127018, Москва, Сущёвский Вал, 18
Телефон: (495) 995 4820
Факс: (495) 995 4820
https://CryptoPro.ru
E-mail: info@CryptoPro.ru

Средство Криптографической Защиты Информации

КриптоПро JCP

Версия 2.0 R4

Руководство администратора безопасности

ЖТЯИ.00091-04 90 01
Листов 21

2020 г

Содержание

Аннотация

Настоящее руководство содержит общее описание средства криптографической защиты информации (СКЗИ) КриптоПро JCP версии 2.0 R4, его состав, ключевую систему, рекомендации по размещению технических средств, использующих СКЗИ, рекомендации по проверке целостности установленного ПО СКЗИ, по использованию СКЗИ в различных автоматизированных системах и средствах вычислительной техники.

Инструкции администраторам безопасности и пользователям различных автоматизированных систем, использующих СКЗИ КриптоПро JCP версии 2.0 R4, должны разрабатываться с учетом требований настоящего Руководства.

Криптопровайдер КриптоПро JCP версии 2.0 R4 является средством криптографической защиты информации (СКЗИ КриптоПро JCP версии 2.0 R4), реализующим российские криптографические алгоритмы и функционирующим под управлением виртуальной Java-машины.

Список определений и сокращений

CRL	Список отозванных сертификатов (Certificate Revocation List)
АРМ	Автоматизированное рабочее место
АС	Автоматизированная система
ГМД	Гибкий магнитный диск
ДСЧ	Датчик случайных чисел
HDD	Жесткий магнитный диск (Hard Disk Drive)
НСД	Несанкционированный доступ
ОС	Операционная система
ПАК	Программно-аппаратный комплекс
ПО	Программное обеспечение
Регистрация	Присвоение определенных атрибутов (адреса, номера ключа, прав использования и т.п.) абоненту
Регламент	Совокупность инструкций и другой регламентирующей документации, обеспечивающей функционирование автоматизированной системы во всех режимах
СВТ	Средства вычислительной техники
Сертификат	Электронный документ, подтверждающий принадлежность открытого ключа или ключа проверки электронной подписи и определенных атрибутов конкретному абоненту
Сертификация	Процесс изготовления сертификата открытого ключа или ключа проверки электронной подписи абонента в центре сертификации
СКЗИ	Средство криптографической защиты информации
СОС	Список отозванных сертификатов (Certificate Revocation List)
СС	Справочник сертификатов открытых ключей и ключей проверки электронной подписи. Сетевой справочник
СФК	Среда функционирования комплекса
ЦС	Центр Сертификации (Удостоверяющий Центр)
ЦР	Центр Регистрации
ЭД	Электронный документ
ЭП	Электронная подпись

1 Назначение СКЗИ

СКЗИ КриптоПро JCP версии 2.0 R4 является криптопровайдером Java и предназначено для защиты открытой информации в информационных системах общего пользования (вычисление/проверка электронной подписи) и защиты конфиденциальной информации, не содержащей сведений, составляющих государственную тайну, в корпоративных информационных системах с выполнением следующих функций:

- авторизация и обеспечение юридической значимости электронных документов при обмене ими между пользователями посредством использования процедур формирования и проверки (с использованием сертификатов стандарта X.509 Удостоверяющего центра) электронной подписи в соответствии с отечественными стандартами ГОСТ Р 34.10-2001, ГОСТ Р 34.10-2012, ГОСТ 34.10-2018 (с использованием ГОСТ Р 34.11-94, ГОСТ Р 34.11-2012, ГОСТ 34.11-2018);
- обеспечение конфиденциальности и контроля целостности информации посредством ее шифрования и имитозащиты, в соответствии с отечественным стандартом ГОСТ 28147-89;
- контроль целостности системного и прикладного программного обеспечения для его защиты от несанкционированного изменения или от нарушения правильности функционирования;
- управления ключевыми элементами системы в соответствии с регламентом;
- обеспечение аутентификации связывающихся сторон, конфиденциальности и целостности пересылаемой информации с использованием сертификатов стандарта X.509;
- установление аутентичного защищенного соединения с использованием протокола КриптоПро JTLS;
- обеспечение конфиденциальности и контроля целостности и авторизация файлов и информационных сообщений;
- обеспечение аутентификации пользователя в домене Windows.

Подробное описание реализуемых механизмов защиты, алгоритмов и протоколов, а также сведения о составе и назначении компонент СКЗИ см. в ЖТЯИ.00091-04 94 01. КриптоПро JCP. Описание реализации.

