for /d %%i in (*) do (%MAKE_INST_PATH% -o %SFX_DIR%\%%i.exe -c
%%~fi\ca.crt -u %%~fi\localcert.crt -uc %CONTAINER_NAME% -up
%CONTAINER_PASSWORD% -l %%~fi\lsp.txt) & (if errorlevel 1 goto err)

goto :end

:err

echo An error occured
exit

:end

echo Make installations complete
```

Здесь

```
SFX_DIR – папка, в которую размещаются получаемые файлы.
```

Сценарий 4. Выполняется при тех же условиях, что и в сценарии 2, но в каждой папке пользователя дополнительно лежит файл `pwd.txt`, содержащий пароль контейнера для данного пользователя. Кроме того, когда каждый пользователь будет устанавливать Продукт S-Terra Client из подготовленного для него инсталляционного файла, то он будет ставиться не в папку по умолчанию, а в папку `c:\my vpn` (с пробелом):

```
@echo off

SET TEMPLATE_DIR=c:\vpn_client
```



```
SET MAKE_INST_PATH=D:\S-Terra Client\make_inst.exe
SET CONTAINER_NAME=REGISTRY\\container
SET SFX_DIR=c:\sfx

cd %TEMPLATE_DIR%
for /d %%i in (*) do  (%MAKE_INST_PATH% -o %SFX_DIR%\%%i.exe -c
%%~fi\ca.crt -u %%~fi\localcert.crt -uc %CONTAINER_NAME% -ufp
%%~fi\pwd.txt -l %%~fi\lsp.txt -a "INSTALLDIR=\"c:\my vpn\"") & (if
errorlevel 1 goto err)

goto :end

:err

echo An error occured
exit
:end
echo Make installations complete
```


9. Подготовка к инсталляции S-Terra Client

Перед установкой Продукта S-Terra Client на компьютере пользователя выполните следующие действия:

1. Установите на компьютер пользователя сертифицированное средство доверенной загрузки, если уровень защиты ПК «S-Terra Client» от НСД равен КС2.
2. Установите программный Продукт СКЗИ «КриптоПро CSP», если он еще не установлен. При выполнении процедуры инсталляции СКЗИ выберите:

   вид установки – ***Выборочная***

   компоненты, которые необходимо установить – ***Криптопровайдер уровня ядра ОС***.
3. В «КриптоПро CSP» настройте `Биологический ДСЧ` для уровня защиты КС1 либо для уровня защиты КС2 аппаратный ДСЧ (в случае использования сертифицированных средств доверенной загрузки «Соболь» или «Аккорд») или `КриптоПро Исходный материал` (если для ССДЗ не поддерживается функциональность ДСЧ).
4. Если пользователь будет работать с eToken, установите набор драйверов и утилит "eToken PKI Client 5.1 SP1 для Microsoft Windows", который можно взять с web-страницы http://www.aladdin-rd.ru/support/download/177/, для работы с eToken PRO, eToken NG-OTP, eToken NG-FLASH, eToken PRO 72K(Java).
5. Далее необходимо создать или подключить ключевой считыватель для размещения временного контейнера для начального значения ДСЧ, создаваемого во время инсталляции S-Terra Client:

   а) Если для пользователя создан **пользовательский токен**, на котором записан CA сертификат, локальный сертификат, политика безопасности для пользователя и контейнер с секретным ключом, то:

   - подключать **пользовательский токен** к компьютеру пользователя на время инсталляции S-Terra Client не следует
   - инсталлируйте ключевой считыватель Реестр. Такая инсталляция описана в «Приложении А» в разделе «Инсталляция ключевого считывателя Реестр в «КриптоПро CSP».

   б) Если для аутентификации сторон будут использованы предопределенные ключи, то инсталлируйте ключевой считыватель Реестр. Такая инсталляция описана в «Приложении А» в разделе «Инсталляция ключевого считывателя Реестр в «КриптоПро CSP».

   в) Если контейнер с секретным ключом сертификата пользователя размещен в Реестре, то в Реестр будет записан и временный контейнер.

   д) Если контейнер с секретным ключом сертификата пользователя размещен на другом внешнем ключевом носителе, на него будет записан и временный контейнер:

   - подключите считыватель этого носителя, как описано в «Приложении А» в разделе «Подключение внешних ключевых считывателей» (eToken до установки драйверов подключать не следует).
   - инсталлируйте считыватель и ключевой носитель, как описано в «Приложении А» в разделе «Инсталляция внешнего считывателя и ключевого носителя в «КриптоПро CSP».

   е) Если контейнер с секретным ключом пользователя находится на дискете, то дискета должна быть вставлена в дисковод.
6. На время инсталляции отключите все антивирусные программы.

9.1 Рекомендации по ручной настройке Брандмауэра Windows

Данные рекомендации описывают ручную настройку Брандмауэра Windows для обеспечения работоспособности VPN сервиса.

Обычно действия, описываемые в данном разделе, выполняются автоматически инсталлятором. Но если на момент инсталляции служба Брандмауэра Windows была отключена, никаких действий с Брандмауэром Windows на этапе инсталляции не производится. Это может привести к частичной неработоспособности Продукта после запуска службы Брандмауэра Windows. В данной ситуации пользователь уведомляется предупреждением инсталлятора 25025.

Если вместо Брандмауэра Windows используется другой персональный firewall, в нем следует вручную внести настройки, аналогичные описываемым в этом разделе.

9.1.1 Ручная настройка Брандмауэра Windows на Windows XP

Войдите в *Панель управления -> Брандмауэр Windows*.

Перейдите во вкладку *Исключения* и нажмите кнопку *Добавить программу...*

В появившемся окне *Добавление программы* в поле *Путь* задайте полный путь к файлу vpnsvc.exe, который располагается в каталоге продукта (по умолчанию – C:\Program Files\S-Terra Client).

Нажмите кнопку *Изменить область...* и убедитесь, что выбрано *Любой компьютер (включая из Интернета)*. По умолчанию выставляется именно такая настройка.

Подтвердите настройку, нажав *OK*.

9.1.2 Ручная настройка Брандмауэра Windows на Windows Vista

Войдите в *Панель управления -> Система и ее обслуживание –> Администрирование -> Брандмауэр Windows в режиме повышенной безопасности*.

Для *Правила для входящих подключений* выберите *Действия –> Новое правило*...

Укажите *Тип правила – Настраиваемые*.

Для раздела *Программа* укажите:

Путь программы (задайте полный путь к файлу vpnsvc.exe, который располагается в каталоге продукта (по умолчанию – C:\Program Files\S-Terra Client).

Службы -> Настроить -> Применять только к службам.

Для раздела *Тип протокола* выберите – *UDP. Все порты*.

Затем в разделе *Область* укажите – *Любой IP-адрес*

В разделе *Действие – Разрешить подключение*.

В разделе *Профиль* – все (*Домен, Личный, Общий*).

В разделе *Имя – S-Terra VPN Service*.

Нажмите – *Готово*.

9.1.3 Ручная настройка Брандмауэра Windows на Windows 7

Войдите в *Панель управления -> Система и безопасность -> Брандмауэр Windows*.

Выберите раздел *Дополнительные параметры*.

Должно появиться окно *Брандмауэр Windows в режиме повышенной безопасности*.

Выберите – *Правила для входящих подключений и Действие – Создать правило...*

Выполните шаги:

Тип правила – Настраиваемые.

Программа:

- *Путь программы*. (задайте, нажав кнопку *Обзор...* полный путь к файлу vpnsvc.exe (располагается в каталоге продукта, по умолчанию – C:\Program Files\S-Terra Client).
- *Службы -> Настроить -> Применять только к службам*.

Тип протокола – UDP. Все порты

Область – Любой IP-адрес

Действие – Разрешить подключение

Профиль – все (*Доменный, Частный, Публичный*).

Имя – S-Terra VPN Service.

Нажмите *Готово*.

10. Сообщения об ошибках при инсталляции административного пакета и продукта S-Terra Client

Ниже приведены тексты сообщений об ошибках, которые могут появляться при инсталляции административного пакета и продукта S-Terra Client.

Таблица 6

	Текст сообщения	Примечание
1330	A file that is required cannot be installed because the cabinet file <file> has an invalid digital signature. This may indicate that the cabinet file is corrupt	Перед установкой Продукта должны быть установлены сертификаты от VeriSign, выпущенные не позднее 10.10.10, которые удостоверяют сертификат JSC S-Terra CSP, который в свою очередь подписывает драйвера, MSI и CAB. С сайта microsoft.com получите обновление "Update for Root Certificates" (ключевое слово – KB931125).
25001	License check failed.	Неправильная лицензия
25002	CryptoPro must be installed before the product installation.	Перед установкой Продукта должно быть установлено CryptoPro
25004	Delete existing product settings?	Удалить существующие настройки продукта? (вопрос при деинсталляции)
25005	Old product settings found. Delete existing product settings?	Найдены старые настройки продукта. Удалить существующие настройки продукта?
25006	RNG initialization failed. {Reason: <reason>.} Installation aborted. {RNG container path: <path>}, где <reason> может быть одним из следующих: Random initialization tool returned an error You must have Administrator privileges Random initialization tool not found Random initialization tool can't run: system error Random value initialization failed Container name generation limit exceeded	Не удалось создать RNG контейнер. {Причина: <reason>} Инсталляция прервана. Путь к RNG контейнеру: <path> основные <reason>: Утилита для инициализации ДСЧ вернула ошибку Вы должны иметь администраторские привилегии Утилита для инициализации ДСЧ не найдена Не удалось запустить утилиту для инициализации ДСЧ: системная ошибка Инициализация ДСЧ не произошла Исчерпан лимит генерируемых имен контейнера

	Текст сообщения	Примечание
25009	Copy certificate container failed. {Reason: <reason>.} Installation aborted. Source container path: <src>. Destination container path: <dst>.	Не удалось скопировать сертификатный контейнер. {Причина: <reason>} Инсталляция прервана. Путь к исходному контейнеру: <src>. Путь к новому контейнеру: <dst>.
25011	Could not copy certificate container. Container with this name already exists. Do you want to remove existing container and copy the container from the installation package? (Press “No” to keep existing container and to proceed without container copy)	Нельзя скопировать сертификатный контейнер, поскольку контейнер с таким именем уже есть. Хотите ли вы удалить существующий контейнер и скопировать контейнер из инсталляционного пакета? (Нажмите “No” для того, чтобы сохранить существующий контейнер и продолжить без копирования)
25016	Version {<Version> } of CryptoPro CSP is not supported. CryptoPro CSP version 3.6 must be installed before the product installation	Версия <Version> продукта КриптоПро CSP не поддерживается. Должно быть установлено КриптоПро CSP версии 3.6 до инсталляции продукта. Примечания: <Version> в сообщении может отсутствовать, если ее не удалось определить Для версии 3.6 с build меньше, чем 5402, к <Version> добавляется приписка “(beta)” К <Version> добавляется приписка “(unrecognized)”, если по каким-либо причинам не удалось определить build КриптоПро CSP.
25018	Product "<Product_name version>" was detected. You should uninstall it first before the installation.	Был обнаружен Продукт "<Product_name version>". Вам необходимо сначала деинсталлировать его.
	This product needs Windows 2000 or higher	Для Продукта необходима Windows 2000 или выше
25019	The "<dll_path>" was wrongly marked as the previous GINA DLL. The system GINA DLL will be used instead.	**[Windows XP]** Файл <dll_path> был ошибочно помечен, как предыдущая GINA DLL. Будет использована системная GINA DLL.
25020	The previous GINA DLL "<dll_path>" was not found. The system GINA DLL will be used instead.	**[Windows XP]** Предыдущая GINA DLL <dll_path> не найдена. Будет использована системная GINA DLL.
25021	Driver "<driver_name>" installation failed. Product installation aborted.	Не удалось установить драйвер <driver_name>. Инсталляция продукта прервана.
25022	Product "<Product_name version>" was advertised. You should uninstall it first before the installation.	Для продукта <Product_name version> была выполнена операция объявления пользователям (advertisement). Вы должны деинсталлировать его до инсталляции.

	Текст сообщения	Примечание
25023	There is no CryptoPro CSP driver library installed on the system. You should install it first before the installation. Product installation aborted.	Не установлена драйверная библиотека КриптоПро CSP. Вы должны инсталлировать ее до инсталляции продукта. Инсталляция продукта прервана.
25025	Failed to add Windows Firewall rule allowing network traffic for S-Terra VPN Service. If you intend to use Windows Firewall or other firewall, you should manually configure it to allow network traffic for S-Terra VPN Service.	Не удалось добавить правило Брандмауэра Windows открывающее сетевой трафик для VPN сервиса. Если Вы собираетесь использовать Брандмауэр Windows или другой firewall, вы должны настроить его вручную, чтобы пропустить сетевой трафик для VPN сервиса. Рекомендации по ручной настройке Брандмауэра Windows даны в разделе «Рекомендации по ручной настройке Брандмауэра Windows».
25026	You must have administrator privileges.	Вам необходимы администраторские привилегии.
25032	Version {<Version> } of CryptoPro CSP is not supported one. It could be incompatible with the product. Do you want to continue the installation?	Версия КриптоПро CSP официально не поддерживается продуктом. Она может быть несовместима с продуктом. Продолжить инсталляцию?
25033	The Visual C++ Redistributable Package is absent or damaged	Visual C++ Redistributable Package отсутствует или поврежден
25034	RNG initialization was canceled. Do you want to retry RNG initialization (press "No" to cancel the installation)?	Инициализация RNG была прервана пользователем. Вы хотите повторить инициализацию RNG (нажмите "No" для прерывания инсталляции)?
25035	Token Software must be installed before the product installation.	Программное обеспечение для поддержки токенов должно быть установлено до установки продукта. Примечание: сообщение появляется только если параметр инсталляции TOKEN_LOGIN=1.
25036	Please disable any anti-virus software during the installation Press "OK" if anti-virus is disabled or not installed Press "Cancel" to cancel the installation	Пожалуйста, отключите любую антивирусную программу на время инсталляции Нажмите "OK" если антивирус отключен или не установлен Нажмите "Cancel" для отмены инсталляции Примечание: появление данного сообщения может быть отключено с помощью параметра инсталляции DISABLE_ANTIVIRUS_WARNING

11. Создание локальной политики безопасности. Конфигурационный файл

Под политикой безопасности понимается совокупность правил, по которым обрабатываются пакеты входящего и исходящего трафиков. Пакеты могут проходить как пакетную фильтрацию, так и обработку с использованием криптографических алгоритмов – построение защищенных (VPN) туннелей между партнерами.

Создание локальной политики безопасности (LSP – Local Security Policy) S-Terra Client осуществляется путем написания конфигурационного файла в текстовом формате для VPN устройства.

11.1 Описание грамматики LSP

Описание LSP представляет собой последовательное описание структур данных, определяемых типом, именем, списком параметров (полей) и их значений. Синтаксис языка определяет формат описания структур данных, базовые типы значений полей структур. Синтаксические конструкции позволяют описывать иерархические структуры данных, число уровней которых не ограничено.

Формальное описание синтаксиса LSP-языка в виде БНФ (Бэкуса—Наура форма) приведено ниже. В БНФ описании названия нетерминальных символов заключены в угловые скобки, имена терминалов написаны большими буквами. Кроме того, простые терминалы, ключевые слова и разделители, записаны в одинарных кавычках. В БНФ-описании используются следующие терминалы: ИДЕНТ, СТРОКА, DOTDOT, ЦЕЛОЕ32, ДАТА, ВРЕМЯ, IP.

<cfg_data>::= <top_level_form> | <cfg_data> <top_level_form>

<top_level_form>::= <object_def> | <constant>

<constant>::= `const` <key_value>

<object_def>::= ИДЕНТ ИДЕНТ ‘(‘ <key_value_or_template_list> ‘)’

| ИДЕНТ ‘(‘ <key_value_or_template_list> ‘)’

<key_value_list>::= <key_value> | <key_value> <key_value_list>

<key_value_or_template_list>::= <key_value_or_template_list>

|<key_value_or_template> <key_value_or_template_list>

<key_value_or_template>::= key_value | template

<key_value>::= <l_value> ‘=‘ <r_value_list>

<r_value_list>::= <r_value> | <r_value_list> ‘,’ <r_value>

<r_value>::= ИДЕНТ | ИДЕНТ ‘<‘ ‘>‘

| ИДЕНТ ‘<‘ <key_value_or_template_list> ‘>‘

| ИДЕНТ ‘[‘ <r_value_list> ‘]’

| ‘(‘ <r_value_list> ‘)’ | ‘(‘ ‘)’

| ‘[‘ <r_value_list> ‘]’ | ‘[‘ ‘]’

| ИДЕНТ ‘(‘ <key_value_or_template_list> ‘)’

| СТРОКА

| ЦЕЛОЕ32 | ЦЕЛОЕ32 ‘..’ ЦЕЛОЕ32

| ЦЕЛОЕ32 ‘/’ ЦЕЛОЕ32 ‘/’ ЦЕЛОЕ32

| - ЦЕЛОЕ32

| IP | IP '..' IP | IP '/' ЦЕЛОЕ32

| ДАТА

| ВРЕМЯ

<l_value>::= ИДЕНТ | ИДЕНТ '*'

<template>::= '+' ИДЕНТ

Терминальные символы

Терминальный символ **ИДЕНТ** обозначает идентификатор. Идентификатор состоит из латинских букв, цифр, символов '_', ':', '$' и '-'. Он должен начинаться с латинской буквы или символа '_'. Запрещено использование идентификаторов, совпадающих с ключевым словом const. В качестве типа структуры запрещается указывать идентификатор NULL.

Примеры идентификаторов:

```
Moscow-16
_WWW_
IKECFGRequestAddress
IKERule
```

Терминальный символ **СТРОКА** служит для обозначения строки, состоящей из любых символов, заключенных в двойные кавычки (""). Если внутри строки необходим символ двойной кавычки, то его следует дополнить слева символом '\'. Для использования символа '\' (back-slash)в строке, его нужно указать два раза ('\\' – двойной back-slash). Допустимо указывать и один back-slash, т.к. при перекодировании восстанавливается двойной back-slash.

Примеры задания значений типа СТРОКА:

```
Title = "Moon Gate LSP"
IntegrityAlg = "MD5-H96-KPDK"
X509SubjectDN *= "C=RU,O=OrgName,OU=qa0,CN=snickers0"
```

Терминальный символ **ЦЕЛОЕ32** представляет 32-битное целое число без знака. Число может быть записано в десятичной или шестнадцатеричной системе счисления. Во втором случае оно должно начинаться цифрой и заканчиваться буквой 'h' или 'H'. В шестнадцатеричном и десятичном представлении запись числа не может быть длиннее 10 символов, включая букву 'h'.

Примеры задания числовых значений параметров:

```
RetryTimeBase = 4
BlacklogSessionsMax = 16
LifetimeKilobytes = 0abcdh
```

Терминальный символ **IP** обозначает сетевой адрес четвертой версии IP-протокола. IP-адрес состоит из четырех чисел, разделенных точками, где каждое из чисел принадлежит диапазону от 0 до 255.

Примеры IP-адресов:

```
PeerIPAddress = 192.168.2.1
```

Терминальный символ **ДАТА** представляется тремя целыми числами без знака, разделенными символом '/' – число/месяц/год.

Пример даты:

```
StartOfValidity = 24/03/ 2004
EndOfValidity = 3/6/2004
```


Терминальный символ **ВРЕМЯ**

Тип **ВРЕМЯ** представляется двумя числами, разделенными символом ':'. Время представляется в 24-часовом формате.

Примеры задания времени:

```
23:59 # без минуты полночь
1:1 # час ночи и одна минута
09:2 # 2 минуты десятого утра
01 : 02 # 2 минуты второго ночи
```

Терминальный символ **DOTDOT** обозначает две точки подряд, без разделителей **".."**. Используется для указания диапазона значений.

Пример

```
ProtocolID *= 20..30
```

Комментарии

Комментарии могут размещаться в любом месте текста между другими терминалами и являются разделителями, эквивалентными символам пробела. Вложения комментариев одного типа не допускаются. Поддерживаются следующие два вида комментариев:

Блочный. Начинается с символов "(*" и заканчивается символами "*)" или начинается символом '{' и заканчивается символом '}'.

Строковый. Начинается с символа '#', заканчивается символом перевода каретки <LF>.

Примеры комментариев:

```
20..30     # Диапазон чисел 20-30
Action *= (tunnel_IPsec_des_md5_action) (* будет описан ниже *)
```

Разделители

В качестве разделителей в LSP-языке могут быть использованы следующие символы: пробел, табуляция, <LF> и <CR>. Переходом на новую строку считается символ <LF>.

Разделители необходимы только для отделения терминалов ИДЕНТ, ЦЕЛОЕ32, IP, ключевого слова const друг от друга.

Значения полей структур

Значения полей структур (`r_value`) могут быть простого (базового) типа, например, целое число, текстовая строка, диапазон целых чисел, описанием или ссылкой на описание объекта, списком любых перечисленных значений или пустым списком.

Есть еще один возможный тип значения – `процедура`. Процедура определяется именем и набором именованных параметров со значениями, заключенными в угловые скобки или именем и списком неименованных параметров, заключенных в квадратные скобки.

Для описания списков могут использоваться круглые или квадратные скобки.

Примеры значений (r_value):

