ГАЗИНФОРМ СЕРВИС

Средство защиты информации от несанкционированного доступа «Блокхост-Сеть 4»

Руководство администратора безопасности

Часть 4. Настройка подсистемы управления токенами, ЦС DogTag

Санкт-Петербург, 2023

Аннотация

В документе содержатся сведения о настройке подсистемы управления жизненным циклом токенов средства защиты информации от несанкционированного доступа «Блокхост-Сеть 4» при взаимодействии с центром сертификации DogTag и сервисом каталогов FreeIPA, а также сведения о настройке сертификатов необходимых для функционирования подсистемы.

Содержание

Введение 5
1 Подсистема жизненного цикла токенов 6
2 Настройки на сервисе каталогов FreeIPA/центре сертификации Dogtag 7
2.1 Создание сервисной учетной записи 7
2.2 Создание сертификата агента регистрации 8
2.3 Создание профиля для выпуска сертификата пользователя 10
3 Настройки на сервере «Блокхост-Сеть» 14
3.1 Установка сертификата агента регистрации 14
3.1.1 Экспорт сертификата агента регистрации 14
3.1.2 Установка сертификата агента регистрации на сервер под управлением ОС Windows 14
3.2 Установка доверенного корневого сертификата и промежуточного сертификата 17
3.2.1 Экспорт доверенного корневого сертификата и промежуточного сертификата 17
3.2.2 Установка доверенного корневого сертификата и промежуточного сертификата на сервер под управлением ОС Windows 18
3.2.3 Установка доверенного корневого сертификата и промежуточного сертификата на сервер под управлением ОС Linux 19
3.3 Создание профилей в настройках подсистемы 20
3.3.1 Создание профиля FreeIPA/DogTag 21
4 Особенности при выпуске токена средствами подсистемы 26
5 Настройка входа по сертификатам ЦС DogTag в ОС Linux 27
5.1 Приоритет проверок средств аутентификации при входе 27
5.2 Настройка входа по сертификатам в ОС Альт 28
5.2.1 Ввод рабочей станции под управлением ОС Альт в домен FreeIPA 28
5.2.2 Общий порядок настройки для входа по сертификатам в ОС Альт 29
5.2.3 Экспорт корневого сертификата FreeIPA 29
5.2.4 Настройка Policy Kit в ОС Альт 31
5.2.5 Установка пакетов для работы с токенами 31
5.2.6 Проверка работы PC/SC интерфейса 32
5.2.7 Проверка работы библиотек вендоров PKCS#11 32

5.2.8 Настройка и проверка работы службы Kerberos 33
5.2.9 Настройка и проверка службы SSSD 34
5.3 Настройка входа по сертификатам в Astra Linux 37
5.3.1 Ввод рабочей станции под управлением ОС Astra Linux в домен FreeIPA 37
5.3.2 Общий порядок настройки для входа по сертификатам в ОС Astra Linux 38
5.3.3 Экспорт корневого сертификата FreeIPA 39
5.3.4 Установка пакетов для работы с токенами 40
5.3.5 Настройка и проверка работы службы Kerberos 40
5.4 Настройка входа по сертификатам в РЕД ОС 42
5.4.1 Ввод рабочей станции под управлением РЕД ОС в домен FreeIPA 42
5.4.2 Общий порядок настройки для входа по сертификатам в РЕД ОС 44
5.4.3 Экспорт корневого сертификата FreeIPA 44
5.4.4 Настройка Policy Kit в РЕД ОС 45
5.4.5 Установка пакетов для работы с токенами 46
5.4.6 Проверка работы PC/SC интерфейса 46
5.4.7 Проверка работы библиотек вендоров PKCS#11 47
5.4.8 Настройка и проверка работы службы Kerberos 47
5.4.9 Настройка и проверка службы SSSD 49
Перечень сокращений 53

Введение

Подсистема управления жизненным циклом токенов предназначена для управления носителями пользователей на протяжении всего жизненного цикла.

Для настройки и установки подсистемы управления жизненным циклом токенов в существующей иерархической структуре уже должны быть развернуты следующие компоненты:

– корневой центр сертификации;

– служба каталогов FreeIPA совместно с центром сертификации DogTag.

Знаки, расположенные на полях руководства, указывают на примечания.

Степени важности примечаний:

Важная информация
Указания, требующие особого внимания.

Дополнительная информация
Указания, позволяющие упростить работу с подсистемой.

1 Подсистема жизненного цикла токенов

Подсистема предназначена для управления устройствами аутентификации пользователей на протяжении всего жизненного цикла и обеспечивает учет токенов в системе, управление токенами и аудит в течении всего жизненного цикла.

Подсистема жизненного цикла токенов поддерживает управление токенами перечисленными в таблице 1.1.