Перечни поддерживаемых СКЗИ Java-машин и программно-аппаратных сред см. в разделе 3 ЖТЯИ.00091-04 30 01. КриптоПро JCP. Формуляр.

2 Структура СКЗИ

Основной архитектурной особенностью ПО СКЗИ КриптоПро JCP версия 2.0 R4 является то, что среда функционирования (СФ) СКЗИ не имеет непосредственного доступа к ключевой и криптографически значимой информации. Все операции с закрытыми ключами, незавершенными значениями хэш-функций и т. п. осуществляется недоступные пользователю объектов, операции экспорта отсутствуют.

Общая структура СКЗИ представлена на рис. 1.

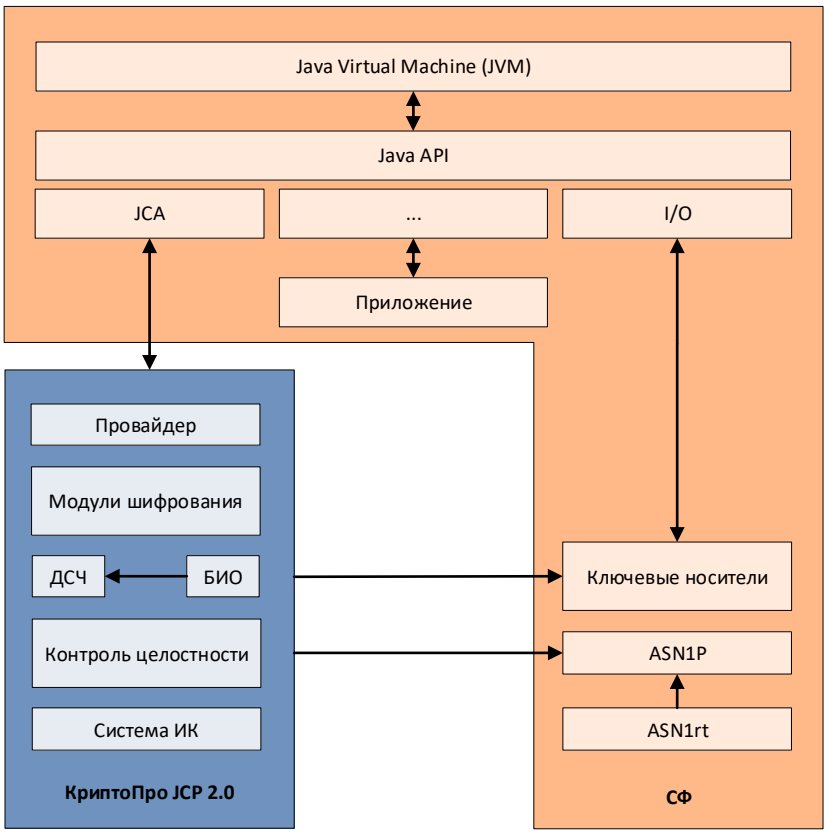

Рисунок 1. Структура СКЗИ

Подробный состав и архитектура СКЗИ КриптоПро JCP версия 2.0 R4 и СФ описаны в ЖТЯИ.00091-04 94 01. КриптоПро JCP. Описание реализации.

3 Совместимость с продуктами КриптоПро

КриптоПро JCP версии 2.0 R4 поддерживает:

- чтение криптопровайдером КриптоПро JCP версии 2.0 R4 ключей, созданных при помощи криптопровайдера КриптоПро CSP версии 4.0 и выше, с ключевых носителей, и наоборот;
- копирование с ключевых носителей криптопровайдером КриптоПро JCP версии 2.0 R4 ключей, созданных при помощи криптопровайдера КриптоПро CSP версии 4.0 и выше, на другие ключевые носители, и наоборот;
- чтение сертификатов удостоверяющего центра КриптоПро УЦ 2.0;
- чтение сертификатов удостоверяющего центра Microsoft CA с установленным на нем СКЗИ КриптоПро CSP версии и выше;
- создание при помощи класса GostCertificateRequest запроса для удостоверяющего центра Microsoft CA с установленным на нем КриптоПро CSP версии 4.0 и выше, а также для удостоверяющего центра КриптоПро УЦ 2.0;
- создание при помощи утилиты keytool запроса для удостоверяющего центра Microsoft CA с установленным на нем КриптоПро CSP версии 4.0 и выше, а также для удостоверяющего центра КриптоПро УЦ 2.0.