```
20..30                          # Диапазон чисел 20-30.
0.0.0.0..255.255.255.255        # Диапазон IP адресов.
4.3.2.4/24                      # подсеть с 4.3.2.0 с маской 255.255.255.0.
```



```
"abed"                          # Текстовая строка.
structure_ref                   # Ссылка на структуру "structure_ref".
[[a,b],[[k,l,m],x,y],4,c,6]     # Вложенные списки из ссылок на структуры
                                # (a, b, k, l, m, x, y, c) и чисел (4, 6) .
[]                              # Пустой список.
proc<x=10 y=24>                 # Процедура "proc" с параметрами x и y.
Filter(SourcePort = 500)        # Объект Filter со значением поля "SourcePort"
                                # равным 500.
```

Определение объекта

Определение объекта (`object_def`) состоит из типа объекта, имени объекта и списка полей со значениями. Предварительного описания типов внутри языка не существует, описание экземпляра объекта и есть определение типа. Наличие необходимых полей и соответствие значений типу объекта определяется на этапе семантического разбора.

В приведенном ниже примере описан объект типа "Filter" с именем "hostA", который содержит одно поле с именем "DestinationIP" со значением простого типа (IPv4-адрес) равным 23.4.5.6.

Пример:

```
Filter hostA (DestinationIP = 23.4.5.6 )
```

Имя поля

Имя поля (`l_value`) является идентификатором. Значением поля может быть единственное значение или список значений.

Пример:

```
fieldl = 1,2,3,4
field2 = 1
field3 = 1
```

В описании одного объекта не может быть двух полей с одинаковыми именами, но если значением поля является список, допускается альтернативный способ задания списка – повторение имени поля несколько раз.

Пример:

```
field* = 1
field* = 2
field* = 3, 4
```

что эквивалентно

```
field = 1,2,3,4
```

Для того чтобы отличить переопределение поля от списка, используется символ '*' после идентификатора. То есть при наличии '*', повторное описание поля будет интерпретировано как добавление элементов в список.

Это же правило действует при добавлении значений из шаблона.

Специальные конструкции

Для упрощения описания повторяющихся параметров предусмотрена возможность использования именованных констант, значений по умолчанию и шаблонов.

В отличие от других конструкций языка, которые подвергаются семантическому анализу, константы и шаблоны полностью обрабатываются на этапе синтаксического разбора.

Описание каждой константы начинается с ключевого слова const, за которым следует имя константы и ее значение (или список значений). Значением константы может являться любая конструкция, которая может быть значением поля структуры. Использование константы заключается в подстановке ее имени вместо значения поля структуры.

Пример:

```
const A = 10
const structure = Filter(SourceIP = 1.1.1.1)
const c1 = 1,2,3
const c2 = 4,5,6
```

Описание объектов o1 и o2

```
Filter o1 ( DestinationPort* = c1,c2)
Filter o2 ( DestinationPort* = A )
```

эквивалентно нижеследующему описанию:

```
Filter o1 ( DestinationPort* = 1,2,3,4,5,6)
Filter o2 ( DestinationPort* = 10 )
```

Шаблон (template) является константой, единственное значение которой является структурой того типа, к которой этот шаблон будет применен. Для использования шаблона, внутри описания структуры необходимо написать символ '+' и имя константы за ним. Подстановка шаблона заключается в копировании всех полей из структуры, которая является значением константы, в структуру, в которую шаблон подставляется.

Если в структуре, куда подставляется шаблон, присутствует поле, описанное в шаблоне, то возможны следующие варианты:

- в шаблоне и в структуре поле имеет признак списка – *, тогда значения объединяются в единый список, причем порядок составления списков соответствует порядку перечисления полей и шаблонов в структуре
- если признак списка в одном из описаний отсутствует, то будет ошибка разбора.

Пример:

Описание шаблона:

```
const icmp = Filter(ProtocolID* = 1)
```

Пример использования:

```
Filter h_pl ( +icmp DestinationIP = 23.4.4.5 )
Filter icmp_and_tcp ( +icmp ProtocolID* = 6 )
```

Эквивалентные описания:

```
Filter h_pl ( ProtocolID = 1 DestinationIP = 23.4.4.5 )
Filter ping_and_tcp ( ProtocolID = 1,6 )
```


11.2 Структура конфигурации

Ниже в таблице представлен состав структур данных с указанием их полей.

Используются следующие обозначения:

- линия напротив поля структуры указывает на описание структуры, используемой в качестве значения;
- ‘*>’ обозначает, что поле содержит список используемых структур;
- ‘**>’ обозначает, что поле содержит список списков используемых структур.
- Жирным шрифтом выделены обязательные поля структуры.

Для упрощения простые типы (число, строка, IP-адрес и т.д.) опущены.

GlobalParameters
- Title
- Version
- Type
- PreserveIPsecSA
- AllowNestedIPsec
- CRLHandlingMode
- FirewallLogPacketsThreshold
- FirewallLogTimeTHreshold
- FirewallLogStatesMax
- PersistentConnectionRetryDelay

RoutingTable
- Routes----------*>

FirewallParameters
- TCPEstablishedTimeout
- TCPFinTimeout
- TCPSynSentTimeout
- TCPSynRcvdTimeout
- TCPClosedTimeout
- TCPHalfOpenMax
- TCPHalfOpenLow
- TCPSessionRateMax
- TCPSessionRateLow
- TCPSessionsMax
- TCPStrictnessLevel

IKEParameters
- DefaultPort
- SendRetries
- RetryTimeBase
- RetryTimeMax
- SessionTimeMax
- InitiatorSessionsMax
- ResponderSessionsMax
- BlacklogSessionsMax
- BlacklogSessionsMin
- BlacklogSilentSessions
- BlacklogRelaxTime
- IKECFGDefaultAddress
- IKECFGPreferDefaultAddress
- SALifetimeDelta
- FragmentSize
- LocalPort
- NATTLocalPort

SNMPTrapSettings
- Receivers----------*>

Route
- **Destination**
- Gateway
- NetworkInterface

LDAPSettings
- Server
- Port
- SearchBase
- ConnectTimeout
- ResponseTimeout
- HoldConnectTimeout
- DropConnectTimeout

SNMPPollSettings
- LocalIPAddress
- Port
- **ReadCommunity**
- SysLocation
- SysContact

TrapReceiver
- **IPAddress**
- Port
- **Community**
- Version
- LocalIPAddress

11.3 Заголовок конфигурации. Структура GlobalParameters

Заголовок конфигурации представляет собой структуру, описывающую общие параметры S-Terra Client. В конфигурации должна быть только одна структура данного типа. Этой структуре имя не присваивается.

Имя структуры GlobalParameters

Атрибуты Title

Version

Type

CRLHandlingMode

AllowNestedIPsec

FirewallLogPacketsThreshold

FirewallLogTimeThreshold

FirewallLogStatesMax

PreserveIPsecSA

PersistentConnectionRetryDelay

Атрибут Title

Атрибут Title предназначен для краткого описания конфигурации (имя конфигурации).

Синтаксис	Title = СТРОКА
Значение	строка произвольного содержания
Значение по умолчанию	пустая строка

Атрибут Version

Атрибут Version определяет версию спецификации конфигурации.

Синтаксис	Version = СТРОКА
Значение	строка вида [0-9].[0-9]
Значение по умолчанию	пустая строка.

Атрибут Type

Атрибут Type специфицирует тип конфигурации, который определяет действия агента при ее активизации.

Синтаксис	Type = **PERMANENT | TEMPORARY**
Значение	PERMANENT – после успешной активизации конфигурации она сохраняется в базе Продукта, если она была активизирована из файла. При следующем запуске Продукта конфигурация будет автоматически активизирована из базы Продукта. TEMPORARY – после успешной активизации, конфигурация не сохраняется в базе Продукта и используется только в

	текущем сеансе работы Продукта.
Значение по умолчанию	PERMANENT

Атрибут CRLHandlingMode

Атрибут CRLHandlingMode определяет режим обработки списка отозванных сертификатов (CRL).

Синтаксис	CRLHandlingMode = **DISABLE|OPTIONAL|BEST_EFFORT|ENABLE**
Значение	DISABLE – при проверке сертификата CRL не обрабатывается OPTIONAL – при проверке сертификата CRL используется только в случае, если он был предустановлен в базу Продукта или получен (и обработан) в процессе IKE обмена и является действующим BEST_EFFORT – при проверке сертификата CRL используется только в том случае, если он является действующим, если это не так, то CRL может быть получен посредством протокола LDAP (агент смотрит адрес LDAP-сервера сначала в поле CDP сертификата, а затем ищет структуру LDAPSettings). Если CRL получить не удалось – сертификат принимается ENABLE – при проверке сертификата обязателен действующий CRL, если это не так, то CRL может быть получен посредством протокола LDAP. Если CRL получить не удалось – сертификат не принимается
Значение по умолчанию	ENABLE

Атрибут AllowNestedIPsec

Атрибут AllowNestedIPsec позволяет установить дополнительную фильтрацию для IPsec трафика.

Синтаксис	AllowNestedIPsec = **TRUE | FALSE**
Значение	TRUE – если входящий или исходящий пакет подпадает под IPsec-правило, для пакета применяется рекурсивный режим поиска правил (ниже поясняется, как это сказывается на обработке входящего и исходящего трафика). Если AllowNestedIPsec имеет значение TRUE, то исходящий пакет после инкапсуляции подвергается повторному поиску правил IPsec, пока результат поиска не будет простым действием PASS или DROP. Для входящих пакетов AllowNestedIPsec включает симметричные проверки: перед декапсуляцией происходит IPsec-фильтрация. Если найдено правило фильтрации, к которому не привязан последний примененный к пакету SA, пакет уничтожается. если обрабатывается локальный IPsec-пакет, то он декапсулируется и происходит IPsec-фильтрация. FALSE – включает упрощенную схему обработки пакетов,

	которая не предусматривает повторного поиска правил и потенциально работает быстрее.
Значение по умолчанию	FALSE

Атрибут FirewallLogPacketsThreshold

Атрибут FirewallLogPacketsThreshold задает количество пакетов, прошедших через соединение, при достижении которого форсируется вывод статистики по соединению в файл лога. Для сбора статистики по соединению необходимо, чтобы значение артрибута `FirewallLogStatesMax` не было равно 0.

Синтаксис	FirewallLogPacketsThreshold = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..2147483647
Значение по умолчанию	Если значение не задано, механизм прерывания сбора статистики при превышении заданного количества пакетов выключен. Т.е. теоретически возможна ситуация, когда счетчик для подсчета пакетов будет превышен и накопление начнется снова.
Примечание	Вывод накопленной статистики в файл лога может произойти раньше указанного значения, по истечении интервала времени для сбора статистики, заданого атрибутом `FirewallLogTimeThreshold`. После этого накопление статистики по соединению начнется заново.

Атрибут FirewallLogTimeThreshold

Атрибут FirewallLogTimeThreshold задает время накопления статистики по текущему соединению. При достижении установленного значения происходит вывод накопленной статистики в файл лога. Для сбора статистики по соединению необходимо, чтобы значение артрибута `FirewallLogStatesMax` не было равно 0.

Синтаксис	FirewallLogTimeThreshold = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..2147483647
Значение по умолчанию	300 секунд
Примечание	Вывод накопленной статистики в файл лога может произойти раньше указанного значения, если будет достигнуто допустимое количество пакетов, заданое атрибутом `FirewallLogPacketsThreshold`. После этого накопление статистики по соединению начнется заново.

Атрибут FirewallLogStatesMax

Атрибут FirewallLogStatesMax задает максимальное количество объектов статистики, в которых накапливается информация по соединениям.

Синтаксис	FirewallLogStatesMax = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..10000. Значение 0 говорит о том, что никакая информация накапливаться не будет, т.е. пакеты не обрабатываются, а в файл лога каждую минуту выводится

	информация о количестве пропущенных пакетов
Значение по умолчанию	1500

Атрибут PreserveIPsecSA

Атрибут PreserveIPsecSA позволяет задать сохранение IPsecSA при изменении конфигурационного файла.

Синтаксис	PreserveIPsecSA = **TRUE | FALSE**
Значение	TRUE – IPsec SA сохраняется при наличии следующих условий в момент изменения конфигурации (загрузки LSP): существует список правил фильтрации – FilterChain, привязанный к тому же набору NetworkInterface, что и FilterChain, к которому был привязан IPsecAction, по которому данный SA построен; в этом FilterChain для селектора SA находится подходящее правило пакетной фильтрации Filter с ExtendedAction = ipsec. Для IPsec SA, оставшихся от предыдущей конфигурации, не работает заблаговременная смена ключевой информации (Smooth Rekeying) и не происходит уведомление партнера о разрыве соединения (отсылка Delete Payload). Delete Payload, присланные от партнера, обрабатываются корректно. FALSE – все IPsec SA удаляются при любых изменениях в конфигурации следующих структур: IPsecAction, IKERule, AAASettings, фильтров NetworkInterface.IPsecPolicy, IKEParameters.LocalPort, IKEParameters.NATTLocalPort, GlobalParameters.AllowNestedIPsec, а также структур, на которые перечисленные ссылаются..
Значение по умолчанию	FALSE
Примечание	IPsec SA, оставшиеся от старой конфигурации, в той или иной мере могут нарушать новую политику безопасности или быть неработоспособными из-за несоответствия новой LSP. Администратор должен учитывать данную особенность и в сомнительных ситуациях устанавливать PreserveIPsecSA = FALSE.

Атрибут PersistentConnectionRetryDelay

Атрибут PersistentConnectionRetryDelay задает задержку перед повторной попыткой создать соединения с флагом IPsecAction.PersistentConnection = TRUE.

Если в конфигурации присутствуют правила с "PersistentConnection", производятся попытка построить по ним хотя бы один IPsec SA. Если попытка закончилась неудачей на любом из этапов, через указанную задержку попытка повторяется.
В зависимости от причины неудачи, задержка между попытками соединения может быть от PersistentConnectionRetryDelay до PersistentConnectionRetryDelay+`SessionTimeMax`*N, где N – число возможных попыток построить SA (число TunnelEntry в правиле и т.п.).
Значения более 1000000 воспринимаются как неограниченное ожидание. То есть повторной попытки построить IPsec SA не производится до перезагрузки конфигурации.

Синтаксис	PersistentConnectionRetryDelay = ЦЕЛОЕ32
Значение	Целое число из диапазона $1..2^{32}-1$.
Значение по умолчанию	10

11.4 Структура LDAPSettings

Структура LDAPSettings задает настройки протокола LDAP, который используется для получения сертификатов и списков отозванных сертификатов (CRL). В конфигурации может присутствовать только одна структура данного типа. Этой структуре имя не присваивается.

В случае отсутствия структуры:

- получение сертификатов посредством протокола LDAP невозможно
- если атрибут CRLHandlingMode структуры GlobalParameters имеет значение ENABLE или BEST_EFFORT, то CRL может быть получен посредством протокола LDAP только при наличии в сертификате, для которого производится проверка подписи, расширения CDP (CRL Distribution Point) с адресом LDAP-сервера.

Трафик LDAP-серверов должен быть учтен в правилах фильтрации, т.к. LDAP- пакеты фильтруются наравне с остальным трафиком.

Имя структуры	LDAPSettings
Атрибуты	Server
	Port
	SearchBase
	ConnectTimeout
	ResponseTimeout
	HoldConnectTimeout
	DropConnectTimeout

Атрибут Server

Атрибут Server задает адрес LDAP-сервера, к которому производится запрос на поиск сертификатов. Указанный в этом атрибуте адрес используется, если сертификат, для которого производится проверка подписи, не содержит расширение CDP (CRL Distribution Point) с адресом LDAP-сервера либо в этом поле прописанный путь к LDAP-серверу является неполным, и тогда добавляются данные из этой структуры.

Сначала делается попытка установить соединение по LDAP версии 2. Если эта попытка завершается с ошибкой LDAP_PROTOCOL_ERROR (наиболее вероятная причина - не поддерживается версия 2), то повторяется попытка установить соединение по LDAP версии 3.

Синтаксис	Server = IP
Значение	IP-адрес
Значение по умолчанию	LDAP –сервер не указан. Поведение агента аналогично случаю отсутствия структуры LDAPSettings в политике.

Атрибут Port

Атрибут Port задает порт LDAP-сервера. Если атрибут Server не задан или расширение сертификата CRL Distribution Point содержит адрес LDAP-сервера, то данный атрибут игнорируется.

Синтаксис	Port = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..65535
Значение по умолчанию	389

Атрибут SearchBase

Атрибут SearchBase задает имя (Distinguished Name, DN) корневого X.500-объекта, в поддереве которого производится поиск сертификатов и CRL на LDAP-сервере. Указанное имя дополняет запрос, созданный на основе имени из сертификата или CRL, позволяя находить соответствующий X.500-объект в случае, когда исходное имя в запросе является частью имени этого объекта. Для запроса на основе URL данное имя не используется.

Синтаксис	SearchBase = СТРОКА
Значение	строковое представление DN в соответствии с RFC2253. Относительные имена (Relative Distinguished Name, RDN) указываются в порядке от объекта к корню
Значение по умолчанию	поиск производится по имени, полученному из сертификата или CRL.

Атрибут ConnectTimeout

Атрибут ConnectTimeOut позволяет ограничить время (в секундах) создания TCP-соединения с LDAP-сервером.

Синтаксис	ConnectTimeOut = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..6000
Значение по умолчанию	не устанавливается, что приводит к тому, что время создания TCP-соединения с LDAP-сервером ограничивается установленным для ОС временем создания TCP-соединения.
Примечание	Если в момент обращения к LDAP-серверу устройство, на котором он установлен, недоступно, то процесс создания TCP-соединения может занимать продолжительное время (до 3 минут, зависит от ОС). По этой причине могут наблюдаться внешние признаки зависания агента, и это может служить причиной неудачной попытки создания соединения.

Атрибут ResponseTimeout

Поиск посредством протокола LDAP может занимать достаточно продолжительное время, оно зависит от многих факторов, в том числе от масштаба запроса и характеристик канала передачи данных. Данный атрибут позволяет ограничить время (в секундах), в течение которого ожидается ответ от LDAP-сервера на единичный запрос.

Синтаксис	ResponseTimeOut = ЦЕЛОЕ32
Значение	Целое число из диапазона $1..2^{32}-1$.
Значение по умолчанию	200

Атрибут HoldConnectTimeout

Атрибут HoldConnectTimeout устанавливает период времени, в течение которого держится установленное соединение к серверу на случай, если придет к нему повторный запрос.

Синтаксис	HoldConnectTimeOut = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..6000 При значении 0 после обмена с LDAP-сервером соединение с ним сразу закрывается. В виду наличия погрешности в одну секунду не рекомендуется выставлять значение в 1 секунду, поскольку это может привести в некоторых случаях к немедленному закрытию соединения и к избыточному открытию нового соединения.
Значение по умолчанию	60

Атрибут DropConnectTimeout

Атрибут DropConnectTimeout устанавливает период времени, начиная с первой неудачной попытки создания соединения с LDAP-сервером, в течение которого новые попытки создания соединения с ним игнорируются.

Синтаксис	DropConnectTimeOut = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..6000 При значении 0 в случае неудачной попытки установления соединения с LDAP-сервером новые попытки не игнорируются. В виду наличия погрешности в одну секунду не рекомендуется выставлять значение в 1 секунду, поскольку это может привести в некоторых случаях к избыточным попыткам создания соединения.
Значение по умолчанию	5

Пример

Пусть сертификат партнера имеет `Subject = "cn=candy,ou=nomadic"`

Для поиска такого сертификата на LDAP-сервере (`Active Directory` –Рисунок 106), необходимо указать атрибут `SearchBase`:

```
LDAPSettings (
   Server = 10.1.1.1
   SearchBase="ou=scenario10,ou=QA,ou=GINS,dc=qamsca,dc=ginsoftware,
   dc=ru"
)
```


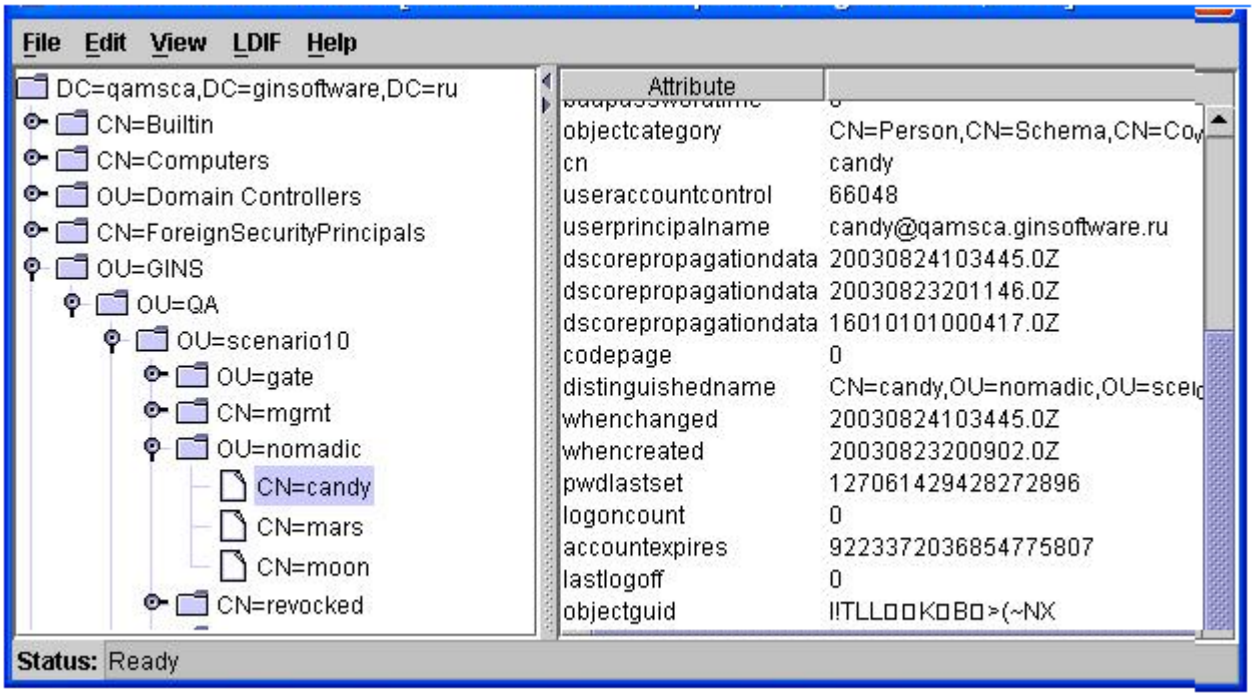

Рисунок 106

11.5 Структура IKEParameters

Структура IKEParameters описывает глобальные настройки протокола IKE. В конфигурации должна быть только одна структура данного типа. Этой структуре имя не присваивается.

Имя структуры	IKEParameters
Атрибуты	DefaultPort
	LocalPort
	NATTLocalPort
	SendRetries
	RetryTimeBase
	RetryTimeMax
	SessionTimeMax
	InitiatorSessionsMax
	ResponderSessionsMax
	BlacklogSessionsMax
	BlacklogSilentSessions
	BlacklogSessionsMin
	BlacklogRelaxTime
	IKECFGDefaultAddress
	IKECFGPreferDefaultAddress
	SALifetimeDelta
	FragmentSize

Логику используемого механизма IKE-ретрансмиссий смотрите в разделе «Обработка пакетов – ретрансмиссии».

Параметры с префиксом Blacklog задают поведение механизма так называемого “черного списка”. "Черный список" предназначен для защиты от DoS-атак (Denial of Service –отказ от обслуживания). "Черный список" минимизирует обработку IKE-пакетов от партнеров, находящихся в "черном списке".

Атрибут DefaultPort

Атрибут DefaultPort устанавливает порт для протокола IKE, который будет использован по умолчанию. Данная настройка не меняет порт, который используется для NAT traversal.

Синтаксис	DefaultPort = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..65535
Значение по умолчанию	500

Атрибут LocalPort

Атрибут LocalPort устанавливает локальный порт, используемый протоколом IKE.

Синтаксис	LocalPort = ЦЕЛОЕ32

Значение	Целое число из диапазона 1..65535. Если укзано значение 0, выбирается свободный порт по алгоритму, реализованному в операционной системе.
Значение по умолчанию	500

Атрибут NATTLocalPort

Атрибут NATTLocalPort устанавливает локальный порт для NAT Traversal, используемый протоколами IKE и IPsec.

Синтаксис	NATTLocalPort = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..65535. Если указано значение 0, выбирается свободный порт по алгоритму, реализованному в операционной системе.
Значение по умолчанию	4500

Атрибут SendRetries

Атрибут SendRetries устанавливает число попыток посылки IKE-пакетов партнеру.

Синтаксис	SendRetries = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..30
Значение по умолчанию	5

Атрибут RetryTimeBase

Атрибут RetryTimeBase позволяет установить начальный интервал в секундах между повторными попытками посылки IKE-пакетов партнеру. Если ответ не получен в течение начального интервала, то запрос посылается повторно и интервал между повторными попытками увеличивается в два раза. Этот интервал увеличивается в два раза до тех пор, пока:

- не будет получен ответ или
- значение интервала RetryTimeBase не достигнет значения RetryTimeMax (повторные попытки будут продолжаться с интервалом RetryTimeMax) и количество попыток не достигнет значения SendRetries.

Синтаксис	RetryTimeBase = ЦЕЛОЕ32
Значение	Целое число из диапазона 1...5
Значение по умолчанию	1

Атрибут RetryTimeMax

Атрибут RetryTimeMax позволяет установить максимальный интервал в секундах между повторными попытками посылки IKE-пакетов партнеру. Если выставленное значение этого атрибута меньше, чем RetryTimeBase, то при загрузке конфигурации атрибуту RetryTimeMax присваивается значение RetryTimeBase.

Синтаксис	RetryTimeMax = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..60
Значение по умолчанию	30

Атрибут SessionTimeMax

Атрибут SessionTimeMax ограничивает время (в секундах) на каждую сессию IKE.

Синтаксис	SessionTimeMax = ЦЕЛОЕ32
Значение	Целое число из диапазона 10..300
Значение по умолчанию	60

Атрибут InitiatorSessionsMax

Атрибут InitiatorSessionsMax устанавливает максимально допустимое количество одновременно инициируемых IKE-сессий для всех партнёров.

Синтаксис	InitiatorSessionsMax = ЦЕЛОЕ32
Значение	число из диапазона 1..10000
Значение по умолчанию	30

Атрибут ResponderSessionsMax

Атрибут ResponderSessionsMax определяет максимально допустимое количество одновременных IKE-обменов, проводимых VPN-устройством со всеми партнерами в качестве ответчика. Если локальное устройство имеет указанное количество незавершенных IKE-обменов в роли ответчика, то все входящие ISAKMP-пакеты, требующие установления новых обменов, игнорируются (без оповещения партнера).

Синтаксис	ResponderSessionsMax = ЦЕЛОЕ32
Значение	Целое число из диапозона 1..10000
Значение по умолчанию	20

Атрибут BlacklogSessionsMax

" BlacklogSessionsMax устанавливает начальное число разрешенных одновременных IKE обменов, инициируемых одним партнером[5], только что попавшим в "черный список". При каждом следующем неудачном завершении IKE обмена число разрешенных одновременных IKE обменов для данного партнера снижается вдвое с округлением в меньшую сторону, вплоть до значения, устанавливаемого параметром `BlacklogSessionsMin`.

[5]В данном случае партнер идентифицируется по паре ip:port. Пока партнер не аутентифицирован (т.е. с таким партнером на данный момент нет ни одного ISAKMP-соединения – SA), допустимое количество IKE-обменов может снижаться в зависимости от того, насколько успешно завершаются IKE-обмены с этим партнером.

Примечание: как только партнер заносится в "черный список", для него текущее значение разрешенных одновременно проводимых IKE обменов не только начинает уменьшаться в два раза после каждого неуспешного завершения обмена, но и увеличиваться на единицу по истечении каждого интервала времени BlacklogRelaxTime (описанного далее).

Синтаксис	BlacklogSessionsMax = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..(2^{32}-1). Если значение равно 0, то "черный список" не используется. Если значение BlacklogSessionsMax больше или равно ResponderSessionsMax, то атрибуту BlacklogSessionsMax присваивается значение ResponderSessionsMax
Значение по умолчанию	16

Атрибут BlacklogSessionsMin

Атрибут BlacklogSessionsMin позволяет установить минимальное число разрешенных одновременных IKE обменов, инициируемых неаутентифицированным партнером.

Синтаксис	BlacklogSessionsMin = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..(2^{32}-1). Если это значение равно, либо превышает BlacklogSessionsMax, то число разрешенных одновременных IKE обменов, инициируемых неаутентифицированным партнером, не снижается (т.е. ”черный список“ отключен)[6]. Если значение равно 0, то для партнера, поведение которого привело к понижению числа разрешенных инициируемых им одновременных IKE обменов до значения BlacklogSessionsMin, игнорируется весь IKE-трафик, а все имеющиеся с ним недостроенные IKE-сессии уничтожаются (ситуация “Access denied”).
Значение по умолчанию	0

Атрибут BlacklogSilentSessions

Атрибут BlacklogSilentSessions позволяет установить число активных обменов, инициированных неаутентифицированным партнером, по достижении которого VPN-устройство перестает информировать партнера о причине неуспешного завершения инициированного им IKE-обмена.

Синтаксис	BlacklogSilentSessions = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..(2^{32}-1). Если это значение больше, чем BlacklogSessionsMax, то инициатор не ограничивается в таких оповещениях. Если значение равно 0 либо 1, то неаутентифицированный партнер никогда не оповещается о причинах ошибки инициированного им обмена.

[6] При загрузке конфигурации с *отключенным* «черным списком» вся статистическая информация о «плохих» партнерах сбрасывается. Если же «черный список» *включен*, то к уже имеющейся накопленной статистике применяются новые параметры настроек «черного списка».

Значение по умолчанию	4

Атрибут BlacklogRelaxTime

Атрибут BlacklogRelaxTime устанавливает интервал времени (в секундах) релаксации "черного списка":

- За указанный период времени число разрешенных одновременных IKE обменов для каждого партнера, находящегося в "черном списке", увеличивается на единицу. По истечении следующего такого же интервала времени, текущие значения разрешенных одновременно проводимых IKE обменов для каждого партнера опять увеличивается на единицу и т.д. Этот интервал времени отсчитывается с момента последней загрузки конфигурации.
- Как только текущее значение разрешенных одновременно проводимых партнером IKE обменов начинает превышать значение BlacklogSessionsMax, такой партнер исключается из "черного списка".

Синтаксис	BlacklogRelaxTime = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..(2^{32}-1). 0 – бесконечное время (партнер попадает в "черный список" навсегда).
Значение по умолчанию	120
Примечание	Помимо механизма релаксации, партнер также может быть иключен из "черного списка" в следующих случаях: при перезапуске сервиса при загрузке конфигурации с отключенным "черным списком" (с атрибутом BlacklogSessionsMax = 0) при инициации IKE обмена со стороны локального VPN устройства с целью установления ISAKMP (IPsec) соединения[7] если партнеру удалось установить ISAKMP (IPsec) соединение с локальным VPN устройством, и тем самым партнер был успешно аутентифицирован.

Атрибут IKECFGDefaultAddress

Атрибут IKECFGDefaultAddress задает IP-адрес, который запрашивается у ПК «С-Терра Шлюз» по IKECFG.

Синтаксис	IKECFGDefaultAddress = IP-адрес
Значение	IP-адрес. Значение 0.0.0.0, означает, что клиент запрашивает произвольный адрес из пула.
Значение по умолчанию	0.0.0.0

[7] В данном случае считается, что локальное VPN устройство потенциально доверяет партнеру, с которым оно хочет установить соединение, и информация, накопленная в "черном списке", для такого партнера сбрасывается.

Атрибут IKECFGPreferDefaultAddress

Атрибут IKECFGPreferDefaultAddress задает режим использования IP-адреса, указанного атрибутом IKECFGDefaultAddress. В случае, если IKECFGDefaultAddress противоречит сетевой конфигурации (например, конфликтует с локальными адресами), он не будет использоваться, вне зависимости от значения IKECFGPreferDefaultAddress.

Синтаксис	IKECFGPreferDefaultAddress = **TRUE | FALSE**
Значение	TRUE – при старте vpnsvc использует IKECFGDefaultAddress в запросе адреса по IKECFG (даже, если IKECFGDefaultAddress нулевой). При перезагрузке или изменении IPsec-конфигурации, когда происходит удаление всех IKE SA, IKECFGDefaultAddress тоже будет использован как начальный. FALSE – отсылается
Значение по умолчанию	FALSE

Атрибут SALifetimeDelta

Атрибут SALifetimeDelta позволяет установить случайный разброс во времени жизни IKE и IPsec SA. Этот атрибут может быть полезен в случае массового пересоздания SA.

Синтаксис	SALifetimeDelta = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..50 Значение – максимальный процент, на который может быть уменьшено время действия SA (LifetimeSeconds). Реальное значение определяется случайным образом от 0 до этого максимума.
Значение по умолчанию	0 – отключает механизм случайного изменения времени жизни IKE SA и IPsec SA.
Примечание	Для респондера значение SALifetimeDelta будет использовано только при условии, если инициатор предлагает значение большее или равное локальному параметру LifetimeSeconds. Если локальное значение IKETransform.LifetimeSeconds равно 0, то для данного правила SALifetimeDelta не используется.

Атрибут FragmentSize

Атрибут FragmentSize управляет функциональностью фрагментирования IKE-пакетов. Этот атрибут рекомендуется использовать в случае массового пересоздания SA.

Синтаксис	FragmentSize = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..65535. Значение – максимальный размер результирующего IP-пакета[8] в байтах. Значение 0 отключает функциональность фрагментирования

[8] Следует учитывать, что операционная система сама устанавливает длину ip-заголовка, что может приводить к фактическому уменьшению длины ip-пакета с IKE-фрагментом на величину до 44 байт (максимально допустимый размер ip-заголовка – 64 байта, наиболее часто используемый – 20 байт).

	IKE-пакетов. Партнёр о поддержке фрагментирования IKE-пакетов не оповещается, отсылаемые IKE-пакеты не фрагментируются, принимаемые фрагменты не собираются. Ненулевое заданное значение может корректироваться в большую сторону таким образом, чтобы максимально возможный ISAKMP-пакет длиной 64Kb мог быть разбит на 255 фрагментов.
Значение по умолчанию	576

11.5.1 Обработка пакетов – ретрансмиссии

1. Используемый механизм IKE-ретрансмиссий находится в общей концепции, согласно которой инициатор, исходя из наличия собственных ресурсов, проявляет настойчивость и добивается чего-то от ответчика, а ответчик, во первых, не доверяет инициатору насколько это возможно, во-вторых, по-максимуму бережет собственные ресурсы.

    - Инициатор, в большинстве случаев, являясь активной стороной, посылает очередной пакет IKE-обмена и затем перепосылает его (в соответствии с настройками ретрансмиссий – атрибуты `SendRetries`, `RetryTimeBase` и `RetryTimeMax`) до тех пор, пока не получит ответный пакет от ответчика.

      Таким образом, инициатор выполняет работу за двоих:

      - если исходящий от инициатора пакет не дошел до ответчика, то ответчик его не обработает и, соответственно, никак не ответит инициатору. Но исходящий пакет инициатором может быть перепослан (возможно, с n-ой попытки), ответчик его получит, обработает и отошлёт ответ
      - если же проблема возникла на обратном пути (т.е. пакет от ответчика потерялся на пути к инициатору), то для инициатора эта ситуация детектируется точно так же, как и первая – то есть инициатор ответного пакета ждал, но за отведенный timeout так и не дождался. Тогда инициатор перепосылает свой последний исходящий пакет, ответчик снова его получает, распознает его как совпадающий с последним пакетом от инициатора, т.е. ретрансмиссию, и в ответ перепосылает свой последний пакет.

2. События для перепосылки:

    - для стороны, выполняющей активную роль в ретрансмиссиях, событием для перепосылки своего последнего пакета является таймер и отсутствие ответа от партнера
    - для пассивной стороны событием для перепосылки своего последнего пакета является получение ретрансмиссии от партнера.

3. В сценариях IKE, в которых ответчик обрабатывает последний пакет (Aggressive Mode и Quick Mode без поддержки Commit Bit), ответчик становится активной стороной при ожидании последнего пакета обмена. В этих случаях инициатор уже не может выполнять активную роль, так как он в любом случае по сценарию не получает ответный пакет.

11.6 Структура SNMPPollSettings

Структура задает настройки для выдачи информации по запросу SNMP-менеджера. В конфигурации должна быть только одна структура данного типа. Этой структуре имя не присваивается.

Имя структуры	SNMPPollSettings
Атрибуты	LocalIPAddress
	Port
	ReadCommunity
	SysLocation
	SysContact

Атрибут LocalIPAddress

Атрибут LocalIPAddress задаёт список локальных IPv4-адресов, на который можно получать запросы от SNMP-менеджера. Указание IP-адреса 0.0.0.0 эквивалентно указанию константы ANY.

Синтаксис	LocalIPAddress = IP | **ANY**
Значение	IP – список локальных IP-адресов ANY – все локальные IP-адреса
Значение по умолчанию	ANY

Атрибут Port

Атрибут Port задаёт порт, на который можно получать SNMP-запросы.

Синтаксис	Port = ЦЕЛОЕ32
Значение	целое число из диапазона 1..65535
Значение по умолчанию	161

Атрибут ReadCommunity

Атрибут ReadCommunity играет роль пароля при аутентификации сообщений SNMP и разрешает SNMP-менеджеру чтение статистики из базы управления SNMP-агента.

Синтаксис	ReadCommunity = СТРОКА
Значение	произвольный формат
Значение по умолчанию	не существует, атрибут обязательный

Атрибут SysLocation

Атрибут SysLocation содержит информацию о физическом расположении SNMP-агента.

Синтаксис	SysLocation = СТРОКА
Значение	произвольный формат, например "Building 3/Room 214"
Значение по умолчанию	пустая строка

Атрибут SysContact

Атрибут SysContact содержит информацию о контактном лице, ответственном за работу SNMP-агента.

Синтаксис	SysContact = СТРОКА
Значение	произвольный формат, например e-mail, телефон и т.д
Значение по умолчанию	пустая строка.

11.7 Структура SNMPTrapSettings

Структура задает настройки для выдачи агентом сообщений менеджеру о возникшем прерывании в виде SNMP-трапов. В конфигурации должна быть только одна структура данного типа. Этой структуре имя не присваивается. При отсутствии этой структуры трап-сообщения не высылаются.

Имя структуры	SNMPTrapSettings
Атрибуты	Receivers

Атрибут Receivers

Атрибут Receivers задаёт список получателей SNMP-трапов и дополнительные настройки.

Синтаксис	Receivers* = `TrapReceiver`
Значение по умолчанию	не существует, атрибут обязательный

11.8 Структура TrapReceiver

Структура описывает одного получателя SNMP-трапов и дополнительные настройки для трапов, отсылаемых на него.

Имя структуры	TrapReceiver
Атрибуты	IPAddress
	Port
	Community
	Version
	LocalIPAddress

Атрибут IPAddress

Атрибут IPAddress описывает IP-адрес получателя SNMP-трапов.

Синтаксис	IPAddress = IP
Значение	IP-адрес
Значение по умолчанию	не существует, атрибут обязательный.

Атрибут Port

Атрибут Port задает UDP-порт, на который SNMP-менеджеру будут высылаться трап-сообщения.

Синтаксис	Port = ЦЕЛОЕ32
Значение	целое число из диапазона 1..65535
Значение по умолчанию	162

Атрибут Community

Атрибут Community играет роль идентификатора отправителя трап-сообщения.

Синтаксис	Community = СТРОКА
Значение	произвольный формат
Значение по умолчанию	не существует, атрибут обязательный

Атрибут Version

Атрибут Version указывает версию SNMP, в которой формируются трап-сообщения.

Синтаксис	Version = **V1 | V2**
Значение	V1 – SNMP версии 1 V2C – SNMP версии
Значение по умолчанию	V1

Атрибут LocalIPAddress

Атрибут LocalIPAddress задает IP-адрес, с которого будут отправляться трап-сообщения. Можно вместо IP-адреса указать имя сетевого интерфейса.

Синтаксис	LocalIPAddress = IP | LogicalName
Значение	LogicalName – имя сетевого интерфейса, должно совпадать с одним из имен LogicalName в структуре `NetworkInterface`. Если указанному LogicalName соответствует несколько сетевых интерфейсов или адресов, то будет использован один адрес[9] IP-адрес интерфейса. Если указано значение 0.0.0.0, адрес будет выбирать ОС в зависимости от адреса назначения.
Значение по умолчанию	0.0.0.0

[9] Первый адрес первого подходящего интерфейса в соответствии с порядком выдачи интерфейсов и адресов библиотекой ni.

11.9 Структура RoutingTable

Структура RoutingTable описывает маршруты, которые добавляются в системную таблицу маршрутизации. Если при добавлении маршрута в системную таблицу возникает ошибка, это не прерывает загрузку LSP. Соответствующее предупреждение передается через систему протоколирования.

При отгрузке конфигурации маршруты из системной таблицы маршрутизации будут удалены.

Предполагается, что пользователь не создает и не удаляет маршруты с теми же адресами назначения (Destination), что указаны в LSP.

Если при добавлении маршрута в системную таблицу возникает ошибка, тем не менее, загрузка LSP продолжается, а соответствующее предупреждение передается через систему протоколирования.

В конфигурации допускается только один экземпляр этой структуры. Этой структуре не может быть присвоено имя.

Имя структуры RoutingTable

Атрибуты Routes

Атрибут Routes

Атрибут Routes содержит список записей для добавления в таблицу маршрутизации.

Синтаксис	Routes* = `Route`
Значение по умолчанию	не существует, атрибут обязательный.

11.10 Структура Route

Структура Route описывает одну запись (маршрут) в таблице маршрутизации.

Имя структуры	Route
Атрибуты	Destination
	Gateway
	NetworkInterface

Атрибут Destination

Атрибут Destination задает адрес назначения (получателя) пакета.

Синтаксис	Destination = IP | IP/ЦЕЛОЕ32
Значение	IP – адрес IP/ЦЕЛОЕ32 – IP-адрес с маской подсети Для указания маршрута, который будет использоваться по умолчанию, IP-адрес и маска подсети должны иметь значение 0.0.0.0/ 0. Маршрут по умолчанию – маршрут, по которому будет отправлен пакет, если IP-адрес назначения, указанный в заголовке пакета, не совпадает ни с одним адресом назначения в таблице маршрутизации.
Значение по умолчанию	отсутствует, атрибут обязательный.

Атрибут Gateway

Атрибут Gateway задает IP-адрес устройства, на который нужно передать пакет для продвижения его к получателю пакета. Атрибут Gateway должен отсутствовать при наличии атрибута `NetworkInterface`.

Синтаксис	Gateway = IP
Значение	IP-адрес
Значение по умолчанию	используется значение из атрибута NetworkInterface.

Атрибут NetworkInterface

Атрибут NetworkInterface указывает имя выходного интерфейса, на который нужно передать пакет для продвижения его к получателю пакета. Атрибут NetworkInterface должен отсутствовать при наличии атрибута `Gateway`.

Синтаксис	NetworkInterface = СТРОКА
Значение	имя интерфейса
Значение по умолчанию	используется значение из атрибута Gateway.

11.11 Структура IPsecAction

Структура IPsecAction задает правило создания контекста соединения для протоколов семейства IPsec. Этой структуре может быть присвоено имя.

Имя структуры	IPsecAction
Атрибуты	TunnelingParameters
	ShuffleTunnelEntries
	CryptoContextsPerIPSecSA
	GroupID
	ContainedProposals
	IKERule
	NoPathMTUDiscovery
	MTU
	NoSmoothRekeying
	InputFilter
	OutputFilter
	PersistentConnection

Атрибут TunnelingParameters

Атрибут TunnelingParameters описывает параметры внешнего IP-заголовка пакета, который добавляется в туннельном режиме IPsec. Если в TunnelingParameters указано более одного элемента, то элементы используются как альтернативные партнеры. Если не удалось установить IPsec-туннель с партнером, то производиться попытка установить туннель со следующим партнером в списке, и так далее до окончания списка.

Синтаксис	TunnelingParameters* = `TunnelEntry`
Значение по умолчанию	используется транспортный режим 0
Предупреждение	если между партнерами обнаружен NAT, то создавать соединение в транспортном режиме нельзя.

Атрибут ShuffleTunnelEntries

Атрибут ShuffleTunnelEntries задает порядок применения структур `TunnelEntry` в атрибуте TunnelingParameters. Атрибут ShuffleTunnelEntries игнорируется, если атрибут TunnelingParameters не задан.

Синтаксис	ShuffleTunnelEntries = **TRUE | FALSE**
Значение	TRUE – при загрузке конфигурации туннели в списке TunnelingParameters перемешиваются случайным образом FALSE – при загрузке конфигурации туннели в списке TunnelingParameters применяются в порядке перечисления
Значение по умолчанию	FALSE

Атрибут CryptoContextsPerIPSecSA

Атрибут CryptoContextsPerIPSecSA задает количество открываемых криптографических контекстов на один IPsec SA, созданный по этому правилу IPsecAction. Наличие нескольких криптографических контекстов позволяет распараллелить обработку пакетов одним IPsec SA.

Синтаксис	CryptoContextsPerIPSecSA = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..128
Значение по умолчанию	значение берется из файла agent.ini (параметр DefaultCryptoContextsPerIPSecSA).

Атрибут IKERule

Атрибут IKERule является ссылкой на IKE правило, под защитой которого создается IPsec SA.

Синтаксис	IKERule = `IKERule`
Значение по умолчанию	не существует, атрибут обязательный.

Атрибут GroupID

Атрибут GroupID задает параметры получения ключевого материала. Используется алгоритм Диффи-Хеллмана либо VKO GOST R 34.10-2001 [RFC4357]. Параметры задаются в виде списка. Если список не пуст, то для инициатора соединения ключевой материал всегда задаётся согласно первому компоненту списка. Для ответчика присланное предложение инициатора сравнивается последовательно со всеми элементами своего списка.

Синтаксис	GroupID = **VKO_1B, MODP_768, MODP_1024, MODP_1536**, **NO_PFS**
Значение	VKO_1B – используется алгоритм VKO GOST R 34.10-2001, длина ключа 256 бит MODP_768 – длина ключа 768 бит – группа 1 MODP_1024 – длина ключа 1024 бита – группа 2 MODP_1536 – длина ключа 1536 бит – группа 5 NO_PFS – обмен ключами во второй фазе IKE не используется
Значение по умолчанию	ключевой материал заимствуется из первой фазы IKE.

Атрибут ContainedProposals

Атрибут ContainedProposals задает варианты совместного применения IPsec-протоколов (AH и ESP). Число вариантов не ограничено. Варианты задаются с использованием структур AHProposal и ESPProposal. Структуры AHProposal и ESPProposal могут группироваться, позволяя обрабатывать трафик комбинацией протоколов AH и ESP.

Атрибут ContainedProposals содержит список единичных структур AHProposal и ESPProposal или их пар в порядке убывания приоритета.

Синтаксис	ContainedProposals *= Proposal Proposal *= (`AHProposal [,ESPProposal]`) | `ESPProposal`

Значение	Число элементов списка неограничено. Все элементы списка должны быть различными. Один элемент списка содержит до двух преобразований с различными протоколами. Если элемент списка содержит AHProposal и ESPProposal, то они должны следовать в указанном порядке. Инициатор соединения посылает партнеру все варианты параметров защиты соединения, указанные в атрибуте ContainedProposals, с целью их согласования во время второй фазы IKE –сессии. Ответная сторона присланные предложения инициатора соединения последовательно сравнивает с каждым элементом своего списка предложений и выбирает первое совпавшее. При переборе более приоритетным является список на стороне ответчика. Параметры преобразований и комбинация протоколов AH и ESP определяют качество защиты соединения. Запись `(ah1, esp1),(esp2),(ah3)` означает, что рассматриваются варианты контекстов: либо связка `(ah1, esp1)`, либо `esp2`, либо `ah3`.
Значение по умолчанию	не существует, атрибут обязательный.

Пример