Таблица 1.1 – Поддерживаемые подсистемой токены

Носитель	**Производитель**	**Наименование носителя**	
		поддерживаемые ОС Windows	**поддерживаемые ОС Linux**
eToken	ЗАО «АЛАДДИН Р.Д.»	eToken Pro, eToken Pro (Java), eToken ГОСТ, eToken NG-FLASH, eToken NG-FLASH (Java), eToken NG-OTP, eToken NG-OTP (Java), eToken GT (Java), eToken PRO (Java) SC, eToken PRO SC	
SafeNet eToken	ООО «Сертифицированные информационные системы»	SafeNet eToken 5100, SafeNet eToken 5105, SafeNet eToken 5200, SafeNet eToken 5205, SafeNet eToken 7200, SafeNet eToken 7300, SafeNet eToken 4100	
ruToken	ЗАО «Актив Софт»	ruToken, ruToken S, ruToken Lite, ruToken ЭЦП	ruToken ЭЦП
JaCarta	ЗАО «АЛАДДИН Р.Д.»	JaCarta-2, JaCarta PKI, JaCarta PRO	JaCarta-2, JaCarta PKI, JaCarta PRO
eSmart Token	ООО «Интеллектуальные системы управления бизнесом»	eSmart Token	eSmart Token

Для работы с сертификатами на токене в подсистеме жизненного цикла СЗИ от НСД «Блокхост-Сеть 4» необходимо выполнить определенные настройки центра сертификации и в самой подсистеме жизненного цикла токенов.

2 Настройки на сервисе каталогов FreeIPA/центре сертификации Dogtag

2.1 Создание сервисной учетной записи

Для первоначальной настройки подсистемы жизненного цикла токенов необходимо создать сервисную учетную запись (агента регистрации), обладающую достаточными правами для создания сертификата и входящую в группу издателей сертификатов сервиса каталогов FreeIPA.

Для создания сервисной учетной записи (например, ***serviceCA***) необходимо выполнить следующие действия:

1) Создать базу данных сертификатов:

```
mkdir /root/nss
certutil -N -d /root/nss
```

После выполнения команды ввести пароль доступа к базе сертификатов и повторить его.

2) Импортировать сертификат CA Агента для доступа к службе выдающего ЦС:

```
pk12util -i /root/ca-agent.p12 -d /root/nss
```

3) Для проверки доступа выполнить команду для вывода списка сертификатов в созданной базе:

```
certutil -L -d /root/nss
```

В результате выполнения команды отобразится список, содержащий сертификат CA Агента.

4) Провести проверку подключения к службе выдающего ЦС с запросом списка профилей сертификатов:

```
sudo pki -d /root/nss/ -n ipa-ca-agent ca-profile-find
```

5) Создать новую сервисную учетную запись от имени пользователя, обладающего правами администратора с помощью команды:

```
sudo pki –d /root/nss/ -n ipa-ca-agent ca-user-add <имя учетной записи> --fullName "псевдоним учетной записи"
```

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -n ipa-ca-agent ca-user-add serviceCA --fullName “CA Агент”
[sudo] password for user:
Enter password for Internal Key Storage Token
----------------------
Added user "serviceCA"
----------------------
  User ID: serviceCA
  Full name: CA Агент
user@alt9cafreeipa ~ $
```

Рисунок 2.1 – Создание сервисной учетной записи

6) Добавить созданную сервисную учетную запись в группу издателей сертификатов Certificate Manager Agents с помощью команды:

```
sudo pki -d /root/nss/ -n ipa-ca-agent ca-group-member-add "Certificate Manager Agents" <имя учетной записи>
```

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -n ipa-ca-agent ca-group-member-add "Certificate Manager Agents" serviceCA
[sudo] password for user:
Enter password for Internal Key Storage Token

----------------------------
Added group member "serviceCA"
----------------------------
  User: serviceCA
user@alt9cafreeipa ~ $
```

Рисунок 2.2 – Добавление сервисной учетной записи в группу издателей сертификатов

2.2 Создание сертификата агента регистрации

Для выпуска пользовательских сертификатов на токене необходим сертификат агента регистрации. В этом случае администратор может обращаться к центру сертификации DogTag для выпуска сертификатов пользователям сервиса каталогов FreeIPA.

Для создания сертификата сервисной учетной записи необходимо выполнить следующие действия:

1) Создать запрос на сертификат агента регистрации от имени пользователя, обладающего правами администратора с помощью команды:

```
sudo pki -d /root/nss/ -c <пароль администратора ЦС> client-cert-request uid=<имя УЗ CA Агента> --profile CustomUserCert
```

или

```
sudo pki -d /root/nss/ -c <пароль администратора ЦС> client-cert-request cn=<имя УЗ CA Агента>,E=<имя УЗ CA Агента>@<CANAME UPPER CASE> --profile CustomUserCert
```

2) Используя данные запроса сертификата *«Request ID»* выполнить создание сертификата агента регистрации с помощью команды:

```
sudo pki -d /root/nss/ -n ipa-ca-agent ca-cert-request-review <request ID> --action approve
```

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -n ipa-ca-agent ca-cert-request-review 162 --action approve
[sudo] password for user:
Enter password for Internal Key Storage Token

WARNING: The --action option has been deprecated. Use the following command instead:
WARNING:   $ pki ca-cert-request-<action> <request ID>
-------------------------------
Approved certificate request 162
-------------------------------
  Request ID: 162
  Type: enrollment
  Request Status: complete
  Operation Result: success
  Certificate ID: 0x51
user@alt9cafreeipa ~ $
```

Рисунок 2.3 – Создание сертификата агента регистрации

3) Используя данные выпуска сертификата *«Certificate ID»* выполнить назначение сертификата сервисной учетной записи с помощью команды:

```
sudo pki -d /root/nss/ -n ipa-ca-agent ca-user-cert-add <имя учетной записи> --serial <certificate ID>
```

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -n ipa-ca-agent ca-user-cert-add serviceCA --serial 0x51
[sudo] password for user:
Enter password for Internal Key Storage Token

-----------------------------------------------------------------------------------------------
Added certificate "2;81;CN=freeipa.local,O=FREEIPA.LOCAL;E=serviceCA@FREEIPA.LOCAL,CN=serviceCA"
-----------------------------------------------------------------------------------------------
  Cert ID: 2;81;CN=freeipa.local,O=FREEIPA.LOCAL;E=serviceCA@FREEIPA.LOCAL,CN=serviceCA
  Version: 2
  Serial Number: 0x51
  Issuer: CN=freeipa.local,O=FREEIPA.LOCAL
  Subject: E=serviceCA@FREEIPA.LOCAL,CN=serviceCA
user@alt9cafreeipa ~ $
```

Рисунок 2.4 – Назначение сертификата учетной записи агента регистрации

4) Выполнить импорт сертификата сервисной учетной записи в базу данных ЦС с помощью команды:

```
sudo pki -d /root/nss/ -c <пароль администратора ЦС> client-cert-import <имя учетной записи> --serial <certificate ID>
```

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -c Gazprom09 client-cert-import serviceCA --serial 0x51
[sudo] password for user:
Imported certificate "serviceCA"
WARNING: Notice: Trust flag u is set automatically if the private key is present.
user@alt9cafreeipa ~ $
```

Рисунок 2.5 – Импорт сертификата учетной записи агента регистрации

5) Выполнить экспорт сертификата и закрытого ключа сервисной учетной записи в файл с помощью команды:

```
sudo pki -d /root/nss/ -c <пароль администратора ЦС> client-cert-show <имя учетной записи> --pkcs12 <имя файла ключей агента регистрации>.p12 --pkcs12-password <пароль к файлу ключей агента регистрации>
```

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -c Gazprom09 client-cert-show serviceCA --pkcs12 serviceCA.p12 --pkcs12-password Gazprom09
[sudo] password for user:
user@alt9cafreeipa ~ $ ls
 freeipa   serviceCA.p12   Документы   Загрузки   Общедоступные   'Рабочий стол'
user@alt9cafreeipa ~ $
```

Рисунок 2.6 – Экспорт сертификата учетной записи агента регистрации

2.3 Создание профиля для выпуска сертификата пользователя

Для выпуска сертификатов на токене пользователей сервиса каталогов FreeIPA необходимо создать профиль пользователя.

Для создания профиля для выпуска сертификата пользователя необходимо выполнить следующие действия:

1) Вывести список всех доступных пользователей с помощью команды

```
sudo pki -d /root/nss/ -n ipa-ca-agent ca-profile-find
```

и найти профиль с идентификатором *CustomUserCert*.

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -n ipa-ca-agent ca-profile-find
[sudo] password for user:
Enter password for Internal Key Storage Token

-----------------
93 entries matched
-----------------
  Profile ID: acmeServerCert
  Name: ACME Server Certificate Enrollment
  Description: This certificate profile is for enrolling server certificates via ACME protocol.

  Profile ID: caCMCserverCert
  Name: Server Certificate Enrollment using CMC
  Description: This certificate profile is for enrolling server certificates using CMC.

  Profile ID: caCMCECserverCert
  Name: Server Certificate wth ECC keys Enrollment using CMC
  Description: This certificate profile is for enrolling server certificates with ECC keys using CMC.

  Profile ID: CustomUserCert
  Name: Custom Manual User Dual-Use Certificate Enrollment
  Description: This certificate profile is for enrolling user certificates.
```

Рисунок 2.7 – Список всех доступных профилей

2) Выгрузить найденный профиль в файл с помощью команды:

```
sudo pki -d /root/nss/ -n ipa-ca-agent ca-profile-show <идентификатор профиля> --raw --output <имя файла>.cfg
```

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -n ipa-ca-agent ca-profile-show CustomUserCert --raw --output CustomUserCert.cfg
[sudo] password for user:
Enter password for Internal Key Storage Token

-----------------------------------------------
Saved profile CustomUserCert to CustomUserCert.cfg
-----------------------------------------------
user@alt9cafreeipa ~ $
```

Рисунок 2.8 – Сохранение профиля в файл

3) Просмотреть выгруженный профиль можно по команде:

```
sudo pluma <имя файла>.cfg
```

CustomUserCert.cfg

```
1 auth.class_id=
2 classId=caEnrollImpl
3 desc=This certificate profile is for enrolling user certificates.
4 enable=true
5 enableBy=admin
6 input.i1.class_id=keyGenInputImpl
7 input.i2.class_id=subjectNameInputImpl
8 input.i3.class_id=submitterInfoInputImpl
9 input.list=i1,i2
10 name=Custom Manual User Dual-Use Certificate Enrollment
11 output.list=o1
12 output.o1.class_id=certOutputImpl
13 policyset.list=userCertSet
14 policyset.userCertSet.1.constraint.class_id=subjectNameConstraintImpl
15 policyset.userCertSet.1.constraint.name=Subject Name Constraint
16 policyset.userCertSet.1.constraint.params.accept=true
17 policyset.userCertSet.1.constraint.params.pattern=^E=[^,]+,CN=.*$
18 policyset.userCertSet.1.default.class_id=userSubjectNameDefaultImpl
19 policyset.userCertSet.1.default.name=Subject Name Default
20 policyset.userCertSet.1.default.params.name=
21 policyset.userCertSet.10.constraint.class_id=renewGracePeriodConstraintImpl
22 policyset.userCertSet.10.constraint.name=Renewal Grace Period Constraint
23 policyset.userCertSet.10.constraint.params.renewal.graceAfter=30
```