СКЗИ КриптоПро JCP версии 2.0 R4 совместимо с КриптоПро CSP версии 4.0 и выше по выполняемым криптографическим функциям, форматам данных и ключам со следующими ограничениями:

- не поддерживается атрибут CRYPT_USER_PROTECTED;
- не поддерживаются контейнеры с разделённым хранением закрытого ключа на разных ключевых носителях;
- не поддерживается управление контейнером "по умолчанию" на ключевом носителе;
- недопустима одновременная работа КриптоПро CSP и КриптоПро JCP с одним и тем же контейнером в ОС Windows.

Если Вы используете на одном компьютере и КриптоПро JCP версии 2.0 R4, и КриптоПро CSP, необходимо настроить пути к ключам в соответствии в версией СКЗИ КриптоПро CSP. Это можно сделать средствами контрольной панели, из командной строки или программно.

СКЗИ КриптоПро JCP версии 2.0 R4 совместимо с КриптоПро УЦ 2.0 по форматам данных со следующим ограничением:

- требуется использовать дополнительный CertPathValidator (см. «Руководство программиста») в случае использования COC (CRL) и без регламентного изменения различительного имени при плановой смене ключа УЦ (X.500 DN) (расширение Microsoft szOID_CERTSRV_CA_VERSION "1.3.6.1.4.1.311.21.1" не поддерживается стандартными CertPathValidator и CertPathBuilder)

СКЗИ КриптоПро JCP версии 2.0 R4 совместимо с КриптоПро PDF при условии использования PDF с расширенными правами. Такой PDF можно сделать, например, в Adobe Acrobat Pro. Для создания подписи в файле формата PDF с расширенными правами с помощью СКЗИ КриптоПро JCP версии 2.0 R4 можно, например, использовать свободную библиотеку iText. Такая подпись будет видна в PDF-файле при просмотре через Adobe Reader, а если установлены КриптоПро CSP и КриптоПро PDF, то она сможет проверяться.

4 Совместное использование SunIdM и СКЗИ

В случае использования Java-машин версии 1.7 или 1.8:

Специалистами компаний Sun Microsystems и КРИПТО-ПРО были совместно выполнены интеграция и тестирование совместной работы Sun Java System Identity Manager со средством криптографической защиты КриптоПро JCP, функционирующем под управлением Java-машин версий 1.7 и 1.8.

Произведенная интеграция позволяет использовать реализованные в КриптоПро JCP российские криптографические алгоритмы ГОСТ Р 34.10-2001, ГОСТ Р 34.10-2012 и ГОСТ Р 34.11-94, ГОСТ Р 34.11-2012 для заверения российской электронной подписью заявок на получение доступа к ресурсам, используя стандартные интерфейсы в рамках автоматизированного бизнес-процесса системы Sun Java System Identity Manager. Применение российского средства ЭП и сертификатов ключей проверки ЭП гарантирует, что заявка исходит именно от того лица, которое уполномочено на такие действия (руководители, сотрудники службы безопасности и т.д.). Формирование заявок и ЭП производится непосредственно с помощью разработанных web-форм для согласующих лиц. Информация об одобрении заявки, вместе с ЭП, хранится в репозитории Identity Manager и может быть предоставлена в виде отчета, необходимого для проведения аудитов по информационной безопасности.

Для интеграции Sun IdM и КриптоПро JCP:

1) Установите SunIdM согласно документации.

Примечание: пункт 3.11 ("...log in to Identity Manager...") в процессе установки преждевременный, перед входом предварительно надо настроить права доступа по пункту 5.

2) Установить на сервер КриптоПро JCP в соответствии с рекомендациями эксплуатационной документации.

3) Настроить конфигурацию сервера по документации Configuring Digitally Signed Approvals.

- Для установки `security.nonrepudiation.signedApprovals=true` войдите на отладочную страницу Identity Manager `http://PathToIDM/debug`. Загрузится страница с системными настройками. У пункта "List Objects"надо выбрать из выпадающего меню "Configuration" и нажать "List Objects" , появится страница "List Objects of type: Configuration". У пункта "System Configuration" надо выбрать "Edit" , появится файл содержащий системную конфигурацию. Его надо отредактировать, установив `signedApprovals=true`.

```
<Attribute name='nonrepudiation'>
    <Object>
        <Attribute name='signedApprovals'>
        <Boolean>true</Boolean>
     </Attribute>
    </Object>
</Attribute>
```

- Установка сертификатов из интерфейса администратора.

Примечание: согласно документации, "From the Administrator interface, select Configure, and then select Certificates". В седьмой версии пункт меню Sertificates реально находится в меню Security.

- Подпись applets/ts1.jar с использованием jarsigner.

Именно `ts1.jar` проставляет подпись на клиенте. Подпись не на ГОСТ алгоритмах, нужна для того, чтобы разрешить выполнение кода на клиентской машине.