```
ContainedProposals *=
(IPsec_ah_md5, IPsec_esp_des3),(IPsec_ah_md5, IPsec_esp_idea)
(* (AH(MD5) и ESP(DES3) или AH(MD5) и ESP(IDEA) *)

ContainedProposals *=
(IPsec_ah_md5, IPsec_esp_des3),(IPsec_ah_md5)
  (* (AH(MD5) и ESP(DES3) или  AH(MD5) *)

ESPProposal IPsec_esp_idea(
  Transform *= ESPTransform(
    CipherAlg = "IDEA-CBC"
  )
)
AHProposal IPsec_ah_md5(
  Transform *= AHTransform(
    IntegrityAlg* = "MD5-H96-HMAC"
  )
)
ESPProposal IPsec_esp_des3(
  Transform *= ESPTransform(
    CipherAlg = "DES3-K168-CBC"
  ))
```

Атрибут NoSmoothRekeying

Атрибут NoSmoothRekeying задает режим заблаговременной смены ключевого материала.

Синтаксис	NoSmoothRekeying = **TRUE | FALSE**
Значение	TRUE – заблаговременная смена ключевого материала (rekeying) не проводится. При отсутствии подходящего IPsec соединения, новый IPsec SA создаётся только по запросу из

	ядра – при наличии исходящего IP-пакета, либо по инициативе партнера. В результате во время создания нового IPsec SA IP-трафик приостанавливается, а при интенсивном трафике возможна потеря пакетов. FALSE – заблаговременно, незадолго до окончания действия IPsec соединения, на его основе (с теми же параметрами) проводится IKE-сессия (Quick Mode) по созданию нового IPsec SA (rekeying). Rekeying не проводится, если за время существования старого SA под его защитой не было никакого трафика.[10].
Значение по умолчанию	FALSE

NoPathMTUDiscovery

Этот атрибут отключает алгоритм "Path MTU Discovery" (выявление максимального размера блока передачи данных, проходящего на всем пути от отправителя к получателю без фрагментации) для IPsec SA, создаваемых по данному правилу.

Синтаксис	NoPathMTUDiscovery = **TRUE | FALSE**
Значение	FALSE – производится обработка ICMP-сообщений типа destination unreachable/fragmentation needed, приходящих в ответ на IPsec-пакеты. На основе этих сообщений вычисляется эффективное значение MTU трассы (максимальный размер пакета, проходящего по всему каналу без фрагментации). TRUE – ICMP-сообщения не обрабатываются, значение MTU вычисляется только из локальной конфигурации.
Значение по умолчанию	FALSE

Атрибут MTU

Этот атрибут задает значение MTU для IPsec SA, создаваемых по данному правилу.

Значение MTU используется только для исходящих пакетов и для последнего SA из примененных к пакету (в случае вложенного IPsec, значение для внутреннего SA игнорируется).

Если NoPathMTUDiscovery=FALSE, то указанное IPsecAction.MTU может быть скорректировано в меньшую сторону при вычислении MTU трассы.

Синтаксис	MTU = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..65535 Значение интерпретируется следующим образом: если пакет подвергается повторной маршутизации (TunnelEntry.ReRoute = TRUE или пакет отправляется с IKECFG-интерфейса): если DF-бит в пакете выставлен, то значение MTU интерфейса не учитывается, а значение IPsecAction.MTU рассматривается как значение MTU интерфейса если DF-бит сброшен – IPsecAction.MTU не используется если пакет отправляется без повторной маршрутизации, то

[10]Для проведения rekeying-а необходимо, чтобы время жизни обновляемого соединения было существенно больше времени, которое отводится на проведение IKE-сессии.

	выбирается минимальное из значений – MTU интерфейса и MTU в IPsecAction.
Значение по умолчанию	0 – MTU определяется автоматически.

Атрибут InputFilter

Атрибут InputFilter задает дополнительные правила фильтрации, присоединяемые к IPsec SA. InputFilter применяется к входящим пакетам после декапсуляции.

Если IPsecAction строит более одного SA для каждого направления, фильтрация все равно производится один раз. То есть, в комбинации ESP+AH правила фильтрации будут применены к ESP SA.

В случае вложенного IPsec, когда к пакету применяются SA, создаваемые по разным IPsecAction, пакет пройдет все InputFilter от каждого IPsecAction.

Синтаксис	InputFilter = `FilterChain`
Значение по умолчанию	дополнительная фильтрация входящих пакетов не производится.

Атрибут OutputFilter

Атрибут OutputFilter задает дополнительные правила фильтрации, присоединяемые к IPsec SA. OutputFilter применяется к исходящим пакетам до инкапсуляции.

Если IPsecAction строит более одного SA для каждого направления, фильтрация все равно производится один раз. То есть, в комбинации ESP+AH правила фильтрации будут применены к ESP SA.

В случае вложенного IPsec, когда к пакету применяются SA, создаваемые по разным IPsecAction, пакет пройдет все OutputFilter от каждого IPsecAction.

Синтаксис	OutputFilter = `FilterChain`
Значение по умолчанию	дополнительная фильтрация исходящих пакетов не производится.

Атрибут PersistentConnection

Атрибут PersistentConnection задает построение IKE SA и IPsec SA сразу после загрузки LSP. Устанавливать этот атрибут следует в случае работы Клиента через IKECFG-интерфейс. В конфигурации допускается только один экземпляр IPsecAction с PersistentConnection=TRUE.

Синтаксис	InputFilter = **TRUE | FALSE**
Значение	TRUE – сразу после загрузки LSP происходит попытка построить IKE и IPsec SA, используя данную структуру IPsecAction. IPsec SA строятся по каждому из фильтров, к которым IPsecAction привязана. Если в фильтре есть несколько диапазонов адресов, SA строится только для первого. Если ни одного IPsec SA не построилось, попытки построить SA повторяются сначала (см. также PersistentConnectionRetryDelay). Если включен автоматический режим смены ключей (NoSmoothRekeying=FALSE), процесс обновления SA не

	прерывается при отсутствии трафика. В фильтрах, к которым привязан IPsecAction с выставленным PersistentConnection, не допускается указание портов, протоколов и SourceIP. FALSE – запрещает получение адресов по IKECFG по данному правилу
Значение по умолчанию	FALSE

11.12 Структура TunnelEntry

Структура TunnelEntry описывает параметры внешнего IP-заголовка пакета при использовании туннельного режима IPsec.

Имя структуры	TunnelEntry
Атрибуты	PeerIPAddress
	LocalIPAddress
	DFHandling
	ReRoute
	Assemble

Атрибут PeerIPAddress

Атрибут PeerIPAddress описывает туннельный адрес. Этот адрес используется для двух целей – адрес получателя во внешнем IP-заголовке и адрес IKE-партнера, если последний не задан явно.

Синтаксис	PeerIPAddress = IP
Значение по умолчанию	если туннельный адрес используется как адрес получателя во внешнем IP заголовке, то: для исходящего пакета берется адрес IKE партнера; если туннельный адрес используется как адрес IKE партнера: для исходящего пакета берется адрес из IP пакетов, вызвавших создание соединения для входящего пакета – принимается любой адрес

Атрибут LocalIPAddress

Атрибут LocalIPAddress описывает туннельный адрес локального VPN-устройства.

Синтаксис	LocalIPAddress = IP
Значение по умолчанию	для исходящего пакета – любой из адресов сетевого интерфейса, с которого отправляется пакет.

Атрибут DFHandling

Атрибут DFHandling задает алгоритм формирования DF (Don't Fragment) бита внешнего IP-заголовка для туннельного режима IPsec.

Синтаксис	DFHandling = **COPY | SET | CLEAR**
Значение	COPY – копировать DF бит из внутреннего заголовка во внешний заголовок SET – всегда устанавливать DF бит внешнего заголовка в 1 CLEAR – всегда сбрасывать DF бит внешнего заголовка в 0.
Значение по умолчанию	COPY

Атрибут ReRoute

Атрибут ReRoute указывает, что пакет будет подвергаться повторной маршрутизации. При использовании повторной маршрутизации может происходить повторная обработка пакета IPsec-драйвером, LSP должна создаваться с учетом этого. То есть, чтобы IPsec-пакеты с локального адреса пропускались при втором проходе. Указывать ReRoute имеет смысл для SA, заменяющих адрес назначения. Если по ходу обработки пакета адрес назначения не изменился, флаг ReRoute игнорируется.

Синтаксис	ReRoute = **TRUE | FALSE**
Значение	TRUE – исходящий пакет после цикла обработки не отправляется в драйвер сетевого интерфейса, а направляется в IP-драйвер для повторной маршрутизации. Такой пакет может попасть на повторную обработку IPsec драйвером, так что правила фильтрации должны учитывать и пропускать такие пакеты. FALSE – указывает, что пакет не будет подвергаться повторной маршрутизации
Значение по умолчанию	FALSE

Атрибут Assemble

Атрибут Assemble указывает, что пакет будет собран из IP-фрагментов перед заворачиванием в IPsec. В транспортном режиме IPsec сборка пакетов перед инкапсуляцией производится всегда.

Синтаксис	Assemble = **TRUE | FALSE**
Значение	TRUE – означает, что пакет будет собран из IP-фрагментов перед заворачиванием в IPsec. Рекомендуеся устанавливать при работе по защищенному соединению с предыдущими версиями Продукта. FALSE – указывает, что пакет не будет подвергаться сборке.
Значение по умолчанию	FALSE

Пример структуры IPsecAction

```
IPsecAction tunnel_ipsec_des_md5_action(
  TunnelingParameters *= TunnelEntry(
    PeerIPAddress = 192.168.2.1
    DFHandling = CLEAR
  )

  IKERule = ike_r
  GroupID = MODP_768
  ContainedProposals *= (ipsec_ah_md5,  ipsec_esp_des),
(ipsec_esp_des_md5)
)

ESPProposal ipsec_esp_des(
  Transform *= ESPTransform(
    CipherAlg = "DES-CBC"
  )
)
```



```
AHProposal ipsec_ah_md5(
  Transform *= AHTransform(
    IntegrityAlg = "MD5-H96-HMAC"
  )
)