Рисунок 2.9 – Просмотр профиля в редакторе

4) При необходимости отредактировать выгруженный профиль:

– если необходимо включить в профиль информацию о точках распространения CRL добавьте следующую секцию:

При изменении обратите внимание на указание правильного адреса сервера и пути, указанного в настройках сервера.

```
policyset.userCertSet.11.constraint.class_id=noConstraintImpl
policyset.userCertSet.11.constraint.name=No Constraint
policyset.userCertSet.11.default.class_id=crlDistributionPointsExtDefaultImpl
policyset.userCertSet.11.default.name=CRL Distribution Points Extension Default
policyset.userCertSet.11.default.params.crlDistPointsCritical=false
policyset.userCertSet.11.default.params.crlDistPointsEnable_0=true
policyset.userCertSet.11.default.params.crlDistPointsIssuerName_0=
policyset.userCertSet.11.default.params.crlDistPointsIssuerType_0=
policyset.userCertSet.11.default.params.crlDistPointsPointName_0=http:// <FQDN-имя сервера
выдающего ЦС>/ipa/crl/MasterCRL.bin
```

```
policyset.userCertSet.11.default.params.crlDistPointsPointType_0=URIName
policyset.userCertSet.11.default.params.crlDistPointsReasons_0=
```

– если необходимо изменить в профиле информацию о сроке действия сертификата отредактируйте следующую секцию:

```
policyset.userCertSet.2.constraint.params.range=1100
policyset.userCertSet.2.default.params.range=1000
```

5) После внесения изменений, переименовать профиль в параметре *profileId* (рисунок 2.11).

```
 1 auth.class_id=
 2 classId=caEnrollImpl
 3 desc=This certificate profile is for enrolling user certificates for 1 year.
 4 enable=true
 5 enableBy=admin
 6 input.i1.class_id=keyGenInputImpl
 7 input.i2.class_id=subjectNameInputImpl
 8 input.i3.class_id=submitterInfoInputImpl
 9 input.list=i1,i2
10 name=Custom Manual User Dual-Use Certificate Enrollment 1 Year
11 output.list=o1
12 output.o1.class_id=certOutputImpl
13 policyset.list=userCertSet
14 policyset.userCertSet.1.constraint.class_id=subjectNameConstraintImpl
15 policyset.userCertSet.1.constraint.name=Subject Name Constraint
103 policyset.userCertSet.8.default.params.subjAltExtPattern_0=(UTF8String)1.3.6.1.4.1.311.20.2.
104 policyset.userCertSet.8.default.params.subjAltExtPattern_1=$request.sn_e$
105 policyset.userCertSet.8.default.params.subjAltExtType_0=RFC822Name
106 policyset.userCertSet.8.default.params.subjAltExtType_1=OtherName
107 policyset.userCertSet.8.default.params.subjAltNameExtCritical=false
108 policyset.userCertSet.8.default.params.subjAltNameNumGNs=2
109 policyset.userCertSet.9.constraint.class_id=signingAlgConstraintImpl
110 policyset.userCertSet.9.constraint.name=No Constraint
111 policyset.userCertSet.-
    9.constraint.params.signingAlgsAllowed=SHA1withRSA,SHA256withRSA,SHA512withRSA,SHA1withEC,SH
    PSS,SHA384withRSA/PSS,SHA512withRSA/PSS
112 policyset.userCertSet.9.default.class_id=signingAlgDefaultImpl
113 policyset.userCertSet.9.default.name=Signing Alg
114 policyset.userCertSet.9.default.params.signingAlg=-
115 policyset.userCertSet.list=1,10,2,3,4,5,6,7,8,9
116 profileId=CustomUserCert1Year
```

Рисунок 2.10 – Изменение идентификатора профиля в редакторе

6) Выполнить загрузку измененного профиля в базу данных с помощью команды:

```
sudo pki -d /root/nss/ -n ipa-ca-agent ca-profile-add <имя профиля>.cfg --raw
```

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -n ipa-ca-agent ca-profile-add CustomUserCertYear1.cfg --raw
Enter password for Internal Key Storage Token

#Thu Mar 17 10:58:38 MSK 2022
policyset.userCertSet.7.constraint.class_id=noConstraintImpl
policyset.userCertSet.6.constraint.params.keyUsageNonRepudiation=true
policyset.userCertSet.8.default.params.subjAltExtType_0=RFC822Name
policyset.userCertSet.5.default.params.authInfoAccessCritical=false
policyset.userCertSet.6.default.params.keyUsageNonRepudiation=true
policyset.userCertSet.9.default.name=Signing Alg
policyset.userCertSet.6.constraint.name=Key Usage Extension Constraint
policyset.userCertSet.1.default.params.name=
policyset.userCertSet.6.default.params.keyUsageDataEncipherment=false
--------------------------------
Added profile CustomUserCertYear1
--------------------------------
user@alt9cafreeipa ~ $
```

Рисунок 2.11 – Загрузка измененного профиля в базу данных

7) Выполнить активацию измененного профиля с помощью команды:

```
sudo pki -d /root/nss/ -n ipa-ca-agent ca-profile-enable <идентификатор профиля>
```

```
user@alt9cafreeipa ~ $ sudo pki -d /root/nss/ -n ipa-ca-agent ca-profile-enable CustomUserCertYear1
[sudo] password for user:
Enter password for Internal Key Storage Token

-----------------------------------
Enabled profile "CustomUserCertYear1"
-----------------------------------
user@alt9cafreeipa ~ $
```

Рисунок 2.12 – Активация измененного профиля

3 Настройки на сервере «Блокхост-Сеть»

3.1 Установка сертификата агента регистрации

Установка сертификата агента регистрации осуществляется на рабочее место администратора безопасности с установленной серверной версией средства защиты от несанкционированного доступа «Блокхост-Сеть 4» для возможности выпуска сертификатов пользователей.

Для сервера «Блокхост-Сеть» под управлением ОС Linux не требуется экспорт и установка сертификата агента регистрации на сервере «Блокхост-Сеть».

3.1.1 Экспорт сертификата агента регистрации

Для экспорта сертификата агента регистрации выполните следующие действия:

1) На рабочей станции с установленным центром сертификации DogTag найти сертификат агента регистрации с помощью команды:

```
sudo ls
```

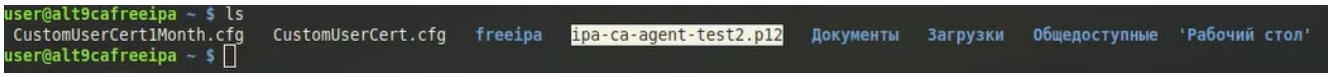


Рисунок 3.1 – Поиск сертификатов

2) Скопировать данный сертификат в каталог пользователя с помощью команды:

```
sudo cp <имя сертификата> ~
```

3) Записать сертификат на переносное устройство для установки на рабочее место администратора безопасности с установленной серверной версией средства защиты от несанкционированного доступа «Блокхост-Сеть 4».

3.1.2 Установка сертификата агента регистрации на сервер под управлением ОС Windows

Для установки сертификата агента регистрации выполните следующие действия:

1) С помощью мастера импорта сертификатов установить сертификат в личное хранилище текущего пользователя (рисунок 3.2).

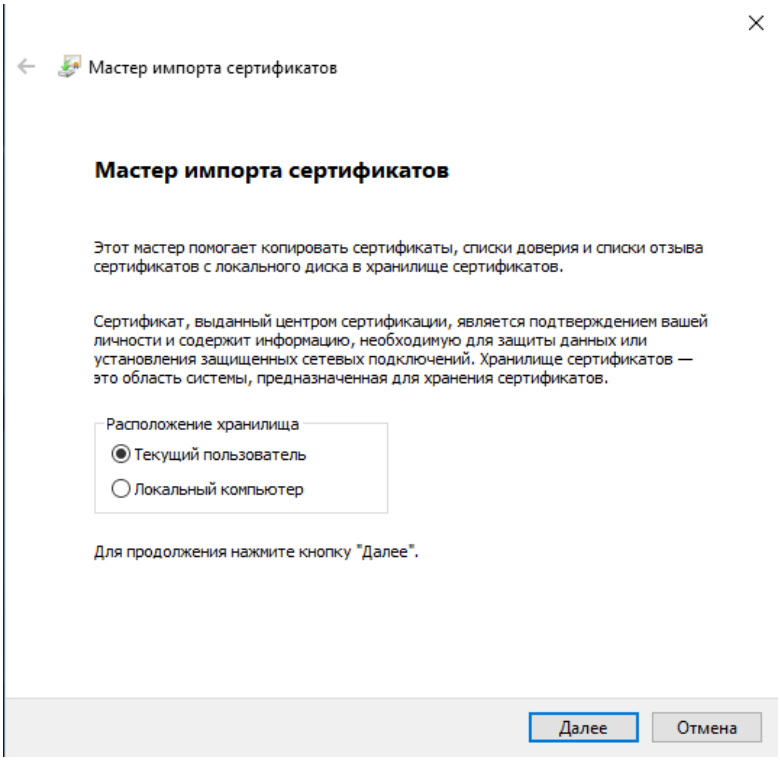


Рисунок 3.2 – Установка сертификата в хранилище пользователя

2) Указать пароль, заданный при экспорте сертификата (рисунок 3.3).

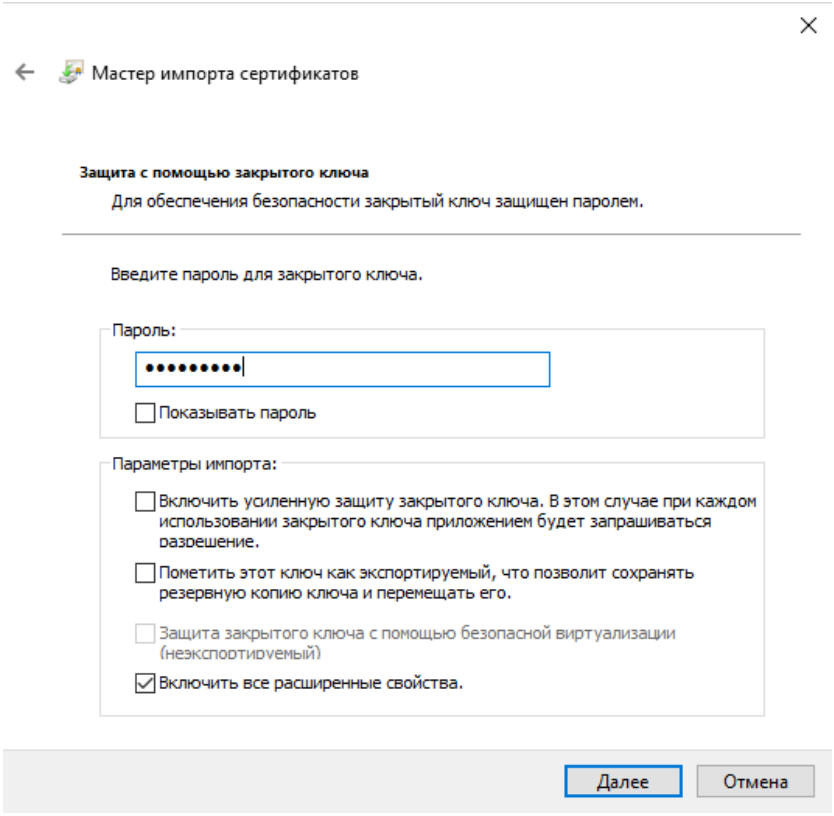


Рисунок 3.3 – Ввод пароля при импорте сертификата в хранилище

3) Выбрать хранилище сертификатов «Личные» (рисунок 3.4).

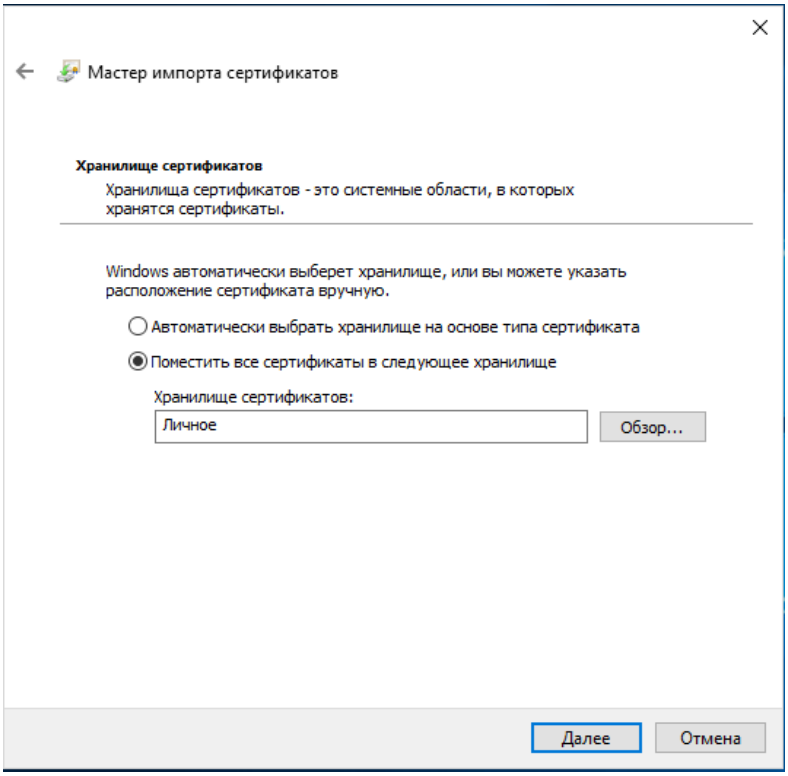


Рисунок 3.4 – Выбор хранилища при импорте сертификата

4) Перейти в директорию *C:\\Windows\System32\drivers\etc* и в файле *hosts* указать доменное имя центра сертификации:

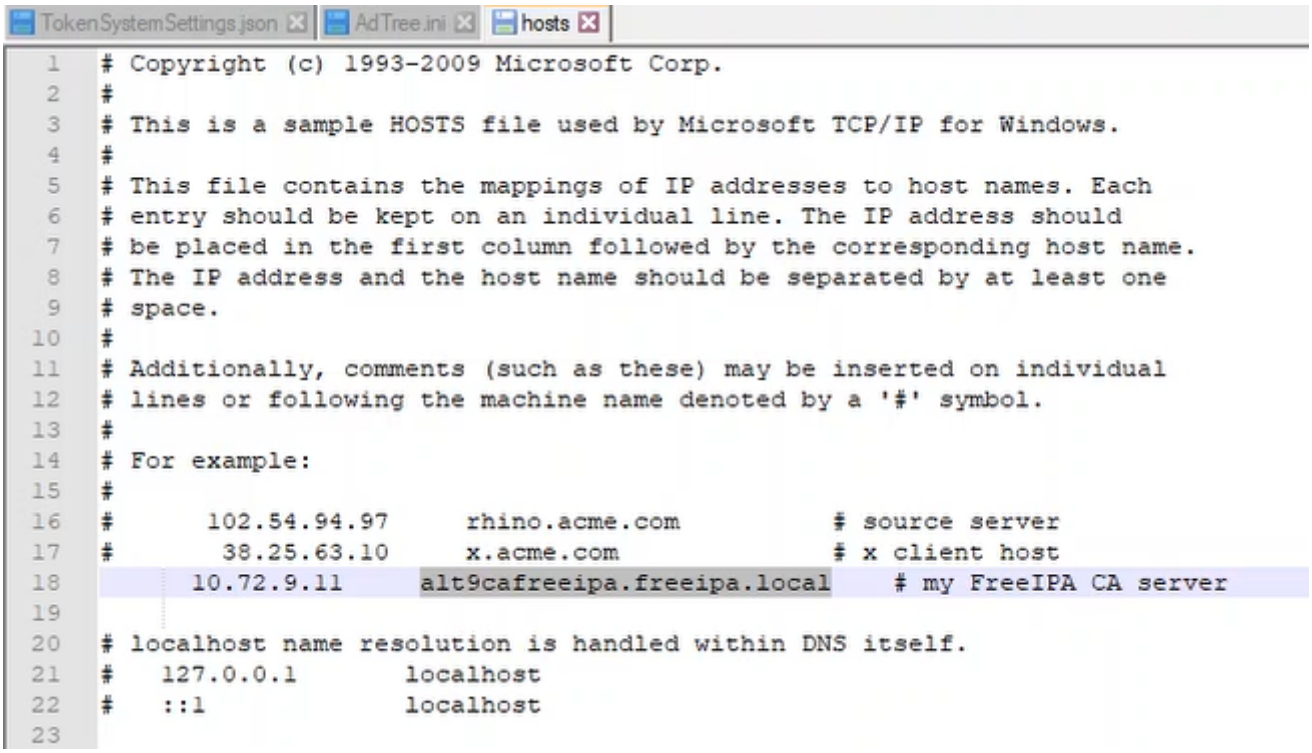


Рисунок 3.5 – Добавление центра сертификации в файл *hosts*

5) Проверить доступность центра сертификации:

```
ping alt9cafreeipa.freeipa.local
```

3.2 Установка доверенного корневого сертификата и промежуточного сертификата

Для корректного построения цепочки сертификации для пользовательского сертификата, выпущенного центром сертификации DogTag необходимо на рабочее место администратора безопасности с установленной серверной версией средства защиты от несанкционированного доступа «Блокхост-Сеть 4» выполнить установку промежуточного и доверенного корневого сертификатов.

3.2.1 Экспорт доверенного корневого сертификата и промежуточного сертификата

Для экспорта промежуточного и доверенного корневого сертификатов выполните следующие действия:

1) На рабочей станции с установленным центром сертификации DogTag найти сертификаты *ipa.cer* и *root.cer* (рисунок 3.6) с помощью команды:

```
sudo ls /root/
```

```
user@alt9cafreeipa ~ $ sudo ls /root/
[sudo] password for user:
ca-agent.p12  cacert.p12  ipa.cer  ipa.csr  kracert.p12  nss  root.cer  root.crl  tmp
```

Рисунок 3.6 – Поиск сертификатов

2) Скопировать данные сертификаты в каталог пользователя с помощью команды:

```
sudo cp /root/<имя сертификата>.cer ~
```

```
user@alt9cafreeipa ~ $ sudo cp /root/ipa.cer ~
user@alt9cafreeipa ~ $ sudo cp /root/root.cer ~
```

Рисунок 3.7 – Копирование сертификатов в каталог пользователя

3) При необходимости сменить владельца сертификатов с помощью команды