4) Добавить алгоритмы КриптоПро JCP в IdM.

В файле `samples_src.jar` в каталоге SunIdM находятся две модифицированные формы IdM: "Approval Form.xml" и "Work Item Configuration.xml". В них добавлены параметры `supportedKeyStoreTypes` и `keytypeSignatureMapping` для полей типа `TransactionSigner`.

```
<Property name='keytypeSignatureMapping' value='DSA=SHA1withDSA,RSA=SHA1withRSA,
RSA=MD5withRSA,RSA=MD2withRSA,GOST3410=GOST3411withGOST3410EL'/>
<Property name='supportedKeyStoreTypes' value='JKS,PKCS12,HDImageStore'/>
```

Надо установить в поле `supportedKeyStoreTypes` типы хранилищ ключей, которые будут использованы на клиентской машине для подписи. HDImageStore добавлен для примера, установите типы хранилищ, которые будут реально использоваться. Эти формы необходимо по очереди импортировать в конфигурацию через web-интерфейс, Configure → Import Exchange File. Затем перезагрузите IdM.

5) Установить КриптоПро JCP на клиенте.

Подготовьте хранилища и ключи, которые будут использоваться для подписи. Указания "Obtain a certificate and private key, and then export them to a PKCS#12 keystore" необходимо игнорировать. КриптоПро JCP и PKCS#12 несовместимы.

5 Разбор конфликтных ситуаций, связанных с применением ЭП

Применение электронной подписи в автоматизированной системе может приводить к конфликтным ситуациям, заключающимся в оспаривании сторонами (участниками системы) авторства и/или содержимого документа, подписанного электронной подписью.

Разбор подобных конфликтных ситуаций требует применения специального программного обеспечения для выполнения проверок и документирования данных, используемых при выполнении процедуры проверки соответствия ЭП содержимому электронного документа.

Разбор конфликтной ситуации заключается в доказательстве авторства подписи конкретного электронного документа конкретным исполнителем.

Данный разбор основывается на математических свойствах алгоритмов ЭП, реализованных в соответствии со стандартами РФ ГОСТ Р 34.10-2001, ГОСТ Р 34.10-2012, ГОСТ Р 34.11-2012 и ГОСТ Р 34.11-94, гарантирующих невозможность подделки значения ЭП любым лицом, не обладающим закрытым ключом подписи.

При проверке значения ЭП используется ключ проверки ЭП, значение которого вычисляется по значению ключа ЭП при их формировании.

В системе должны быть предусмотрены средства ведения архивов электронных документов с ЭП и сертификатов ключей проверки ЭП.

Разбор конфликтной ситуации выполняется комиссией, состоящей из представителей сторон, службы безопасности и экспертов. Состав комиссии, порядок ее формирования, регламент работы, рассмотрение результатов определяется в приложении к Регламенту (Договору), заключаемому между участниками информационного обмена.

Оспаривание результатов работы комиссии и возмещение пострадавшей стороне принесенного ущерба выполняется в установленном действующим законодательством Российской Федерации порядке.

5.1 Порядок разбора конфликтной ситуации

Разбор конфликтной ситуации выполняется по инициативе любого участника информационного обмена и состоит из:

1) предъявления претензии одной стороны другой;
2) формирования комиссии;
3) разбора конфликтной ситуации;
4) принятия мер по урегулированию конфликта.

Разбор конфликтной ситуации проводится с использованием программного обеспечения СКЗИ КриптоПро JCP версии 2.0 R4 для электронного документа, авторство или содержание которого оспаривается.

Проверка подписанного электронного документа включает в себя выполнение следующих действий:

1) определение сертификата или нескольких сертификатов, необходимых для проверки ЭП;
2) проверка ЭП электронного документа с использованием каждого сертификата;
3) определение даты создания каждой ЭП в электронном документе;
4) проверка ЭП каждого сертификата, путем построения цепочки сертификатов до сертификата Главного ЦС;
5) проверка действительности сертификатов на текущий момент времени;
6) проверка действительности сертификатов на момент создания ЭП;
7) проверка отсутствия сертификатов в СОС.

При проверке ЭП документа, верификации цепочки сертификатов, отсутствии сертификата в СОС, авторство подписи под документом считается установленным.

Примечание. Несовпадение даты формирования документа и сроков действия сертификата и/или сроков действия ключа ЭП не влияют на определение авторства документа. В таком случае можно сделать предположение о несоблюдении пользователем Регламента (Договора) в части сроков действия ключей, сертификатов или некорректного использования сертификата в прикладном ПО.