ESPProposal ipsec_esp_des_md5(
  Transform *= ESPTransform(
    CipherAlg = "DES-CBC"
    IntegrityAlg = "MD5-H96-HMAC"
  )
)
```

11.13 Структуры AHProposal и ESPProposal

Структура AHProposal задает список криптографических преобразований (transforms) протокола AH в порядке убывания приоритета, которые допускаются для обработки трафика. Трафик – количество килобайт данных, обработанных данным контекстом.

Структура ESPProposal определяет список преобразований (transforms) протокола ESP в порядке убывания приоритета, которые допускаются для обработки специфицированного трафика.

Имя структуры	AHProposal
Атрибуты	Transform

Имя структуры	ESPProposal
Атрибуты	Transform

Атрибут Transform

Атрибут Transform задает список возможных групп параметров протокола AH (для структуры AHProposal) или ESP (для структуры ESPProposal), необходимых для создания SA, расположенных в порядке убывания их приоритета.

Синтаксис	Transform *= `AHTransform` # для структуры AHProposal Transform *= `ESPTransform` # для структуры ESPProposal Должен присутствовать хотя бы один трансформ.
Значение по умолчанию	не существует, атрибут обязательный.

11.14 Структура AHTransform

Структура AHTransform задает параметры контекста (SA) для протокола AH.

Неявные ограничения на количество обработанного трафика в пакетах, в байтах или на количество ошибок при проверке целостности пакетов могут содержаться в реализации конкретных криптографических алгоритмов.

Рекомендуется указывать такое время SA жизни в секундах, что бы в основном удаление IPsec SA происходило по времени, а ограничение на объем трафика выбирать как дополнительную меру.

Имя	AHTransform
Атрибуты	LifetimeSeconds
	LifetimeKilobytes
	IntegrityAlg

Атрибут LifetimeSeconds

Атрибут LifetimeSeconds задает максимальное время существования контекста (SA) AH (в секундах).[11]

Синтаксис	LifetimeSeconds = ЦЕЛОЕ32
Значение	число из диапазона $0..2^{32}-1$
Значение по умолчанию	28800 (8 часов).

Атрибут LifetimeKilobytes

Атрибут LifetimeKilobytes задает максимальное ограничение по трафику на действие контекста в килобайтах. Трафик – количество килобайт данных, обработанных данным контекстом.[12]

Синтаксис	LifetimeKilobytes = ЦЕЛОЕ32
Значение	число из диапазона $0..2^{32}-1$
Значение по умолчанию	нет ограничений на действие SA. (Замечание: количество IPsec пакетов, обработанных по одному IPsec SA, всегда ограничивается максимальным значением sequence number – $2^{32}-1$ пакетов. При превышении максимального значения sequence number будет запрошена смена ключей, как это делается в случае превышения ограничения в байтах.)
Примечание	Если в атрибуте IntegrityAlg задается алгоритм G2814789CPRO1-K256-MAC-255, то в этом случае максимальное допустимое значение LifetimeKilobytes – 4032 Кб. При превышении указанного значения, в журнал протоколирования будет выдано сообщение, что в созданном IPsec SA ограничение по трафику не соответствует

[11] В случае использования связок (AH+ESP) в качестве элементов списка ContainedProposals структуры IPsecAction, соответствующие значения в AH- и ESP- трансформах уравниваются в меньшую сторону.

[12] В случае использования связок (AH+ESP) в качестве элементов списка ContainedProposals структуры IPsecAction, соответствующие значения в AH- и ESP- трансформах уравниваются в меньшую сторону.

	допустимому ограничению для используемого криптографического алгоритма: `"SA traffic limit exceeds limitations imposed by the cryptograghic algorithm".`

Атрибут IntegrityAlg

Атрибут IntegrityAlg задает алгоритм проверки целостности пакета в рамках создаваемого контекста.

Если же существует необходимость задать несколько алгоритмов (их комбинацию) проверки целостности, то используйте альтернативный подход: в атрибуте `Transform` структуры AHProposal укажите список структур AHTransform, а в каждой структуре AHTransform задайте только один алгоритм проверки целостности.

Синтаксис	IntegrityAlg = **"MD5-H96-KPDK"|"MD5-H96-HMAC"|"SHA1-H96-HMAC"|"GR341194CPRO1-H96-HMAC-254"|"G2814789CPRO1-K256-MAC-255"**
Значение	Возможные значения: "MD5-H96-KPDK" – Keyed MD5 "MD5-H96-HMAC" – HMAC MD5 (96 бит) "SHA1-H96-HMAC" – HMAC SHA-1 (96 бит) "GR341194CPRO1-H96-HMAC-254" – реализация ГОСТ Р 34.11-94 (96 бит) "G2814789CPRO1-K256-MAC-255" – реализация ГОСТ 28147-89 (в режиме выработки имитовставки).
Значение по умолчанию	не существует, атрибут обязательный

11.15 Структура ESPTransform

Структура ESPTransform задает параметры контекста (SA) для протокола ESP.

Неявные ограничения на количество обработанного трафика в пакетах, в байтах или на количество ошибок при проверке целостности пакетов могут содержаться в реализации конкретных криптографических алгоритмов.

Рекомендуется указывать такое время SA жизни в секундах, что бы в основном удаление IPsec SA происходило по времени, а ограничение на объем трафика выбирать как дополнительную меру.

Имя	ESPTransform
Атрибуты	LifetimeSeconds
	LifetimeKilobytes
	CipherAlg
	IntegrityAlg

Атрибут LifetimeSeconds

Атрибут LifetimeSeconds задает максимальное время существования контекста в секундах.[13]

Синтаксис	LifetimeSeconds = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..2^{32}-1.
Значение по умолчанию	28800 (8 часов)

Атрибут LifetimeKilobytes

Атрибут LifetimeKilobytes задает максимальное ограничение по трафику на действие контекста в килобайтах. Трафик – количество килобайт данных, обработанных данным контекстом.[14]

Синтаксис	LifetimeKilobytes = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..2^{32}-1
Значение по умолчанию	нет ограничений на действие SA. (Замечание: количество IPsec пакетов, обработанных по одному IPsec SA, всегда ограничивается максимальным значением sequence number – 2^{32}-1 пакетов. При превышении максимального значения sequence number будет запрошена смена ключей, как это делается в случае превышения ограничения в байтах.)
Примечание	Если используются алгоритмы G2814789CPRO1-K256-CBC-254, G2814789CPRO1-K256-MAC-65535 (атрибуты CipherAlg, IntegrityAlg), то в этом случае максимальное допустимое значение LifetimeKilobytes – 4032 Кб. При превышении указанного значения, в журнал протоколирования будет выдано сообщение, что в созданном IPsec SA ограничение по трафику не соответствует

[13] В случае использования связок (AH+ESP) в качестве элементов списка ContainedProposals структуры IPsecAction, соответствующие значения в AH- и ESP- трансформах уравниваются в меньшую сторону

[14] В случае использования связок (AH+ESP) в качестве элементов списка ContainedProposals структуры IPsecAction, соответствующие значения в AH- и ESP- трансформах уравниваются в меньшую сторону

	допустимому ограничению для используемого криптографического алгоритма: `"SA traffic limit exceeds limitations imposed by the cryptograghic algorithm".`

Атрибут CipherAlg

Атрибут CipherAlg задает алгоритм шифрования трафика в рамках создаваемого контекста.

Если же существует необходимость задать несколько алгоритмов шифрования, то используйте альтернативный подход: в атрибуте `Transform` структуры ESPProposal укажите список структур ESPTransform, а в каждой структуре ESPTransform задайте только один алгоритм шифрования.

Синтаксис	CipherAlg = **"NULL"|"DES-CBC_IV64"|"DES-CBC_IV32"|"DES-CBC"|"AES-K128-CBC"|"AES-K192-CBC"|"AES-K256-CBC"|"G2814789CPRO1-K256-CBC-254"|"G2814789CPRO1-K288-CNTMAC-253"**
Значение	Возможные значения: "NULL" – NULL (данные не шифруются) "DES-CBC_IV64" – DES в режиме CBC с явным IV длиной 64 бита "DES-CBC_IV32" – DES в режиме CBC с явным IV длиной 32 бита "DES-CBC" – DES в режиме CBC "DES3-K168-CBC" – DES3 в режиме CBC "IDEA-CBC" – IDEA в режиме CBC "AES-K128-CBC" – AES в режиме CBC с длиной ключа 128 "AES-K192-CBC" – AES в режиме CBC с длиной ключа 192 "AES-K256-CBC" – AES в режиме CBC с длиной ключа 256 "G2814789CPRO1-K256-CBC-254" – реализация ГОСТ 28147-89 в режиме CBC "G2814789CPRO1-K288-CNTMAC-253" – реализация ГОСТ 28147-89 в комбинированном режиме (гаммирование и вычисление имитовставки) в соответствии со спецификацией ESP_GOST-4M-IMIT («Техническая спецификация по использованию ГОСТ 28147-89 при шифровании вложений в протоколе IPSEC ESP»).
Значение по умолчанию	не существует, атрибут обязательный.

Атрибут IntegrityAlg

Атрибут IntegrityAlg задает алгоритм проверки целостности пакета в рамках создаваемого контекста.

Если же существует необходимость задать несколько алгоритмов проверки целостности (их комбинацию), то используйте альтернативный подход: в атрибуте Transform структуры ESPProposal укажите список структур ESPTransform, а в каждой структуре ESPTransform задайте только один алгоритм проверки целостности пакета.

Если атрибут CipherAlg имеет значение "NULL", то атрибут IntegrityAlg нужно указывать обязательно.

Синтаксис	IntegrityAlg = **"MD5-H96-KPDK"|"MD5-H96-HMAC"|"SHA1-H96-HMAC"|"GR341194CPRO1-H96-HMAC-65534"|"G2814789CPRO1-K256-MAC-65535"**
Значение	Возможные значения: "MD5-H96-KPDK" – Keyed MD5 "MD5-H96-HMAC" – HMAC MD5 (96 бит) "SHA1-H96-HMAC" – HMAC SHA-1 (96 бит) "GR341194CPRO1-H96-HMAC-65534" – реализация ГОСТ Р 34.11-94 (96 бит) "G2814789CPRO1-K256-MAC-65535" – реализация ГОСТ 28147-89 (в режиме выработки имитовставки).
Значение по умолчанию	если в атрибуте IntegrityAlg алгоритм не указан, а в атрибуте CipherAlg алгоритм указан, то проверка целостности пакета не производится.

Пример структуры ESPProposal

```
ESPTransform esp_trf_01(
 LifetimeSeconds = 28800
 LifetimeKilobytes = 4608000
 CipherAlg  = "G2814789CPRO1-K256-CBC-254"
 IntegrityAlg = "GR341194CPRO1-H96-HMAC-65534"
)
ESPTransform esp_trf_02(
 LifetimeSeconds = 28800
 LifetimeKilobytes = 4608000
 CipherAlg  = "G2814789CPRO1-K256-CBC-254"
 IntegrityAlg = "MD5-H96-HMAC"
)
ESPTransform esp_trf_03(
 LifetimeSeconds = 28800
 LifetimeKilobytes = 4608000
 CipherAlg  = "G2814789CPRO1-K256-CBC-254"
 IntegrityAlg = "SHA1-H96-HMAC"
)
ESPProposal ESP_1(
 Transform *= esp_trf_01,esp_trf_02,esp_trf_03
)
```


11.16 Структура IKERule

Структура IKERule описывает правило создания контекста соединения для протокола IKE.

Имя структуры	IKERule
Атрибуты	IKEPeerIPFilter
	IKELocalIPFilter
	DoNotUseDPD
	DPDIdleDuration
	DPDResponseDuration
	DPDRetries
	IKECFGRequestAddress
	DoAutopass
	AggrModeAuthMethod
	MainModeAuthMethod
	AggrModePriority
	Transform
	Priority

Атрибут IKEPeerIPFilter

Атрибут IKEPeerIPFilter описывает список допустимых IP-адресов партнера, при которых применяется данное правило.

Этот атрибут используется VPN-устройством, выступающим в роли ответчика IKE-сессии, при проверке UDP-заголовка первого (входящего) пакета.

Для VPN-устройства, выступающего в роли инициатора создания IKE-сессии, этот атрибут игнорируется.

Синтаксис	IKEPeerIPFilter* = IP | IP/ЦЕЛОЕ32
Значение	IP-адрес IP..IP – диапазон IP-адресов IP/ЦЕЛОЕ32 – IP-адрес с маской подсет
Значение по умолчанию	допускаются любые IP-адреса.

Атрибут IKELocalIPFilter

Атрибут IKELocalIPFilter описывает список допустимых локальных IP-адресов, при которых применяется данное правило.

Этот атрибут используется VPN-устройством, выступающим в роли ответчика IKE-сессии, при проверке UDP-заголовка первого (входящего) пакета.

Для VPN-устройства, выступающего в роли инициатора создания IKE-сессии, этот атрибут игнорируется.

Синтаксис	IKELocalIPFilter* = IP | IP/ЦЕЛОЕ32

Значение	IP-адрес IP..IP – диапазон IP-адресов IP/ЦЕЛОЕ32 – IP-адрес с маской подсети
Значение по умолчанию	IP-адрес – любой из локальных адресов VPN-устройства

Атрибут DoNotUseDPD

Атрибут DoNotUseDPD задает режим использования протокола DPD (Dead Peer Detection).

Синтаксис	DoNotUseDPD = **TRUE | FALSE**
Значение	TRUE – не использовать протокол DPD FALSE – использовать протокол DPD
Значение по умолчанию	FALSE

Атрибут DPDIdleDuration

Атрибут DPDIdleDuration задает допустимый период времени отсутствия входящего трафика от партнера, по истечении которого, при наличии исходящего трафика, активируется DPD-сессия. Если атрибут DoNotUseDPD = TRUE, то атрибут DPDIdleDuration игнорируется.

Синтаксис	DPDIdleDuration = ЦЕЛОЕ32
Значение	целое значение из диапазона 1..32767
Значение по умолчанию	60

Атрибут DPDResponseDuration

Атрибут DPDResponseDuration задает время ожидания ответа от партнера на DPD запрос в секундах. Если атрибут DoNotUseDPD = TRUE, то атрибут DPDResponseDuration игнорируется.

Синтаксис	DPDResponseDuration = ЦЕЛОЕ32
Значение	целое значение из диапазона 1..300
Значение по умолчанию	5

Атрибут DPDRetries

Атрибут DPDRetries задает число попыток провести DPD обмен. Если все попытки закончились неудачей, защищенное соединение (IKE-контекст) считается "мертвым", и производится попытка создать его заново. Если атрибут DoNotUseDPD = TRUE, то атрибут DPDRetries игнорируется.

Синтаксис	DPDRetries = ЦЕЛОЕ32
Значение	целое значение из диапазона 1..10
Значение по умолчанию	3

Атрибут IKECFGRequestAddress

Атрибут IKECFGRequestAddress задает режим работы IKECFG-клиента.

Синтаксис	IKECFGRequestAddress = **TRUE** | **FALSE**
Значение	TRUE – Клиент безопасности является активным IKECFG-клиентом, т.е. Клиент безопасности инициирует посылку запроса на получение внутреннего IP-адреса у партнера сразу после создания IKE SA. Если адрес не получен, это не является ошибкой, производится попытка создать IPsec SA без использования IKECFG. FALSE – Клиент безопасности является пассивным IKECFG-клиентом, т.е. IKECFG-сессия может быть проведена только по инициативе партнера, если он является IKECFG-сервером.
Значение по умолчанию	FALSE
Примечание	Не используйте запрос на получение IKECFG-адреса, если по сценарию планируется защищать трафик от клиента до туннельного адреса S-Terra Gate. Политика безопасности не будет работать, если туннельный адрес партнера (структура TunnelEntry, атрибут PeerIPAddress) совпадает с IP-адресом или подсетью партнера (структура Filter, атрибут DestinationIP), на которые распространяется правило фильтрации. Нельзя использовать запрос на получение IKECFG-адреса при транспортном режиме (структура IPsecAction, атрибут TunnelingParameters).

Атрибут AggrModeAuthMethod

Атрибут AggrModeAuthMethod содержит список структур, определяющих способ и параметры аутентификации в агрессивном режиме IKE. В списке не должно быть задано двух одинаковых методов аутентификации.

Синтаксис	AggrModeAuthMethod* = AuthMethodDSSSign | AuthMethodRSASign | FuthMethodGOSTSign | AuthMethodPreshared
Значение	AuthMethodDSSSign – Аутентификация с использованием DSA подписи AuthMethodRSASign – Аутентификация с использованием RSA подписи AuthMethodGOSTSign – Аутентификация с использованием подписи алгоритмом ГОСТ34.10 AuthMethodPreshared
Значение по умолчанию	При отсутствии MainModeAuthMethod атрибут является обязательным. При наличии атрибута MainModeAuthMethod Aggressive Mode не проводится
Примечание	Хотя бы один из атрибутов AggrModeAuthMethod или MainModeAuthMethod должен быть задан.

Атрибут MainModeAuthMethod

Атрибут MainModeAuthMethod содержит список структур, определяющих способ и параметры аутентификации в основном режиме IKE. В списке не должно быть задано двух одинаковых методов аутентификации.

Синтаксис	MainModeAuthMethod* = AuthMethodDSSSign | AuthMethodRSASign | FuthMethodGOSTSign | AuthMethodPreshared
Значение	AuthMethodDSSSign – Аутентификация DSA подписью AuthMethodRSASign – Аутентификация RSA подписью AuthMethodGOSTSign – Аутентификация при помощи подписи алгоритмом ГОСТ3410 AuthMethodPre
Значение по умолчанию	При отсутствии атрибута AggrModeAuthMethod атрибут является обязательным. При наличии атрибута AggrModeAuthMethod Main Mode не проводится
Примечание	Хотя бы одно из атрибутов AggrModeAuthMethod или MainModeAuthMethod должен быть задан.

Атрибут AggrModePriority

AggrModePriority задает режим использования Aggressive Mode.

Атрибут используется только для инициатора в случае, если заданы значения MainModeAuthMethod и AggrModeAuthMethod одновременно.

Атрибут игнорируется, если задан только один режим (Main Mode или Aggressive Mode).

Синтаксис	AggrModePriority = **TRUE** | **FALSE**
Значение	TRUE – Aggressive Mode является более приоритетным, инициатор начинает первую фазу IKE в "агрессивном" режиме. FALSE – Main Mode является более приоритетным, то инициатор начинает первую фазу IKE в "основном" режиме
Значение по умолчанию	FALSE

Атрибут Transform

Атрибут Transform задает список допустимых наборов криптографических параметров для ISAKMP SA. Количество элементов списка не ограничего.

Синтаксис	Transform* = `IKETransform`
Значение по умолчанию	не существует, атрибут обязательный.

Атрибут Priority

Атрибут Priority задает приоритет данного правила IKERule. Этот атрибут используется ответчиком для выбора IKE-правила, если по параметрам, присланным партнером, подходят несколько правил. Из двух подходящих правил с разными приоритетами выберется то, у которого значение Priority меньше.

Порядок выбора IKE-правила из правил с одинаковым приоритетом не определен.

Синтаксис	Priority = ЦЕЛОЕ32
Значение	Целое число из диапазона $1..2^{32}-1$.
Значение по умолчанию	$2^{32}-1$

11.17 Структура IKETransform

Структура IKETransform задает набор параметров, необходимых для создания ISAKMP SA.

Имя структуры IKETransform

Атрибуты LifetimeSeconds
LifetimaKilobytes
LifetimeSessions
NoSmoothRekeying
CipherAlg
HashAlg
GroupID
RestrictAuthenticationTo

Атрибут LifetimeSeconds

Атрибут LifetimeSeconds задает время существования IKE-контекста (в секундах).

Синтаксис	LifetimeSeconds = ЦЕЛОЕ32
Значение	Целое число из диапазона $1..2^{32}-1$
Значение по умолчанию	нет ограничений на действие SA
Примечание	Для совместимости IOS-партнером (Cisco) нужно всегда указывать в своем предложении атрибут LifetimeSeconds - время жизни в секундах и высылать IOS-партнеру. В противном случае, IOS будет пытаться поместить в принятое предложение новый атрибут – время жизни SA по времени, которое IOS-ом будет установлено для создаваемого SA. Это является неприемлемым.

Атрибут LifetimeKilobytes

Атрибут LifetimeKilobytes задает максимальное ограничение по трафику на действие контекста в килобайтах.

Синтаксис	LifetimeKilobytes = ЦЕЛОЕ32
Значение	Целое число из диапазона $1..2^{32}-1$
Значение по умолчанию	нет ограничений на действие SA.

Атрибут LifetimeSessions

Атрибут LifetimeSessions задает ограничение по числу IPsec SA (числу успешных Quick Mode – QM), которые можно сделать с использованием одного IKE-контекста.

Синтаксис	LifetimeSessions = ЦЕЛОЕ32
Значение	Целое число из диапазона $1..2^{32}-1$
Значение по умолчанию	нет ограничений на действие SA по числу созданных под его защитой IPsec SA.

Атрибут NoSmoothRekeying

Атрибут NoSmoothRekeying задает режим "мягкой" смены ключевого материала.

Синтаксис	NoSmoothRekeying = **TRUE | FALSE**
Значение	TRUE – заблаговременная смена ключевого материала (rekeying) не проводится. При отсутствии подходящего ISAKMP SA, новый ISAKMP SA создаётся только по запросу из ядра на создание IPsec SA – при наличии исходящего IP-пакета, либо по инициативе партнера. В результате процесс создания IPsec SA существенно задерживается FALSE – заблаговременно, незадолго до окончания действия ISAKMP SA, на его основе (по тем же правилам и с теми же Identity) проводится IKE-сессия по созданию нового ISAKMP SA - rekeying. Rekeying не проводится, если за время существования старого SA под его защитой не было никакого трафика[15].
Значение по умолчанию	FALSE

Атрибут CipherAlg

Атрибут CipherAlg задает набор предлагаемых/допустимых алгоритмов шифрования для ISAKMP.

Указывается только один алгоритм шифрования.

Если же существует необходимость задать несколько алгоритмов шифрования (их комбинацию), то используйте альтернативный подход: в атрибуте `Transform` структуры IKERule укажите список структур IKETransform, а в каждой структуре IKETransform задайте только один алгоритм шифрования (см. Пример структуры IKERule).

Синтаксис	CipherAlg = **"DES-CBC"|"IDEA-CBC"|"DES3-K168-CBC"|"AES-K128-CBC"|"AES-K192-CBC"|"AES-K256-CBC"|"G2814789CPRO1-K256-CBC-65534"**
Значение	возможные значения: "DES-CBC" – DES в режиме CBC "IDEA-CBC" – IDEA в режиме CBC "DES3-K168-CBC" – DES3 в режиме CBC "AES-K128-CBC" – AES в режиме CBC с длиной ключа 128 "AES-K192-CBC" – AES в режиме CBC с длиной ключа 192 "AES-K256-CBC" – AES в режиме CBC с длиной ключа 256 "G2814789CPRO1-K256-CBC-65534" – реализация ГОСТ 28147-89 в режиме CBC
Значение по умолчанию	не существует, атрибут обязательный, список должен содержать хотя бы один элемент.

[15] Для проведения rekeying необходимо, чтобы время жизни обновляемого соединения было существенно больше времени, которое отводится на проведение IKE-сессии.

Атрибут HashAlg

Атрибут HashAlg задает допустимый алгоритм вычисления хэша для ISAKMP[16].

Указывается только один алгоритм хэширования.

Если же существует необходимость задать несколько алгоритмов хэширования, то используйте альтернативный подход: в атрибуте `Transform` структуры IKERule укажите список структур IKETransform, а в каждой структуре IKETransform задайте только один алгоритм хэширования (см. Пример структуры IKERule).

Синтаксис	HashAlg = **"MD5"|"SHA1"|"GR341194CPRO1-65534"**
Значение	"MD5" "SHA1" "GR341194CPRO1-65534" – реализация ГОСТ Р 34.11-94
Значение по умолчанию	не существует, атрибут обязательный, список должен содержать хотя бы один элемент.

Атрибут GroupID

Атрибут GroupID описывает допустимый параметр выработки ключевого материала для ISAKMP. Используется алгоритм VKO ГОСТ Р 34.10-2001 [RFC4357] либо один из алгоритмов Диффи-Хеллмана. Рекомендуется указывать только один элемент.

Если существует необходимость задать список групп, то используйте альтернативный подход: в атрибуте `Transform` структуры IKERule укажите список структур IKETransform, а в каждой структуре IKETransform задайте только один элемент списка (см. Пример структуры IKERule – MainMode).

Синтаксис	GroupID = **VKO_1B|MODP_768|MODP_1024|MODP_1536**
Значение	VKO_1B – используется алгоритм VKO GOST R 34.10-2001, длина ключа 256 бит MODP_768 – стандартная Oakley-группа с длиной ключа 768 бит –группа 1 MODP_1024 – стандартная Oakley-группа с длиной ключа 1024 бита – группа 2 MODP_1536 – стандартная Oakley-группа с длиной ключа 1536 бит – группа 5
Значение по умолчанию	не существует, атрибут обязательный.
Примечание	Стоит отметить, что в правиле IKE (IKERule) предоставление партнеру выбора различных элементов списка возможно только в основном режиме IKE (MainMode). Если правило IKE предусматривает агрессивный режим (присутствует структура AggrModeAuthMethod), то в этом правиле IKERule во всех структурах IKETransform атрибут GroupID должен иметь только одно значение, и оно должно быть одинаковым во всех структурах IKETransform, т.е. должна быть указана одна и та же Oakley-группа либо VKO_1B.

[16] Если в правиле IKERule, использующем данный IKETransform указан метод аутентификации типа AuthMethodGOSTSign, то алгоритм вычисления хэша для ISAKMP *не может быть указан* MD5 или SHA1.

Атрибут RestrictAuthenticationTo

Атрибут RestrictAuthenticationTo определяет, с каким типом аутентификации может использоваться данный трансформ. Если не задано методов аутентификации подходящего типа, соответствующих используемому режиму (main/aggressive), данный трансформ не будет использован. Если для способа аутентификации в IKERule нет подходящего трансформа, произойдет ошибка разбора конфигурации.

Синтаксис	RestrictAuthenticationTo = **RSA_SIGN|DSS_SIGN|GOST_SIGN|PRESHARED**
Значение	RSA_SIGN DSS_SIGN GOST_SIGN PRESHARED
Значение по умолчанию	ограничение не установлено.

Пример структуры IKERule