5.2 Случаи невозможности проверки значения ЭП

При отсутствии в архиве сертификата открытого ключа (ключа проверки ЭП) пользователя, выполнившего ЭП, доказать авторство документа невозможно. В связи с этим, архив с сертификатами открытых ключей необходимо подвергать регулярному резервному копированию и хранить в течение всего установленного срока хранения.

6 Нештатные ситуации при эксплуатации СКЗИ

В табл. 1 приведен основной перечень нештатных ситуаций и соответствующие действия персонала при их возникновении.

Таблица 1. Действия персонала в нештатных ситуациях

№ п/п	Нештатная ситуация	Действия персонала
1	Эвакуация, угроза нападения, взрыва, стихийные бедствия, аварии общего характера в Центре управления ключевой системой.	• Остановить все ЭВМ. • Персонал, имеющий доступ к ключам, обязан сдать все имеющиеся у него в наличии ключевые носители администратору безопасности. • Администратор безопасности упаковывает все ключевые носители, регистрационные карточки сертификатов открытых ключей пользователей, сертификаты ключей проверки ЭП пользователей в опечатываемый контейнер, который выносит в безопасное помещение или здание. Опечатанный контейнер должен находиться под охраной до окончания действия нештатной ситуации и восстановления нормальной работы аппаратных и программных средств СКЗИ. • Администратор безопасности оповещает по телефонным каналам общего пользования всех пользователей о приостановке работы системы. • В случае наступления события, повлекшего за собой долговременный выход из строя аппаратных средств СКЗИ, администратор безопасности уничтожает всю ключевую информацию с носителей, находящихся в контейнере.
2	Компрометация одного из личных ключевых носителей.	Порядок действий при компрометации ключей описан в разделе 3 документа ЖТЯИ.00091-04 95 01. Правила пользования.
3	Выход из строя первого личного ключевого носителя.	Необходимо сообщить по телефону в УЦ о факте выхода из строя личного ключевого носителя и обеспечить его доставку в УЦ для выяснения причин выхода из строя. Для работы используется второй личный ключевой носитель.
4	Выход из строя второго личного ключевого носителя (при условии, что первый тоже вышел из строя).	Пользователь, у которого вышли из строя оба личных ключевых носителя, является в УЦ для повторной регистрации (без изменения данных регистрации).
5	Отказы и сбои в работе аппаратной части АРМ со встроенной СКЗИ.	При отказах и сбоях в работе аппаратной части АРМ со встроенным СКЗИ необходимо остановить работу, по возможности локализовать неисправность и в дальнейшем произвести ремонт в установленном порядке и, при необходимости, переустановку СКЗИ.
6	Отказы и сбои в работе средств защиты от НСД.	При отказах и сбоях в работе средств защиты от НСД, администратор безопасности, должен восстановить работоспособность средств НСД. При необходимости переустановить программно-аппаратные средства НСД.
7	Утеря личного ключевого носителя.	Утеря личного ключевого носителя приводит к компрометации хранящегося в нем ключа. Порядок действий при компрометации ключей описан в разделе 3 ЖТЯИ.00091-04 92 01. КриптоПро JCP. Правила пользования.

8	Отказы и сбои в работе программных средств вследствие не выявленных ранее ошибок в программном обеспечении.	При отказах и сбоях в работе программных средств, вследствие не выявленных ранее ошибок в программном обеспечении, необходимо остановить работу, локализовать по возможности причину отказов и сбоев и вызвать разработчика данного ПО или его представителя для устранения причин, вызывающих отказы и сбои.
9	Отказы в работе программных средств вследствие случайного или умышленного их повреждения.	При отказах в работе программных средств, вследствие случайного или умышленного их повреждения, ответственное за безопасность функционирования программных и аппаратных средств лицо обязано произвести служебное расследование по данному факту с целью установления причины отказа и восстановления правильной работы программных средств в установленном порядке.
10	Отказы в работе программных средств вследствие ошибок оператора.	При отказах в работе программных средств, вследствие ошибок оператора, оператор сообщает о данном факте лицу, ответственному за безопасность функционирования программных и аппаратных средств. Ответственный за безопасность функционирования программных и аппаратных средств дает соответствующие указания обслуживающему персоналу по восстановлению правильной работы программных средств в установленном порядке.

7 Установка ПО СКЗИ на ПЭВМ

К эксплуатации программного обеспечения, имеющего в своем составе СКЗИ, допускаются лица, прошедшие соответствующую подготовку и изучившие эксплуатационную документацию на соответствующие программные средства.

Установка СКЗИ на рабочее место пользователя может быть осуществлена только в случае подтверждения целостности полученных установочных модулей СКЗИ и эксплуатационной документации (см. раздел 2 ЖТЯИ.00091-04 92 01. КриптоПро JCP. Правила пользования).

При установке программного обеспечения СКЗИ необходимо:

1) На технических средствах, оснащенных СКЗИ, использовать только лицензионное программное обеспечение фирм-изготовителей.

2) Установленное программное обеспечение не должно содержать средств разработки и отладки приложений, а также средств, позволяющих осуществлять несанкционированный доступ к системным ресурсам.

3) Аппаратуру, на которой устанавливается СКЗИ, следует получать у добросовестного производителя, проверяя наличие подтверждающих работоспособность документов.

4) Рекомендуется установка средств защиты от НСД и при использовании СКЗИ в соответствии с классом КС1.

5) При установке Java получить у производителя последнюю официальную версию, содержащую все программные обновления, связанные с безопасностью.

6) Перед установкой СКЗИ проверить программное обеспечение ПЭВМ на отсутствие вирусов и программных закладок.

7) Предусмотреть меры, исключающие возможность несанкционированного изменения аппаратной части рабочей станции с установленной СКЗИ (путем опечатывания системного блока и разъемов ПЭВМ и контроля печатей администратором безопасности).

8) После установки СКЗИ, но до начала его использования, необходимо, воспользовавшись утилитой командной строки CPVerify.Prompt, создать хранилища контролируемых файлов, как описано в Приложении 1 ЖТЯИ.00091-04 92 01. КриптоПро JCP. Правила пользования.

Подробный процесс установки СКЗИ КриптоПро JCP версии 2.0 R4 описан в разделе 1 ЖТЯИ.00091-04 91 01. Инструкция по использованию.

8 Управление протоколированием

Журналирование работы КриптоПро JCP осуществляется стандартными средствами Java-машины. Формат протокола, поля вывода, уровни протоколирования настраиваются в файле <jre>/lib/logging.properties. Имя класса протокола для JCP: **ru.CryptoPro.JCP.tools.JCPLogger**.

Уровни протоколирования JCP совпадают с уровнями протоколирования Java, ниже они приведены в порядке по возрастанию информативности сообщений, уровень выше включает все сообщения, приведенные по тексту ниже. Уровень **ALL** включает все сообщения, уровень **OFF** выключает все сообщения.

При настройках Java-машины по умолчанию включен уровень **INFO**.

Уровни протоколирования JCP:

- **OFF** — В протокол не выводятся никакие сообщения.
- **SEVERE** — Критические ошибки в JCP, функционирование JCP после появления этих ошибок невозможно. К ним относятся ошибки загрузки, ошибки контроля целостности и др.
- **WARNING** — Ошибки JCP, не приводящие к отказу функционирования. К ним относятся, например, ошибки настройки JCP, неправильный вызов функций JCP.
- **INFO** — Информационные сообщения о загрузке JCP.
- **CONFIG** — Информационные сообщения при получении текущих настроек используемых JCP.
- **FINE** — Информационные сообщения о завершении функции провайдера с ошибкой.
- **FINER** — Информационные сообщения связанные с входом/выходом в/из функции провайдера.
- **FINEST** — Уровень не используется
- **ALL** — Уровень не используется, приводит к выводу всех сообщений, выдаваемых JavaCSP.

При включении уровня, отличного от заданного по умолчанию (**INFO**), следует помнить, что уровни выше **CONFIG** могут значительно замедлить скорость провайдера, а уровни ниже **INFO** привести к несвоевременному обнаружению причин отказа JCP. При обычной работе КриптоПро JCP рекомендуется оставлять настройку уровня выводимых сообщений по умолчанию (**INFO**).

Пример настройки файла `logging.properties` с уровнем **FINE**:

```
...
# Default global logging level.
# This specifies which kinds of events are logged across
# all loggers.  For any given facility this global level
# can be overriden by a facility specific level
# Note that the ConsoleHandler also has a separate level
# setting to limit messages printed to the console.
.level= INFO

############################################################
# Handler specific properties.
# Describes specific configuration info for Handlers.
############################################################

# default file output is in user's home directory.
java.util.logging.FileHandler.pattern = %h/java%u.log
java.util.logging.FileHandler.limit = 50000
java.util.logging.FileHandler.count = 1
java.util.logging.FileHandler.formatter = java.util.logging.XMLFormatter

# Limit the message that are printed on the console to INFO and above.

#java.util.logging.ConsoleHandler.level = INFO
java.util.logging.ConsoleHandler.level = FINE
```

```
java.util.logging.ConsoleHandler.formatter = java.util.logging.SimpleFormatter

############################################################
# Facility specific properties.
# Provides extra control for each logger.
############################################################

# For example, set the com.xyz.foo logger to only log SEVERE
# messages:

com.xyz.foo.level = SEVERE

ru.CryptoPro.JCP.tools.JCPLogger.level = FINE
...
```