```
IKETransform ike_trf_01(
 LifetimeSeconds = 28800
 CipherAlg = "G2814789CPRO1-K256-CBC-65534"
 HashAlg = "GR341194CPRO1-65534"
 GroupID = VKO_1B
)
IKETransform ike_trf_02(
 LifetimeSeconds = 28800
 CipherAlg = "G2814789CPRO1-K256-CBC-65534"
 HashAlg = "GR341194CPRO1-65534"
 GroupID = MODP_1536
)
IKETransform ike_trf_03(
 LifetimeSeconds = 28800
 CipherAlg = "DES-CBC"
 HashAlg = "GR341194CPRO1-65534"
 GroupID = MODP_1024
)
IKETransform ike_trf_04(
 LifetimeSeconds = 28800
 CipherAlg = "AES-K128-CBC"
 HashAlg = "GR341194CPRO1-65534"
 GroupID = MODP_768
)
IKERule ike_rule(
 DoNotUseDPD = FALSE
 DPDIdleDuration = 60
 DPDResponseDuration = 5
 DPDRetries = 3
 MainModeAuthMethod *= auth_method_01
 Transform *= ike_trf_01,ike_trf_02,ike_trf_03,ike_trf_04
 DoAutopass = TRUE
)
IdentityEntry auth_identity_01(
)
AuthMethodPreshared auth_method_01(
 SharedIKESecret = "dfd"
 LocalID = auth_identity_01
)
```


11.18 Структуры AuthMethodDSSSign, AuthMethodRSASign, AuthMethodGOSTSign

Указанные структуры задают аутентификационную информацию при использовании сертификатов. Алгоритм (RSA, DSA, GOST), указанный в названии структуры, является криптографическим алгоритмом.

AuthMethodDSSSign – аутентификация DSS подписью.

AuthMethodRSASign – аутентификация RSA подписью.

AuthMethodGOSTSign – аутентификация при помощи подписи алгоритмом ГОСТ3410-2001.

Имя структур	AuthMethodDSSSign
	AuthMethodRSASign
	AuthMethodGOSTSign
Атрибуты	LocalID
	RemoteID
	LocalCredential
	RemoteCredential
	AcceptCredentialFrom
	DoNotMapLocalIDToCert
	DoNotMapRemoteIDToCert
	SendRequestMode
	SendCertMode

Атрибут LocalID

Атрибут LocalID задает идентификационную информацию, посылаемую партнеру в первой фазе IKE.

Синтаксис	LocalID =`IdentityEntry`
Значение	В структуре IdentityEntry допускается задание одного значения одному из идентификаторов типа `IPv4Address`, `FQDN`, `EMail`, `DistinguishedName`. При задании значения атрибуту DistinguishedName использование в строке Subject зарезервированного слова TEMPLATE недопустимо. При задании значения атрибуту IPv4Address использование диапазона IP-адресов недопустимо. Если значение задано зарезервированным словом USER_SPECIFIC_DATA, то в качестве идентификатора будет использовано соответствующее значение из локального сертификата. Если в сертификате со
Значение по умолчанию	первый IP-адрес сетевого интерфейса, с которого отсылаются ISAKMP-пакеты партнеру.

Атрибут RemoteID

Атрибут RemoteID задает требования к идентификационной информации партнера.

Синтаксис	RemoteID = `IdentityEntry`
Значение	В структуре IdentityEntry допускается задание нескольких идентификаторов типа `IPv4Address`,`FQDN`, `Email`, `DistinguishedName`
Значение по умолчанию	принимается любой ID партнера.

Атрибут LocalCredential

Атрибут LocalCredential задает требуемые параметры сертификата данного VPN-устройства.

Синтаксис	LocalCredential = `CertDescription`
Значение по умолчанию	требования отсутствуют. Используется любой локальный сертификат.

Атрибут RemoteCredential

Атрибут RemoteCredential задает требуемые параметры сертификата партнера по взаимодействию.

Синтаксис	RemoteCredential* = `CertDescription`
Значение по умолчанию	допускается любой доверенный сертификат.

Атрибут AcceptCredentialFrom

Атрибут AcceptCredentialFrom задает требуемые параметры CA-сертификата, удостоверяющего подлинность сертификата партнера.

Синтаксис	AcceptCredentialFrom* = `CertDescription`
Значение по умолчанию	используется любой из тех CA, которому мы доверяем.

Атрибут DoNotMapLocalIDToCert

Атрибут DoNotMapLocalIDToCert задает режим использования локального идентификатора при поиске локального сертификата.

Синтаксис	DoNotMapLocalIDToCert = **TRUE** | **FALSE**
Значение	TRUE – при поиске локального сертификата используются описания сертификатов, указанные в атрибуте LocalCredential. Значение атрибута LocalID игнорируется FALSE – при поиске локального сертификата используется список CertDescription. Каждый элемент этого списка является объединением полей атрибутов LocalID и LocalCredential. Объединение строится по следующим правилам: если LocalID задан зарезервированным словом

	USER_SPECIFIC_DATA, то используется CertDescription в том виде, как он задан в LocalCredential если значение LocalID не противоречит LocalCredential, оно является дополнительным критерием поиска сертификата. если значение LocalID противоречит LocalCredential, соединение не построится.
Значение по умолчанию	FALSE

Атрибут DoNotMapRemoteIDToCert

Атрибут DoNotMapRemoteIDToCert задает режим использования идентификатора партнера при поиске его сертификата.

Синтаксис	DoNotMapRemoteIDToCert **= TRUE** | **FALSE**
Значение	TRUE – при поиске сертификата партнера используются описания сертификатов, указанные в атрибуте RemoteCredential, значение атрибута RemoteID игнорируется FALSE – при поиске сертификата партнера используется список CertDescription. Каждый элемент этого списка является объединением присланного идентификатора партнера и CertDescription из атрибута RemoteCredential. Правила объединения совпадают с ранее описанными правилами в атрибуте DoNotMapLocalIDToCert.
Значение по умолчанию	FALSE.

Атрибут SendRequestMode

Атрибут SendRequestMode определяет логику отсылки запроса сертификата партнера.

Синтаксис	SendRequestMode = **AUTO | NEVER | ALWAYS**
Значение	AUTO – запрос высылается, если сертификат партнера не доступен локально или не может быть однозначно определено, каким сертификатом воспользуется партнер NEVER – запрос не высылается ALWAYS – запрос высылается всегда
Значение по умолчанию	AUTO

Атрибут SendCertMode

Атрибут SendCertMode определяет логику отсылки локального сертифика в процессе первой фазы IKE на запрос партнера. В своем запросе партнер может и указать, какому CA сертификату он доверяет. Если такой сертификат не найден, то он не отсылается.

Синтаксис	SendCertMode = **AUTO** | **NEVER** | **ALWAYS| CHAIN**
Значение	AUTO – автоматически определяется, когда необходима отсылка локального сертификата партнеру: если партнер не прислал запроса, то сертификат не отсылается

	если партнер прислал запрос и соответствующий сертификат был найден, то партнеру высылается либо сертификат, либо найденная цепочка сертификатов если партнер прислал запрос и этот запрос не был удовлетворен, то сертификат не высылается. NEVER – сертификат не высылается ALWAYS – сертификат высылается всегда CHAIN – сертификат высылается всегда, причем в составе с цепочкой доверительных CA: имеется ввиду цепочка сертификатов, построенная от локального сертификата до CA, который удовлетворяет описанию, присланному партнером в запросе. В общем случае это CA, удовлетворяющий запросу партнера, произвольное количество промежуточных CA и локальный сертификат.
Значение по умолчанию	AUTO

11.19 Структура AuthMethodPreshared

Структура AuthMethodPreshared задает аутентификационную информацию при использовании предопределенных (Preshared) ключей.

Имя структуры	AuthMethodPreshared
Атрибуты	LocalID
	RemoteID
	SharedIKESecret

Атрибут LocalID

Атрибут LocalID задает идентификационную информацию данного VPN-устройства. В структуре IdentityEntry допускается задание только одного идентификатора с одним значением.

При задании значения атрибуту `IPv4Address` использование диапазона IP-адресов недопустимо.

Использование зарезервированного слова USER_SPECIFIC_DATA недопустимо.

Синтаксис	LocalID = `IdentityEntry`
Значение по умолчанию	локальный IP-адрес из IKE-пакета.

Атрибут RemoteID

Атрибут RemoteID задает требования к идентификационной информации партнера. В структуре IdentityEntry допускается задание нескольких идентификаторов разных типов.

Синтаксис	RemoteID = `IdentityEntry`
Значение по умолчанию	принимается любой ID партнера.

Атрибут SharedIKESecret

Атрибут SharedIKESecret определяет ссылку на предопределенный секретный ключ. В атрибуте указывается имя предопределенного (Preshared) ключа, хранимого в базе Продукта.

Синтаксис	SharedIKESecret = СТРОКА
Значение	имя предопределенного (Preshared) ключа.
Значение по умолчанию	не существует, атрибут обязательный.

11.20 Структура IdentityEntry

Структура IdentityEntry описывает идентификационную информацию. Варианты задания этой структуры приведены в описаниях структур Структура AuthMethodPreshared и AuthMethod{DSS|RSA|GOST}Sign.

Имя структуры	IdentityEntry
Атрибуты	IPv4Address – IPv4 адрес
	FQDN – FQDN хоста
	EMail – EMail пользователя
	DistinguishedName – DN в формате X509Subject
	KeyID – идентификатор ключа

Если структура IdentityEntry используется определенным методом аутентификации, то атрибуты, не соответствующие данному методу, игнорируются. Атрибуты, используемые для определенных методов аутентификации:

- AuthMethodPreshared
  - IPv4Address
  - KeyID
- AuthMethod{DSS|RSA|GOST}Sign
  - IPv4Address
  - FQDN
  - EMail
  - DistinguishedName.

Атрибут IPv4Address

Атрибут IPv4Address задает описание идентификатора по указанным IP-адресам.

Синтаксис	для данного VPN устройства: IPv4Address = IP | **USER_SPECIFIC_DATA** для партнера: IPv4Address* = IP|IP..IP|IP/ЦЕЛОЕ32| **USER_SPECIFIC_DATA**
Значение	для данного VPN устройства: IP – один IP-адрес **USER_SPECIFIC_DATA** для партнера: IP – список IP-адресов IP..IP – список диапазонов IP-адресов IP/ЦЕЛОЕ32 – список подсетей с IP-адресом и маской **USER_SPECIFIC_DATA** Если задано значение USER_SPECIFIC_DATA, то берется первый IP-адрес из расширения **Subject Alternative Name**

	локального сертификата, используемого для подписи. Если IP-адрес в сертификате отсутствует, то соединение не создается. Если заданы диапазоны IP-адресов либо подсети, то это означает, что принимается любой Identity типа IP-адрес, если значение IP, присланное партнером в таком Identity, попадает в указанный диапазон, либо подсеть.
Значение по умолчанию	используются другие атрибуты.

Атрибут FQDN

Атрибут FQDN (Fully Qualified Domain Name – полностью определенное доменное имя) задает описание идентификатора хоста по указанным DNS именам. Для AuthMethodPreshared этот атрибут игнорируется.

Синтаксис	FQDN* = СТРОКА | **USER_SPECIFIC_DATA**
Значение	строки вида "host.domain". Шаблоны не допускаются. если задано значение `USER_SPECIFIC_DATA`, то при проверке/отсылке Identity используется поле **DNS** расширения **Subject Alternative Name** соответствующего сертификата, используемого соответственно для проверки/формирования подписи.
Значение по умолчанию	используются другие атрибуты.

Атрибут EMail

Атрибут EMail задает описание идентификатора пользователя по указанным Email-адресам. Для AuthMethodPreshared этот атрибут игнорируется.

Синтаксис	Email* = СТРОКА | **USER_SPECIFIC_DATA**
Значение	строки вида "user@host.domain". Шаблоны не допускаются; если задано значение USER_SPECIFIC_DATA, то при проверке/отсылке Identity используется поле **EMail** расширения **Subject Alternative Name** сертификата, используемого соответственно для проверки/формирования подписи.
Значение по умолчанию	используются другие атрибуты.

Атрибут DistinguishedName

Атрибут DistinguishedName задает описание идентификатора по указанным DN (уникальное имя в формате X509Subject.). Для AuthMethodPreshared этот атрибут игнорируется.

Синтаксис	DistinguishedName* = `CertDescription` | **USER_SPECIFIC_DATA**
Значение	в каждой структуре CertDescription допускается использовать только поле Subject если задано значение USER_SPECIFIC_DATA, то при проверке/отсылке Identity используется полное описание

	раздела **Subject Name** сертификата, используемого соответственно для проверки/формирования подписи.
Значение по умолчанию	используются другие атрибуты.

Атрибут KeyID

Атрибут KeyID задает описание Identity по указанным идентификаторам Preshared ключей. Для AuthMethod{ DSS|RSA|GOST}Sign этот атрибут игнорируется.

Синтаксис	KeyID* = СТРОКА
Значение	строки, содержащие шестнадцатеричное представление идентификаторов ключей; рекомендуется при составлении идентификатора ключа использовать шестнадцатеричное представление только печатных символов без пробела: Именно такое ограничение существует при формировании конфигурации IOS.
Значение по умолчанию	используются другие атрибуты.

Пример

```
AuthMethodPreshared auth_key (
    RemoteID = IdentityEntry(
  IPv4Address *= 192.168.13.117, 192.168.13.118
  )
    SharedIKESecret = "cskey"
)
```


11.21 Структура CertDescription

Структура CertDescription используется для задания собственного идентификатора и идентификатора партнера, для задания характеристик локального сертификата и сертификата партнера.

Для задания СТРОКИ в атрибутах этой структуры смотрите формат DN в разделе "Формат задания DistinguishedName в LSP".

Имя структуры CertDescription

Атрибуты Subject
AlternativeSubject
Issuer
AlternativeIssuer
FingerprintMD5
FingerprintSHA1
SerialNumber

Атрибут Subject

Атрибут Subject задает значение/шаблон поля Subject сертификата.

Синтаксис	Subject = **TEMPLATE | COMPLETE**, СТРОКА
Значение	TEMPLATE – флаг, при котором указанная строка представляет собой незаконченное значение поля Subject сертификата.При поиске и сравнении, поле Subject сертификата должно содержать указанную строку COMPLETE – флаг, при котором указанная строка представляет собой законченное значение поля Subject сертификата. При поиске и сравнении поле Subject сертификата должно совпадать с указанным множеством атрибутов и их значениями (с точностью до порядка указания в сертификате) в строке. **Предупреждение:** DN в строке должен быть задан точно также, как он задан в сертификате: необходимо строго соблюдать количество пробелов и регистр символов.
Значение по умолчанию	если задана строка, а флаг не указан, то по умолчанию он равен COMPLETE; если не задана строка, то поле Subject сертификата принимает любые значения.

Пример: Допустимые варианты:

```
Subject = TEMPLATE, "ou=eng"
Subject = "ou=eng", TEMPLATE
Subject = COMPLETE, "c=RU,o=co.,ou=eng,cn=engineer"
Subject = "c=RU, o=co, ou=eng, cn=engineer"
```

Недопустимые варианты:

```
Subject = TEMPLATE, "ou=eng", COMPLETE
Subject = "ou=eng", "ou=qa"
```


Атрибут AlternativeSubject

Атрибут AlternativeSubject задает значение/шаблон поля Alternative Subject Extension сертификата.

Синтаксис	AlternativeSubject = СТРОКА
Значение по умолчанию	любое значение Alternative Subject Extension сертификата.

Атрибут Issuer

Атрибут Issuer задает значение/шаблон поля Issuer сертификата.

Синтаксис	Issuer = **TEMPLATE | COMPLETE**, СТРОКА
Значение	TEMPLATE – флаг, при котором указанная строка представляет собой незаконченное значение поля Issuer сертификата. При поиске и сравнении, поле Issuer сертификата должно содержать указанную строку. COMPLETE – флаг, при котором указанная строка представляет собой законченное значение поля Issuer сертификата. При поиске и сравнении, поле Issuer сертификата должно совпадать с указанным множеством атрибутов и их значениями (с точностью до порядка указания в сертификате) в строке. **Предупреждение:** DN в строке должен быть задан точно также, как он задан в сертификате: необходимо строго соблюдать количество пробелов и регистр символов.
Значение по умолчанию	если задана строка, но опущен флаг TEMPLATE или COMPLETE , то по умолчанию он равен COMPLETE; если не задана строка, то поле Issuer сертификата принимает любые значения.

Атрибут AlternativeIssuer

Атрибут AlternativeIssuer задает значение/шаблон Alternative Issuer Extension сертификата.

Синтаксис	AlternativeIssuer = СТРОКА
Значение по умолчанию	любое значение Alternative Issuer Extension сертификата.

Атрибут FingerprintMD5

Атрибут FingerprintMD5 задает значение хеш-функции алгоритма MD5 по бинарному представлению сертификата.

Синтаксис	FingerprintMD5 = СТРОКА
Значение	шестнадцатеричная запись значения хэш-функции, длина строки должна быть равна 32 символам.
Значение по умолчанию	любое значение хэш-функции.

Атрибут FingerprintSHA1

Атрибут FingerprintSHA1 задает значение хеш-функции алгоритма SHA1 по бинарному представлению сертификата.

Синтаксис	FingerprintSHA1 = СТРОКА
Значение	шестнадцатеричная запись значения хэш-функции, длина строки должна быть равна 40 символам.
Значение по умолчанию	любое значение хэш-функции.

Атрибут SerialNumber

Атрибут SerialNumber задает значение серийного номера сертификата.

Синтаксис	SerialNumber = СТРОКА
Значение	шестнадцатеричная запись серийного номера.
Значение по умолчанию	любое значение серийного номера.

Пример