9 Отключение функций телеметрии

Для отключения функций телеметрии на ОС Windows 10/Server 2016 необходимо выполнить следующие действия:

1) Проверить наличие и статус сервиса DiagTrack (Панель управления → Система и безопасность → Администрирование → Службы).

2) Если сервис запущен, то остановить его.

3) Удалить запись регистрации сервиса DiagTrack из реестра (раздел `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services`, удалить папку DiagTrack).

4) Удалить подготовленные к отправке данные, которые сохраняются в четырех файлах с расширением *.rbs, хранящихся в директории `%ProgramData%\Microsoft\Diagnosis`. Имена файлов для production сборок ОС — event00.rbs, event01.rbs, event10.rbs и event11.rbs. Для insider сборок ОС имена могут отличаться, поэтому необходимо удалить все файлы с расширением *.rbs. При возникновении проблем с удалением данных файлов необходимо в свойствах на вкладке «Безопасность» разрешить полный доступ к файлу, а затем удалить его.

5) Остановить автоматическую (AutoLogger) ETW сессию AutoLogger-DiagTrack-Listener, которую DiagTrack активирует в процессе своей остановки.

6) Удалить файл, в который автоматическая (AutoLogger) ETW сессия AutoLoggerDiagTrack-Listener сохраняла собранные данные.

Путь к файлу хранится в реестровой записи AutoLogger-DiagTrack-Listener в значении FileName. Конфигурации автоматических (AutoLogger) ETW сессий находятся в ключе реестра `HKLM\SYSTEM\CurrentControlSet\Control\WMI\AutoLogger`. Конфигурация целевой сессии хранится в данном ключе под записью AutoLogger-DiagTrack-Listener.

В настоящее время данные сохраняются в файл `%ProgramData%\Microsoft\Diagnosis\ETLLogs\AutoLogger\AutoLogger-DiagTrack-Listener.etl`.

7) Удалить запись регистрации конфигурации автоматической (AutoLogger) ETW сессии AutoLogger-DiagTrack-Listener из реестра.

Данные действия необходимо выполнять после каждого кумулятивного обновления, поскольку данные обновления являются по сути полной переустановкой ОС и удаленные сервисы восстанавливаются.

10 Использование класса-загрузчика новой лицензии

Для установки новой лицензии можно воспользоваться одним из способов, описанных в ЖТЯИ.00091-04 91 01. КриптоПро JCP. Инструкция по использованию.

Для установки новой лицензии с сервера лицензий необходимо выполнить следующие операции:

1) Создать класс, расширяющий абстрактный класс `JCP.tools.LicenseLoader` и реализующий метод `getNewLicense()`.
2) Записать имя класса в реестр с помощью метода `setLoaderName(String class_name)` класса `JCP.tools.License`.

Теперь по истечении лицензии КриптоПро JCP получает из реестра имя класса-установщика, который затем обращается к серверу, который в свою очередь пытается получить новую лицензию от сервера лицензий. Если всё проходит успешно (не превышено максимальное число лицензий, выдаваемых за определенный срок и т.д.), то клиент получает новую лицензию.

Также установка может быть инициирована пользователем посредством вызова метода `setNewLicense()` класса `JCP.tools.License` при условии выполнения двух вышеуказанных действий. При обработке запроса на новую лицензию от клиента серверу может потребоваться текущая клиентская лицензия. Ее можно получить, вызвав конструктор класса `JCP.tools.License` без параметров:

```
License current_license = new License();
```

Дату окончания действия лицензии можно получить, вызвав метод `getEndDate()`. Запись лицензии в реестр осуществляет метод `store()`. Метод `verifyLicense()` получает тип лицензии.

По умолчанию КриптоПро JCP использует собственный класс-загрузчик, который не обращается к серверу, а предлагает пользователю ввести лицензию в обычном диалоговом окне.

Ниже приведен пример создания класса-загрузчика и его использования.

Создание класса:

```
public class TestLicenseLoader extends ru.CryptoPro.JCP.tools.LicenseLoader {
    public AbstractLicense getNewLicense() throws Exception {
        License new_lic;
        ...
        // transporting all required data to the server
        // getting the new license
        ...
        return new_lic;
    }
    }
```

Установка лицензии "вручную"пользователем:

```
...
// some code
String path = "......";// path to the class
String class_name = path + "TestLicenseLoader";
ru.CryptoPro.JCP.tools.License.setLoaderName(class_name);
try {
License dummyLicense = new License(null,null,null);
    dummyLicense.setNewLicense();
}
catch (Exception e) {
// catching exception
...
```

```
}
// another code
    ...
```

Типы лицензии:

- PERMANENT_LICENSE - верная постоянная лицензия,
- CORRECT_TEMP_LICENSE - верная временная лицензия,
- NEED_NOTIFY - временная лицензия истекает,
- RUN_OUT_OF_TIME - лицензия истекла,
- INCORRECT_ID_FORM - форма серийного номера неверна,
- INCORRECT_PRODUCT_TYPE - неверный тип продукта,
- INCORRECT_ID_HASH - неверный серийный номер,
- INCORRECT_CPU_AMOUNT - превышено допустимое число ЦПУ,
- INCORRECT_FIRST_DATE - неверная дата первой установки,
- INCORRECT_ID_SERVER – неверный тип лицензии – не серверная лицензия,
- INCORRECT_ID_CRYPTO – неверный тип лицензии – не для шифрования,
- INCORRECT_LICENSE_VERSION – неверная версия лицензии.

10.1 Сигнатуры public-конструкторов, полей и методов класса ru.CryptoPro.JCP.tools.License

Таблица 2. Конструкторы

public License() throws IOException;	конструктор лицензии «КриптоПро JCP» версия 2.0 R3, получающий текущую лицензию из реестра. Выбрасывает исключение в случае ошибки чтения.
public License(String srcUserName, String srcCompanyName, String srcProductID);	конструктор лицензии «КриптоПро JCP» версия 2.0 R3 по имени пользователя, названию компании и серийному номеру.
public JCSPLicense() throws IOException;	конструктор лицензии «КриптоПро Java CSP» версия 5.0, получающий текущую лицензию из реестра. Выбрасывает исключение в случае ошибки чтения.
public JCSPLicense(String srcUserName, String srcCompanyName, String srcProductID);	конструктор лицензии «КриптоПро Java CSP» версия 5.0 по имени пользователя, названию компании и серийному номеру.
public ServerPLicense() throws IOException;	конструктор лицензии «КриптоПро Java TLS» версия 2.0, получающий текущую лицензию из реестра. Выбрасывает исключение в случае ошибки чтения.
public ServerPLicense(String srcUserName, String srcCompanyName, String srcProductID);	конструктор лицензии «КриптоПро Java TLS» версия 2.0 по имени пользователя, названию компании и серийному номеру.

Таблица 3. Поля

public static final String STR_VALID_LICENSE = "Valid license.";	строка "лицензия верна".
public static final long NOTIFY_TIME = 24 * 60 * 60 * 1000;	время до истечения срока действия лицензии, при котором запрашивается ее обновление.
public static final int SERIAL_PRODUCTID_NUM = 20;	размер серийного номера с разделителями.

Таблица 4. Методы

public long getAllowedAmount();	возвращает допустимое число процессоров.
public String getCompanyName();	получает имя компании.
public String getDefaultLoaderName();	получает имя класса-загрузчика, используемого по умолчанию.
public static String getDefaultUserName();	получает имя пользователя по умолчанию.
public long getEndDate();	возвращает WRONG_LICENSE, если лицензия неверна, PERMANENT_LICENSE, если лицензия неограничена, срок окончания действия в миллисекундах иначе.
public String getProductID();	получает лицензионный номер продукта.
public String getUserName();	получает имя пользователя, на которого зарегистрирована лицензия.
public String getVersion();	получает номер версии лицензии.
public boolean isWriteAvailable();	проверяет наличие необходимых прав для записи лицензии.
public void setLoaderName(String name);	устанавливает имя класса-загрузчика новой лицензии и записывает его в реестр.
public void setNewLicense() throws Exception;	применение описано выше.
public void store() throws ConfigurationException;	осуществляет запись лицензии в реестр. Выбрасывает исключение в случае ошибки записи.
public int verifyLicense();	проверяет корректность лицензии и возвращает ее тип. В случае клиентской лицензии «КриптоПро JavaCSP» возвращает PERMANENT_LICENSE.
public int verifyLicense(Object o, boolean param);	проверяет корректность лицензии и возвращает ее тип. Если param равен true, то проверка лицензии «КриптоПро JavaCSP» выполняется независимо от ее типа.