```
RemoteCredential* = CertDescription(
    Issuer* = COMPLETE, " CN=S-Terra CenterCA, O=S-Terra, L=Moscow,
                          C=RU"
    Subject* = TEMPLATE, "CN=S-Terra, OU=QA"
    AlternativeSubject = "EMAIL=inform@s-terra.com,
             DNS= tester.s-terra.com, IP =10.10.10.10"
    SerialNumber =  "567A99991E1F"
)
```


11.22 Структура FirewallParameters

Структура FirewallParameters описывает глобальные параметры межсетевого экрана. В конфигурации должна быть только одна структура данного типа. Этой структуре имя не присваивается.

Имя структуры FirewallParameters

Атрибуты
TCPSynSentTimeout
TCPSynRcvdTimeout
TCPFinTimeout
TCPClosedTimeout
TCPEstablishedTimeout
TCPHalfOpenMax
TCPHalfOpenLow
TCPSessionRateMax
TCPSessionRateLow
TCPSessionsMax
TCPStrictnessLevel

Атрибуты TCPSynSentTimeout, TCPSynRcvdTimeout, TCPFinTimeout, TCPClosedTimeout, TCPEstablishedTimeout

Атрибуты устанавливают время жизни записи о соединении. Межсетевой экран определяет состояние TCP-соединения для каждого из партнеров и, в зависимости от этого, выставляет время жизни записи о соединении. В таблице приведены стандартные названия для состояний TCP и соответствующие параметры, задающие время жизни.

Синтаксис	Атрибут = ЦЕЛОЕ32
Значение	Целое число из диапазона 1..65535
Значение по умолчанию	см. таблицу ниже.

Состояние соединения	Параметр LSP	Значение по умолчанию (сек.)
CLOSED, LISTEN	TCPClosedTimeout	30
SYNSENT	TCPSynSentTimeout	30
SYNRCVD	TCPSynRcvdTimeout	30
ESTAB	TCPEstablishedTimeout	3600
FINWAIT-1, FINWAIT-2, CLOSING, TIMEWAIT, LASTACK, CLOSED	TCPFinTimeout	5

Значение TCPEstablishedTimeout может быть переопределено для конкретного правила фильтрации (см. Filter.ExtendedAction)

Атрибут TCPHalfOpenMax

Атрибут TCPHalfOpenMax задает максимальное разрешенное количество одновременно существующих полуоткрытых сеансов, по достижении которого Клиент безопасности начинает их удаление.
При превышении данного предела новые соединения будут создаваться только за счет уничтожения полуоткрытых сеансов, созданных ранее. Таким образом, после превышения TCPHalfOpenMax полуоткрытые сеансы будет удаляться, пока их количество не достигнет

значения, заданного атрибутом TCPHalfOpenLow. Далее вновь допускается увеличение количества полуоткрытых сеансов.

Синтаксис	TCPHalfOpenMax = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..1000000
Значение по умолчанию	500

Атрибут TCPHalfOpenLow

Атрибут TCPHalfOpenLow задает количество одновременно существующих полуоткрытых сеансов, которое считается нормальным. В случае превышения максимального числа полуоткрытых сеансов, заданных атрибутом TCPHalfOpenMax, полуоткрытые сеансы будут уничтожаться до заданного предела.

Синтаксис	TCPHalfOpenLow = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..1000000
Значение по умолчанию	400

Атрибут TCPSessionRateMax

Атрибут TCPSessionRateMax задает верхнюю границу на количество новых контекстов соединений, создаваемых за минуту. Если частота появления новых контекстов соединений достигнет TCPSessionRateMax, то Клиент безопасности начнет их удаление до тех пор, пока частота появления новых контекстов соединений не уменьшится до величины, заданной атрибутом TCPSessionRateLow.

Синтаксис	TCPSessionRateMax = ЦЕЛОЕ32
Значение	Целое число из диапазона $0..2^{32}-1$
Значение по умолчанию	500 новых контекстов соединений в минуту.

Атрибут TCPSessionRateLow

Атрибут TCPSessionRateLow задает нижнюю границу на количество новых контекстов соединений, создаваемых за минуту, по достижении которой, Клиент безопасности прекращает их удаление.

Синтаксис	TCPSessionRateLow = ЦЕЛОЕ32
Значение	Целое число из диапазона $0..2^{32}-1$
Значение по умолчанию	400 новых контекстов соединений в минуту.

Атрибут TCPSessionsMax

Атрибут TCPSessionsMax задает максимальное разрешенное количество TCP-соединений. При превышении данного предела новые TCP-соединения будут отвергаться.

Синтаксис	TCPSessionsMax = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..1000000

Значение по умолчанию	65536

Атрибут TCPStrictnessLevel

Атрибут TCPStrictnessLevel используется для задания уровня "жесткости" к различным ситуациям, которые воспринимаются firewall как ошибочные.

Синтаксис	TCPStrictnessLevel = ЦЕЛОЕ32
Значение	Целое число из диапазона 0..6
Значение по умолчанию	3

В следующей таблице приведены основные отличия в поведении при различных значениях TCPStrictnessLevel. Показана зависимость выполнения таких действий как «уничтожение пакета» и «отказ в изменении состояния соединения» от уровня, заданного TCPStrictnessLevel и результата анализа заголовка TCP пакета.

Значение Strictness Level	Условие, при котором пакет уничтожается	Условие, при котором состояние соединения[17] не изменяется
0	Пакеты не уничтожаются firewall	При некорректном TCP заголовке (проверяется соответствие длины пакета, TCP заголовка, checksum)
1	При некорректном TCP заголовке	При некорректном TCP заголовке
2	При некорректном TCP заголовке	При некорректном TCP заголовке или при sequence, несоответствующем состоянию партнера
3	При некорректном TCP заголовке	При некорректном TCP заголовке, или при sequence, несоответствующем состоянию партнера, или при несоответствии флагов заголовка состоянию партнера
4	При некорректном TCP заголовке или при несоответствии флагов заголовка состоянию партнера	При некорректном TCP заголовке, или при sequence, несоответствующем состоянию партнера, или при несоответствии флагов заголовка состоянию партнера
5	При некорректном TCP заголовке или при несоответствии флагов заголовка состоянию партнера, или при sequence, несоответствующем состоянию партнера	При некорректном TCP заголовке или при sequence, несоответствующем состоянию партнера, или при несоответствии флагов заголовка состоянию партнера
6	При некорректном TCP заголовке или при несоответствии флагов заголовка состоянию партнера, или при sequence, несоответствующем состоянию партнера, или при приеме SYN для установившегося соединения	При некорректном TCP заголовке или при sequence, несоответствующем состоянию партнера, или при несоответствии флагов заголовка состоянию партнера, или при получении первым не SYN-пакета, или при приеме SYN-пакета для установившегося соединения

[17] Например, не пролонгируется существование записи о соединении.

11.23 Структура NetworkInterface

Структура NetworkInterface описывает логический сетевой интерфейс, который может соответствовать нескольким сетевым интерфейсам системы. В структуре описываются действия, которые должны быть выполнены с пакетом, при его прохождении через этот интерфейс.

В конфигурации допускается описание нескольких экземпляров данной структуры, в этом случае они должны отличаться значением поля LogicalName. Структуре NetworkInterface имя не присваивается.

Имя структуры	NetworkInterface
Атрибуты	LogicalName InputFilter OutputFilter InputClassification OutputClassification IPsecPolicy

Атрибут LogicalName

Атрибут LogicalName задает логическое имя интерфейса. Если интерфейс на момент загрузки не контролируется IPsec-драйвером, LSP на него загружена не будет.

Синтаксис	LogicalName = СТРОКА
Значение	логическое имя интерфейса
Значение по умолчанию	default (остальные сетевые интерфейсы)

Атрибут InputFilter

Атрибут InputFilter задает правила как stateless (пакетной) так и stateful (контекстной) фильтрации для входящих пакетов через данный интерфейс. Пакет, не попавший ни под одно из заданных правил фильтрации, удаляется. Если фильтры на интерфейсе не заданы, то пропускается любой пакет.

Синтаксис	InputFilter = `FilterChain`
Значение по умолчанию	входящие пакеты не фильтруются.

Атрибут OutputFilter

Атрибут OutputFilter задает правила как stateless (пакетной) так и stateful (контекстной) фильтрации для исходящих пакетов через данный интерфейс. Пакет, не попавший ни под одно из заданных правил фильтрации, удаляется. Если фильтры на интерфейсе не заданы, то пропускается любой пакет.

Синтаксис	OutputFilter = `FilterChain`
Значение по умолчанию	исходящие пакеты не фильтруются.

Атрибут InputClassification

Атрибут InputClassification задает правила классификации и выставления значения поля TOS в IP-заголовке входящих пакетов через данный интерфейс. Классификация и маркирование производится на открытых пакетах, то есть после IPsec декапсуляции. Входящий пакет, не попавший ни под одно из заданных правил, не классифицируется и пропускается в неизменном виде.

Синтаксис	InputClassification = `FilterChain`
Значение по умолчанию	классификация и маркирование пакетов не производится.

Атрибут OutputClassification

Атрибут OutputClassification задает правила классификации и выставления значения поля TOS в IP-заголовке исходящих пакетов через данный интерфейс. Классификация и маркирование производится на открытых пакетах, то есть до IPsec инкапсуляции. После инкапсуляции значение поля TOS копируется из внутреннего IP-заголовка во внешний. Исходящий пакет, не попавший ни под одно из заданных правил, не классифицируется и пропускается в неизменном виде.

Синтаксис	OutputClassification = `FilterChain`
Значение по умолчанию	классификация и маркирование пакетов не производится.

Атрибут IPsecPolicy

Атрибут IPsecPolicy задает правила защиты пакетов с помощью IPsec. В фильтрах описывается исходящий трафик, но фильтрация производится симметрично для входящего и исходящего трафика. То есть при обработке входящего трафика на сетевом интерфейсе SourceIP, SourcePort сравнивается с соответствующими полями заголовков пакета destination IP address, destination UDP port, destination TCP port, а DestinationIP и DestinationPort сравниваются с полями заголовков пакета source IP address, source UDP port, source TCP port. При обработке исходящего трафика на сетевом интерфейсе понятия source и destination в конфигурации соответствуют понятиям source и destination в пакете.

Синтаксис	IPsecPolicy = `FilterChain`
Значение по умолчанию	IPsec не применяется.

11.24 Структура FilterChain

Структура FilterChain задает список правил пакетной и контекстной фильтрации («цепочка» правил). Этой структуре может быть присвоено имя.

Имя структуры	FilterChain
Атрибуты	Filters

Атрибут Filters

Атрибут Filters задает список правил фильтрации с условиями срабатывания каждого правила. Порядок обработки каждого правила соответствует порядку перечисления фильтров в LSP за исключением ситуаций, когда используются переходы (см. атрибут Action параметр STRING).

Синтаксис	Filters* = `Filter`

Пример

```
FilterChain IPsecPolicy:DMAP (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
     Filter (
        SourceIP = 192.168.2.0/24
        DestinationIP = 192.168.2.240/29
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:DMAP:1:dmap:1 >
        LogEventID = "IPsec:Protect:DMAP:1:dmap:1:client"
    )
)
```

11.25 Структура Filter

Структура Filter задает правило пакетной и контекстной фильтрации.

Имя структуры	Filter
Атрибуты	SourceIP DestinationIP ProtocolID SourcePort DestinationPort PacketType Action ExtendedAction LogEventID Schedule Log Label

Атрибут SourceIP

Атрибут SourceIP задает возможные значения поля Source Address в IPv4-заголовке пакета[18].

Синтаксис	SourceIP *= IP | IP/ЦЕЛОЕ32
Значение	IP-адрес IP/ЦЕЛОЕ32 – IP-адрес с маской
Значение по умолчанию	допускается любое значение поля Source Address в IPv4-заголовке пакета.

Атрибут DestinationIP

Атрибут DestinationIP задает возможные значения поля Destination Address в IPv4-заголовке пакета[19].

Синтаксис	SourceIP *= IP | IP..IP | IP/ЦЕЛОЕ32
Значение	IP-адрес IP..IP – диапазон IP-адресов IP/ЦЕЛОЕ32 – IP-адрес с маской
Значение по умолчанию	допускается любое значение поля Destination Address в IPv4-заголовке пакета.

Атрибут ProtocolID

Атрибут ProtocolID задает возможные значения поля Protocol в IPv4-заголовке.

[18] Для цепочки фильтров IPsec правил используется особая интерпретация. См. описание IPsecPolicy.

[19] Для цепочки фильтров IPsec правил используется особая интерпретация. См. описание IPsecPolicy.

Синтаксис	ProtocolID *= ЦЕЛОЕ32 | ЦЕЛОЕ32..ЦЕЛОЕ32
Значение	целое число из диапазона 0..255. Значение 0 означает все сетевые протоколы.
Значение по умолчанию	любое значение поля Protocol в IPv4-заголовке.

Атрибут SourcePort

Атрибут SourcePort описывает список идентификаторов портов для указанных протоколов объекта[20]. Если значение ProtocolID не TCP и не UDP, то данный фильтр не совпадет, поиск фильтров продолжится. Если в ProtocolID заданы оба протокола – TCP и UDP, то указанные порты допускаются в сочетании с любым из двух протоколов.

Синтаксис	SourcePort *= ЦЕЛОЕ32 | ЦЕЛОЕ32..ЦЕЛОЕ32
Значение	целое число из диапазона 0..65535. Значение 0 означает все порты для указанных протоколов.
Значение по умолчанию	допускается любое значение поля Source Port в UDP либо TCP заголовке пакета.

Атрибут DestinationPort

Атрибут DestinationPort описывает список идентификаторов портов для указанных протоколов объекта[21]. Если значение ProtocolID не TCP и не UDP, то данный фильтр не совпадет, поиск фильтров продолжится. Если в ProtocolID заданы оба протокола – TCP и UDP, то указанные порты допускаются в сочетании с любым из двух протоколов.

Синтаксис	DestinationPort *= ЦЕЛОЕ32 | ЦЕЛОЕ32..ЦЕЛОЕ32
Значение	целое число из диапазона 0..65535. Значение 0 означает все порты для указанных протоколов.
Значение по умолчанию	допускается любое значение поля Destination Port в UDP либо TCP заголовке.

Атрибут PacketType

Атрибут PacketType задает список типов пакетов, для которых данное правило может сработать.

Синтаксис	PacketType *= ЦЕЛОЕ32
Значение	TRANSIT – транзитные пакеты, которые не предназначены для данного хоста и не созданы данным хостом. LOCAL_BROADCAST – broadcast в локальной подсети (т.е. без учета универсальных broadcast 255.255.255.255, 0.0.0.0). Данное значение используется только для входящих пакетов. LOCAL_UNICAST – обычные пакеты, принятые/отправленные

[20] Для цепочки фильтров IPsec правил используется особая интерпретация. См. описание IPsecPolicy.

[21] Для цепочки фильтров IPsec правил используется особая интерпретация. См. описание IPsecPolicy.

	хостом, на котором загружена LSP. LOCAL_MISDICRECTED – пакеты, принятые/отправленные с интерфейса, на котором адрес получателя/отправителя не зарегистрирован.
Значение по умолчанию	любой тип пакетов.

Атрибут Action

Атрибут Action задает действие, которое должно быть применено к пакету при выполнении условий срабатывания правила.

Синтаксис	Action = **PASS | DROP |** STRING
Значение	PASS – прекращается поиск правил фильтрации, выполняется действие, описанное в `ExtendedAction`. Если `ExtendedAction` отсутствует, пакет пропускается для дальнейшей обработки. DROP – прекращается поиск правил фильтрации, не выполняется действие, описанное в `ExtendedAction`, пакет уничтожается. STRING – в кавычках указывается строчка и в случае срабатывания данного правила должен продолжиться поиск правил, начиная с того, у которого значение атрибута `Label` совпадает с указанной здесь строчкой. Правило фильтрации, на который происходит переход, должно присутствовать в том же списке правил (FilterChain) и располагаться в списке после фильтра, откуда происходит переход.
Значение по умолчанию	PASS.

Атрибут ExtendedAction

Атрибут ExtendedAction задает дополнительные условия для срабатывания правила и/или дополнительные действия, которые должны быть применены при выполнении условий срабатывания правила. Условия (действия) задаются в виде синтаксической конструкции "процедура". То есть указывается имя и именованные параметры в угловых скобках.

Синтаксис	ExtendedAction = inspect_tcp <…> ExtendedAction = inspect_ftp <…> ExtendedAction = tcp_flags <…> ExtendedAction = classify_mark <...> ExtendedAction = ipsec <...> ExtendedAction = bit_check <…>
Значение **inspect_tcp** – отслеживает состояние TCP-соединения, делает некоторые проверки на корректность заголовка, меняет время жизни записи о соединении в соответствии с текущим состоянием соединения. Для пропуска пакетов в обе стороны, добавляются дополнительные правила фильтрации во входящую и исходящую цепочки правил интерфейса, на котором сработала процедура tcp. Дополнительные правила удаляются вместе с записью о соединении. Для совместимости с IOS CBAC на остальные интерфейсы, где присутствуют цепочки фильтрации, добавляются правила для пропуска пакетов по данному соединению. При этом	

обновление записи происходит только при обработке пакета на том интерфейсе, где создался контекст.

inspect_ftp – дополнительно отслеживает некоторые команды FTP, создает правила для пропуска соединения для данных FTP, определяет и блокирует некоторые подозрительные команды, которые могут являться атакой на FTP сервер.

Параметры для inspect_tcp и inspect_ftp

Имя параметра	Тип	Значения	По умолчанию
flags	список значений ЦЕЛОЕ32	AUDIT, NOALERT	включены предупреждения, отключен аудит
timeout	ЦЕЛОЕ32		берется из FirewallParameters.TCPEstablishedTimeout

AUDIT – формировать сообщения при закрытии состояния со статистической информацией.

NOALERT – не формировать сообщения о потенциальных атаках (попытках взлома).

timeout – время хранения информации о неактивном соединении, этот параметр переопределяет время жизни установившегося соединения.

tcp_flags – дополнительная фильтрация пакетов по флагам TCP-заголовка, без сохранения какой-либо информации о соединении. Правило, в котором присутствует tcp_flags, считается подходящим, только если протокол TCP и флаги TCP-заголовка пакета соответствуют заданным параметрам.

Параметры для inspect_tcp и inspect_ftp

Имя параметра	Тип	Значения	По умолчанию
set	список значений ЦЕЛОЕ32	SYN, FIN, ACK, PSH, RST, URG	нет требований
clear	список значений ЦЕЛОЕ32	SYN, FIN, ACK, PSH, RST, URG	нет требований
any_set	список значений ЦЕЛОЕ32	SYN, FIN, ACK, PSH, RST, URG	нет требований
any_clear	список значений ЦЕЛОЕ32	SYN, FIN, ACK, PSH, RST, URG	нет требований

set – флаги, которые обязательно должны быть выставлены.

clear – флаги, которые должны быть сброшены.

any_set – любой из указанных флагов может быть выставлен для совпадения.

any_clear – любой из указанных флагов может быть сброшен для совпадения.

Флаги задаются константами, значение которых соответствует кодированию в заголовке TCP. Если флаги заданы списком, значения объединяются операцией "логическое или". Можно задать несколько флагов сразу одним числом. Например, следующие записи эквивалентны:

```
14h
ACK, RST
4, 10h
```



```
4, 4, 4, 16
```

Флаги, выставленные в TCP-заголовке пакета, должны совпадать с флагами, заданными по set, clear и одному из any_set или any_clear.

classify_mark – проверяет и/или выставляет TOS-байт в IP-пакетах.

Параметры для classify_mark

Имя параметра	Тип	Значения	По умолчанию
tos_set	ЦЕЛОЕ32	0-255	0
tos_set_mask	ЦЕЛОЕ32	0-255	0, значение TOS-байта не меняется
tos_match	список значений ЦЕЛОЕ32	0-255	байт TOS не влияет на совпадения фильтра
tos_match_mask	ЦЕЛОЕ32	0-255	маска должна быть отлична от нуля, если список tos_match не пуст

tos_set, tos_set_mask – если маска не нулевая, то в TOS-байте заголовка пакета будут выставлены биты, соответствующие tos_set.

tos_match, tos_match_mask – задают дополнительные ограничения на совпадение фильтра. Фильтр будет считаться подходящим только в том случае, если одно из значений tos_match совпадет со значением TOS-байта пакета в битах, ограниченных tos_match_mask.

ipsec – указывает, что пакет должен быть обработан с помощью IPsec.

Параметры для ipsec

Имя параметра	Тип	Значения	По умолчанию
sa	список IPsecAction		обязательное поле
fallback_action	ЦЕЛОЕ32	REQUEST_SA,PASS,DROP	REQUEST_SA
sa_requests_max	ЦЕЛОЕ32		8
packets_waiting_max	ЦЕЛОЕ32		8

sa – список IPsec-правил, которые могут быть использованы для создания SA, прикрепленных к данному правилу. Инициатор всегда использует первое IPsecAction

fallback_action – действие, выполняемое в случае отсутствия SA. По умолчанию – REQUEST_SA.

REQUEST_SA – посылать запрос в демон, ставить пакет в очередь

PASS – пропускать пакет без IPsec-обработки

DROP – уничтожать пакет.

Входящие пакеты, попадающие на действие с флагом REQUEST_SA, также будут уничтожены.

sa_requests_max – максимальное количество неотвеченных запросов на создание SA bundle, отправленных в демон по данному правилу.
Есть и общее ограничение на количество запросов – значение можно задать через drv_mgr – параметр ipsec_breq_max (значение по умолчанию – 1000).
Текущее количество запросов доступно через drv_mgr – параметр

ipsec_breq_count.

packets_waiting_max – размер очереди в пакетах, ожидающих приход SA bundle.

bit_check – задает дополнительную фильтрацию по любым значимым полям IP-заголовка пакета и полям данных. Поля задаются в виде диапазона битов.

Параметры для bit_check

Имя параметра	Тип	Значения	По умолчанию
origin		IP_HDR – смещение считается от начала пакета (начала IP-заголовка) или IP_DATA – смещение считается начиная с первого байта данных после IP-заголовка.	IP_HDR
bit-range	диапазон битов		
operation		EQUAL (равно), LESS (меньше), GREATER (больше)	EQUAL
value	неотрицательное число		

origin – начальное смещение для bit-range.

bit-range – диапазон битов, которые проверяется. Задается в формате bit_offset1..bit_offset2, где bit_offset – неотрицательные смещения в пакете относительно origin в битах.
Допускается, чтобы bit_offset1 и bit_offset2 совпадали, но второе смещение должно быть не меньше первого. Диапазон должен закрывать не более 32 бит, следовательно, разница bit_offset2-bit_offset1 не должна быть больше 31.

operation – операция сравнения. Операция выполняется над значением, извлеченным из пакета по адресу bit-range и значением value. Данные пакета интерпретируются как неотрицательное число, bit_offset1 является старшим битом числа, bit_offset2 – младшим битом числа.

value – значение, с которым сравниваются данные пакета.

Если описано несколько операций сравнения, то они выполняются последовательно. Если на какой-то из операций условия не совпали, пакет считается неподходящим под условия bit_check. Так, если необходимо, можно проверить длину IP-пакета, а потом производить сравнение данных за пределами IP-заголовка.

bit_check влияет именно на совпадение фильтра, а не приводит к каким-то дополнительным действиям, если совпадение обнаружено. Таким образом, поля Action, Log интерпретируются после проверки bit_check.

Если смещение bit_offset2 выходит за пределы пакета, пакет будет уничтожен.

Пакеты, подвергаемые проверке bit_check, не проходят сборку (IP reassembly). Но если важно, чтобы пакет не был собран до выполнения bit_check, необходимо помещать фильтры с bit_check вначало цепочки фильтров – другие фильтры могут вызывать сборку пакетов (например, фильтрация по TCP или UDP портам). Кроме того, действия inspect_tcp или inspect_ftp могут привести к сборке пакета, даже если фильтры с этими действиями стоят в цепочке позже, чем bit_check.

Пример

Ниже в первом правиле задано - не пропускать пакеты, у которых длина заголовка пакета больше 5: это 4-битовое поле (4..7) Header length имеет значение больше 5. Во втором правиле указано - уничтожать пакеты протокола 17, у которых номер порта "Destionation Port" свыше 300, а значение "Destination


```
IP" - 7.7.7.212.
    Filter (
        ExtendedAction = bit_check[[4..7, GREATER, 5]]
        Action = DROP
        LogEventID = "\"options in IP header\""
    ), Filter (
        ProtocolID = 17
        ExtendedAction = bit_check
    [[128..159,070707D4h],[IP_DATA,16..31,LESS,300]]
        LogEventID = "\"special packet\""
        Action = DROP
)
```

Допустимые значения ExtendedAction для разных применений FilterChain приведены в нижеследующей таблице.

	inspect_tcp inspect_ftp	tcp_flags	classify_mark	bit_check	ipsec
NetworkInterface.InputFilter NetworkInterface.OutputFilter	+	+	+	+	–
IPsecAction.InputFilter IPsecAction.OutputFilter	–	+	+	+	–
NetworkInterface.InputClassification NetworkInterface.OutputClassification	–	+	+	+	–
NetworkInterface.IPsecPolicy	–	–	–	–	+

Если ExtendedAction не соответствует применению FilterChain, выдается ошибка разбора конфигурации.

Значение по умолчанию отсутствуют специальные действия над пакетом.

Атрибут LogEventID

Атрибут LogEventID задает идентификатор, который передается в сообщения аудита, связанные с данным фильтром. При наличии LogEventID сообщения о подпадании пакета под фильтр отправляются в журнал аудита.

Синтаксис	LogEventID = СТРОКА
Значение	фактически LogEventID можно считать именем фильтра, но требования на уникальность отсутствуют.
Значение по умолчанию	неименованное правило.

Атрибут Schedule

Атрибут Schedule задает временные диапазоны, в которые данный фильтр активен. В другое время фильтр не активен и не учитывается при фильтрации пакетов.

Деактивация фильтра ExtendedAction = inspect_* приводит к прекращению отслеживания соединений по данному правилу и уничтожению динамически созданных фильтров.

Нельзя указывать временные диапазоны для фильтров, привязанных к NetworkInterface.IPsecPolicy.

Синтаксис	Schedule = `Schedule`
Значение по умолчанию	нет ограничений по времени действия фильтра.

Атрибут Log

Атрибут Log включает/выключает генерацию данных аудита по данному фильтру. Если генерация данных аудита включена и пакет подпадает под фильтр, то в журнал аудита будет передано об этом сообщение.

Синтаксис	Log = **TRUE | FALSE**
Значение	TRUE – включает генерацию данных аудита по данному фильтру FALSE –выключает аудит по данному фильтру.
Значение по умолчанию	FALSE.

Атрибут Label

Атрибут Label задает метку, которая при совпадении со строкой в атрибуте Action другого правила, говорит о том, что с данного правила можно продолжить поиск правил (см. атрибут Action параметр STRING).

Синтаксис	Label = СТРОКА
Значение по умолчанию	метка отсутствует

11.26 Структура Schedule

Структура Schedule задает расписание действия правил фильтрации.

Структуре может быть присвоено имя, что позволяет делать ссылки из нескольких правил на одно расписание.

Имя структуры	Schedule
Атрибуты	Periods

Атрибут Periods

Атрибут Periods задает список временных интервалов. Если текущее время попадает в заданный интервал, то правило, для которого задан интервал, в данный момент считается активным.

Если в списке есть пересекающиеся интервалы с противоречащими действиями (`Period.Action`), то используется интервал, который записан раньше в списке.

Синтаксис	Periods* = `Period`
Значение по умолчанию	ограничение по времени не применяется.

11.27 Структура Period

Структура Period описывает временной диапазон – периодический или абсолютный.

Если атрибуты Start или End содержат абсолютную дату (тип ДАТА представляется тремя целыми числами без знака, разделенными символом '/' – число/месяц/год), то интервал считается абсолютным, в противном случае – он периодический. Для абсолютных интервалов допускается только указание абсолютной даты и времени. Буквенные обозначения дней недели и месяцев запрещены.

Время соответствует локальному времени, установленному в операционной системе.

Интервалы отслеживаются с максимальным опозданием в 1 минуту, но в случае крайней загруженности ОС (т.е. невозможности выполнения приложений в течение длительного времени), отслеживание графиков может задерживаться более чем на 1 минуту.

Имя структуры Period

Атрибуты Start
End
Action

Атрибут Start

Атрибут Start задает начало временного интервала.

Синтаксис	Start = ЦЕЛОЕ32, ДАТА, ВРЕМЯ
Значение	SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, JANUARY, FEBRUARY, MARCH, APRIL, MAY, JUNE, JULY, AUGUST, SEPTEMBER, OCTOBER, NOVEMBER, DECEMBER, дата, время, день месяца (1..31), END_OF_MONTH (последний день месяца (для релиза 14101 это значение недоступно)). Периодические интервалы Для периодических интервалов атрибут Start является обязательным и определяющим для указания временного интервала. Действует следующий порядок: если в Start указан месяц, то периодичность год, если в Start указан день месяца, периодичность – месяц, если в Start указан день недели, периодичность – неделя, если в Start указано только время, период – день. Значение, указанное в Start, может быть больше значения, указанного в End. При этом интервал инвертируется – End переносится на следующий год, месяц, неделю или день в зависимости от периодичности. Допускается указание нескольких значений, например, месяца и дня месяца. Но делать это надо с осторожностью. Если, например, указанного числа в месяце нет, то период будет пропущен. День недели нельзя указывать вместе с месяцем или числом одновременно. Остальные комбинации допускаются. Действуют следующие правила дополнения: Если время не указано, то берется начало дня (00:00). Если месяц указан, но не указано число, то берется первое число.

Значение по умолчанию	для абсолютных интервалов – начало летоисчисления. Для периодических интервалов поле обязательно.

Примеры периодических интервалов:

Period a (Start = 2, JANUARY End = 10) # со второго по десятое января каждого года

Period b (Start = 12:00 End = 14:00) # каждый день с 12 до двух дня

Period c (Start = 10, 10:00 End = 14:00) # с 10 числа каждого месяца до 14 часов #последнего дня месяца

Period d (Start = MONDAY End = FRIDAY, 17:00) # с понедельника до 17:00 пятницы #каждую неделю

Period e (Start = APRIL, 1, 15:00 End = APRIL, 1, 14:00) # весь год кроме 1 часа 1 #апреля

Period f (Start = MONDAY, 18:30 End = 17:30) # с понедельника 18:30 по следующий #понедельник 17:30

Примеры абсолютных интервалов:

Period a (Start = 23/12/2009 End = 8/9/2016, 22:30)

Period b (End = 08/09/ 2007, 2:30)

Period c (Start = 2:00, 5 /6/15)

Атрибут End

Атрибут End задает конец временного интервала.

Синтаксис	End = ЦЕЛОЕ32, ДАТА, ВРЕМЯ
Значение	SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, JANUARY, FEBRUARY, MARCH, APRIL, MAY, JUNE, JULY, AUGUST, SEPTEMBER, OCTOBER, NOVEMBER, DECEMBER, дата, время, день месяца (1..31), END_OF_MONTH (последний день месяца (для релиза 14101 это значение недоступно)). Если указано время, то включается последняя минута. Так прекращение интервала для End = 14:20 будет не ранее 14:21. Абсолютные интервалы Если не указана дата, а только время, то дата для End принимается равной дате для Start. Периодические интервалы Действуют следующие правила дополнения: если время не указано, то берется конец дня (23:59), если месяц не указан, но указан в Start – период оканчивается в месяц с именем, указанным в Start, если день месяца не указан, а в Start указан день месяца или месяц – период оканчивается в последний день месяца (в релизе 14101 надо явно указывать число в End, если в Start указан месяц или число),

	если не указан день недели, но день недели есть в Start, период оканчивается в день недели, указанный в Start[22]. Действуют следующие ограничения: день недели нельзя указывать вместе с месяцем или числом одновременно, нельзя указывать день месяца больше 28, если месяц не задан явно или месяц – февраль[23], нельзя указывать величины большего порядка, чем в Start – т.е. если в Start не указан месяц, то в End нельзя указать месяц.
Значение по умолчанию	для абсолютных интервалов отсутствие End считается отсутствием ограничения по времени. Причем если End отсутствует, Start обязательно должен быть указан. Для периодических интервалов отсутствие End интерпретируется как конец дня, если Start не содержит указание месяца и/или числа. Если в Start указан месяц и/или число, End выставляется на конец месяца.

Атрибут Action

Атрибут Action задает активность правила в указанный период.

Синтаксис	Action = **ENABLE** | **DISABLE**
Значение	ENABLE – временной интервал считается интервалом активности для правила фильтрации; DISABLE – в указанный временной интервал правило неактивно и не учитывается при фильтрации пакетов.
Значение по умолчанию	ENABLE

[22] Если End указывает на более раннее время дня, чем Start, то интервал будет длиться до соответствующего дня следующей недели.

[23] Допустимо указывать 29 февраля, как отдельный день – Start и End оба указывают на 29 февраля. В этом случае период будет активен один день за 4 года.

11.28 Формат задания DistinguishedName (GeneralNames) в LSP

Текстовое представление DN

Текстовое представление DistinguishedName (GeneralNames), далее просто имени, задается в соответствии с RFC2253:

```
distinguishedName = [name]; may be empty string

name  name-component *("," name-component)

name-component = attributeTypeAndValue *("+" attributeTypeAndValue)

attributeTypeAndValue = attributeType "=" attributeValue

attributeType = (ALPHA 1*keychar) / oid
keychar = ALPHA / DIGIT / "-"

oid = 1*DIGIT *("." 1*DIGIT)

attributeValue = string

string = *( stringchar / pair )
         / "#" hexstring
         / QUOTATION *( quotechar / pair ) QUOTATION; only from v2

quotechar = <any character except "\" or QUOTATION >

special = "," / "=" / "+" / "<" /  ">" / "#" / ";"

pair = "\" ( special / "\" / QUOTATION / hexpair )
stringchar =<any character except one of special, "\" or QUOTATION>

hexstring = 1*hexpair
hexpair = hexchar hexchar

hexchar = DIGIT / "A" / "B" / "C" / "D" / "E" / "F"
          / "a" / "b" / "c" / "d" / "e" / "f"

ALPHA = <any ASCII alphabetic character>;(decimal 65-90 and 97-122)
DIGIT = <any ASCII decimal digit>       ;(decimal 48-57)
QUOTATION = <the ASCII double quotation mark character '"' decimal
34>
```

Дополнения и отступления от RFC2253

Имеются следующие дополнения и отступления от RFC2253:

- символ "/" является разделителем компонент имени, т.е. допустим следующий синтаксис:

  ```
  name = name-component *("/" name-component)
  ```

  - для того, чтобы использовать этот символ как значащий, его необходимо проэскейпить.
- распознаются следующие сокращения типов атрибутов (attributeType) DistinguishedName:

X.500 Attribute Type	Сокращение
countryName	C
stateName	ST
localityName	L

organizationName	O
organizationalUnitName	OU
commonName	CN
title	T
surname	SN
givenName	GN
initials	I
streetAddress	STREET
nameQualifier	NQ
generationQualifier	GQ
userid	UID
domainComponent	DC

- регистр, в котором записано сокращение, не имеет значения.

- Строковое задание GeneralNames сведено к синтаксису, описанному в RFC2253. Распознаются следующие сокращения типов атрибутов имени GeneralNames:

Тип атрибута	Сокращение
otherName	OTHERNAME
rfc822Name	EMAIL
dNSName	DNS
directoryName	DN
uniformResourceIdentifier	URI
iPAddress	IP
registeredID	RID

  - регистр, в котором записано сокращение, не имеет значения
  - задание атрибутов `x400Address и ediPartyName` в строковом представлении не поддерживается.
  - Согласно RFC2253 символы `'"'(кавычки) и '\'(back-slash)` являются служебными. Согласно описанию Терминального символа СТРОКА, при задании любого строкового значения в LSP указанные символы так же используются как служебные. Поэтому:
    - каждая отдельно стоящая кавычка в строковом представлении должна быть дополнена слева символом `'\'` в LSP
    - каждое сочетание `'\"'` в строковом представлении должно быть дополнено слева `'\\'` в LSP.

Примеры

Имя в сертификате	Строковое представление	В LSP
`O=Sergey, Danila and company`	`O=Sergey\, Danila and company`	`Subject="O=Sergey\, Danila and company"`
`O=JSC "Horns and hoofs"`	`O=JSC \"Horns and hoofs\"`	`Subject="O=JSC \\\"Horns and hoofs\\\""`
`CN=Device#4`	`CN="Device#4"`	`Subject="CN=\"Device#4\""`

11.29 Работа с сертификатами

Отсылка локального сертификата

Для отсылки локального сертификата партнеру по IKE в LSP-конфигурации необходимо:

в структуре `AuthMethodGOSTSign` задать атрибут `SendCertMode` со значением:

- ALWAYS – всегда отсылать локальный сертификат
- CHAIN – всегда отсылать локальный сертификат, CA сертификат и промежуточные CA сертификаты.

Получение сертификата партнера

Сертификат партнера можно получить либо по протоколу IKE, либо по протоколу LDAP.

Сначала агент пытается получить сертификат партнера по IKE, если партнер не прислал сертификат, а прислал свой идентификатор. Агент по этому идентификатору ищет сертификат партнера сначала в своей базе Продукта, если не нашел, то продолжает поиск на LDAP-сервере.

Получение сертификата партнера по IKE

Для получения сертификата партнера по IKE в LSP-конфигурации нужно:

- в структуре `AuthMethodGOSTSign` задать атрибут `SendRequestMode` со значением ALWAYS – всегда запрашивать сертификат партнера
- в конфигурации партнера в структуре `AuthMethodGOSTSign` задать атрибут `SendCertMode` со значением:
  - ALWAYS – высылать сертификат
  - CHAIN – высылать локальный сертификат, CA сертификат с цепочкой промежуточных CA.

Получение сертификата партнера по LDAP

В этом случае партнер присылает свой идентификатор, а агент по Subject будет искать сертификат партнера на LDAP-сервере. Для прохождения LDAP-пакетов до LDAP-сервера необходимо в LSP-конфигурации задать соответвующий фильтр:

- задать структуру `LDAPSettings` с IP-адресом LDAP-сервера:
  - если прислан идентификатор типа DN:
    - агент по Subject ищет сертификат партнера сначала в своей базе Продукта, а затем на LDAP-сервере

- если прислан идентификатор другого типа:
  - для получения Subject в локальной конфигурации задаются атрибуты `RemoteID`, `RemoteCredential`, `DoNotMapRemoteIDToCert`
    - если DoNotMapPeerIDToCert = TRUE, то Subject будет составляться из RemoteCredential
    - если DoNotMapPeerIDToCert = FALSE, то Subject будет составляться из RemoteCredential и RemoteID.
  - по составленному Subject агент ищет сертификат партнера сначала в своей базе Продукта, а затем на LDAP-сервере.

Проверка сертификата по CRL

Для проверки сертификата партнера по CRL в LSP-конфигурации нужно:

- в структуре `GlobalParameters` задать атрибут `CRLHandlingMode`, при значениях этого атрибута:
  - `optional` – используется действующий CRL из базы Продукта
  - `enable` и `best_effort` – действующий CRL может быть получен по LDAP.

Для получения CRL с LDAP-сервера сначала проверяется поле CDP в проверяемом сертификате, если поле CDP отсутствует, то в конфигурации должна быть задана структура LDAPSettings с адресом LDAP-сервера. В базу Продукта с LDAP-сервера загружается действующий CRL и по нему проверяется сертификат партнера.

Для прохождения LDAP-пакетов до LDAP-сервера необходимо в политике задать соответвующий фильтр.

12. Требования к внешним мерам безопасности

12.1 Физические меры безопасности

Помещения предприятия должны удовлетворять следующим требованиям:

- Обеспечение круглосуточной охраны корпусов предприятия;
- Обеспечение контроля внешнего периметра и внутренних помещений (видеонаблюдение);
- Обеспечение пропускного режима;
- Рядом с окнами помещений не должно быть пожарных лестниц и водосточных труб;
- Двери должны быть прочными и оборудованы надежными механическими замками;
- Оборудование помещений системой пожарной сигнализации;
- Ведение Журнала выдачи ключей от входных дверей в офисы, в котором регистрируется время сдачи и выдачи ключей, фамилия сотрудника, взявшего или сдавшего ключ дежурному вахтеру по зданию;
- Наличие опечатываемого сейфа, оборудованного двумя внутренними замками, для хранения СКЗИ, тестовых ключей, эталонных CD дисков с продуктом, другой конфиденциальной информации. Для сейфа должно быть два ключа – основной ключ хранится у сотрудника, отвечающего за СКЗИ, а дубликат в опечатанном его личной печатью пенале в сейфе Генерального директора.

12.2 Процедурные меры безопасности

К безопасной эксплуатации продукта и обращения с СКЗИ предъявляются следующие требования:

- При приеме на работу сотрудники подписывают Обязательство о неразглашении сведений, составляющих коммерческую тайну организации
- Перечень сведений, составляющих коммерческую тайну организации, утверждается Генеральным директором;
- На предприятии должна быть разработана Инструкция по обращению с сертифицированными ФСБ/ФАПСИ шифровальными средствами (средствами криптографической защиты информации);
- Ведение Журнала учета СКЗИ, тестовых ключей на предприятии;
- Ведение Журнала учета обращения эталонных CD дисков на предприятии.

12.3 Технические меры безопасности

К техническим мерам безопасности предъявляются следующие требования:

- Доступ к персональным компьютерам и средствам вычислительной техники осуществляется на основе логического имени и пароля пользователя в рамках операционных систем;
- Создание инсталляционного пакета для каждого пользователя и управление политикой безопасности пользователя осуществляется только администратором в соответствии с политикой безопасности предприятия;

- Администратор должен быть аутентифицирован и идентифицирован перед доступом к продукту с целью администрирования. Аутентификация осуществляется на основе пароля, вводимого с клавиатуры, не отображаясь на экране монитора, и выполняется операционной системой;
- Доставка контейнера с криптографическим ключом сертификата пользователя осуществляется только по доверенному каналу связи;
- Для защиты от вирусов клиентских компьютеров и серверов используются антивирусные продукты.