```
sudo chown user:user <имя сертификата>.cer
```

```
user@alt9cafreeipa ~ $ sudo chown user:user ipa.cer
user@alt9cafreeipa ~ $ sudo chown user:user root.cer
```

Рисунок 3.8 – Смена владельца сертификатов

4) Записать на переносное устройство для установки промежуточного и доверенного корневого сертификата на рабочее место администратора безопасности с установленной серверной версией средства защиты от несанкционированного доступа «Блокхост-Сеть 4».

3.2.2 Установка доверенного корневого сертификата и промежуточного сертификата на сервер под управлением ОС Windows

Для установки доверенного корневого сертификата и промежуточного сертификата выполните следующие действия:

1) С помощью мастера импорта сертификатов установить сертификаты в хранилище локального компьютера (рисунок 3.9).

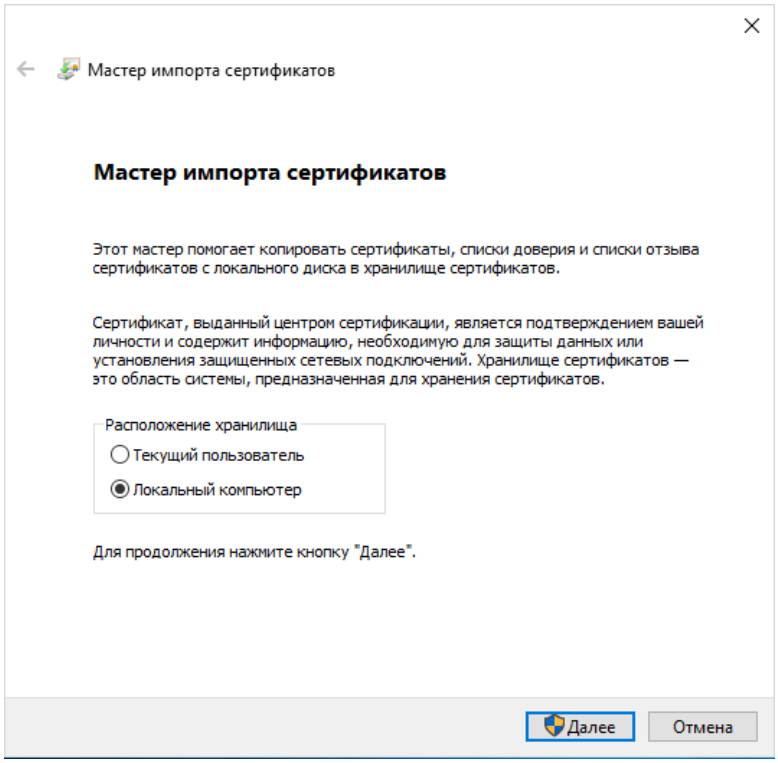


Рисунок 3.9 – Установка сертификата в хранилище пользователя

2) Выбрать хранилище сертификатов «Доверенные корневые центры сертификации» для корневого сертификата *root.cer* и «Промежуточные центры сертификации» для сертификата *ipa.cer* (рисунок 3.10).

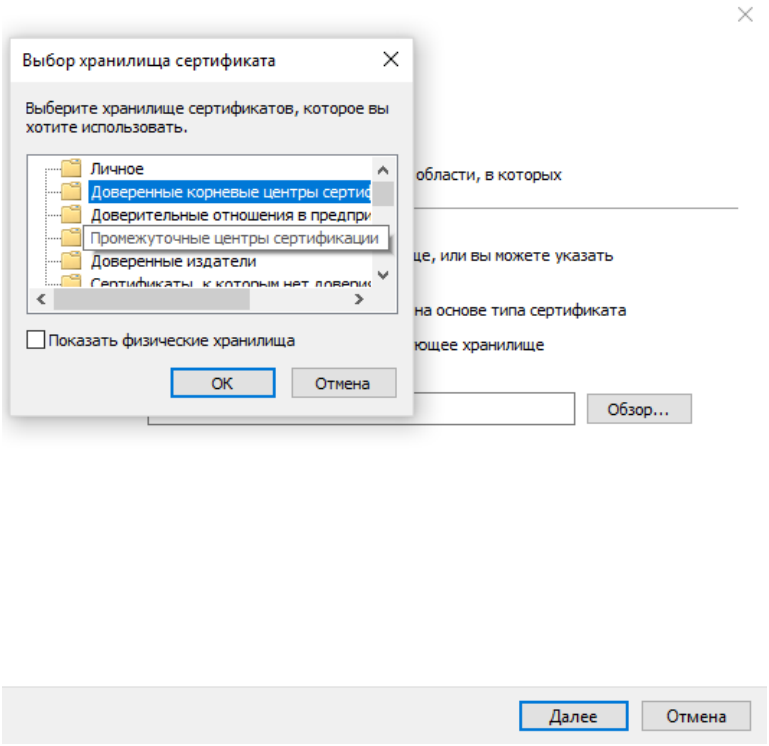


Рисунок 3.10 – Выбор хранилища при импорте сертификата

3.2.3 Установка доверенного корневого сертификата и промежуточного сертификата на сервер под управлением ОС Linux

Для установки доверенного корневого сертификата и промежуточного сертификата выполните следующие действия:

1) Установить «КриптоПро CSP» с помощью *deb* или *rpm* пакета в соответствии с версией ОС Linux (скачать «КриптоПро CSP» можно по ссылке https://www.cryptopro.ru/products/csp).

2) Перейти в директорию установки архива «КриптоПро CSP» и выполнить распаковку архива с помощью команды:

```
tar -xvf <название архива>
```

3) Перейти в директорию распаковки командой:

```
cd <название папки>
```

4) Выполнить запуск установочного скрипта командой:

```
sudo ./install.sh или sudo bash install.sh

sudo ./install_gui.sh или sudo bash install_gui.sh
```

5) Выполнить установку доверенного корневого сертификата и промежуточного

сертификата:

```
/opt/cprocsp/bin/amd64/certmgr -inst -store root -file /home/user/root.cer
/opt/cprocsp/bin/amd64/certmgr -inst -store root -file /home/user/ipa.cer
```

3.3 Создание профилей в настройках подсистемы

Для корректной работы подсистемы управления жизненным циклом токенов по выпуску сертификатов необходимо создать профиль для выпуска сертификата в настройках подсистемы управления жизненным циклом токенов.

Откройте консоль администрирования Блокхост-Сеть и перейдите в раздел ***Настройки*** → ***Выпуск сертификатов*** по кнопке (рисунок 3.11).

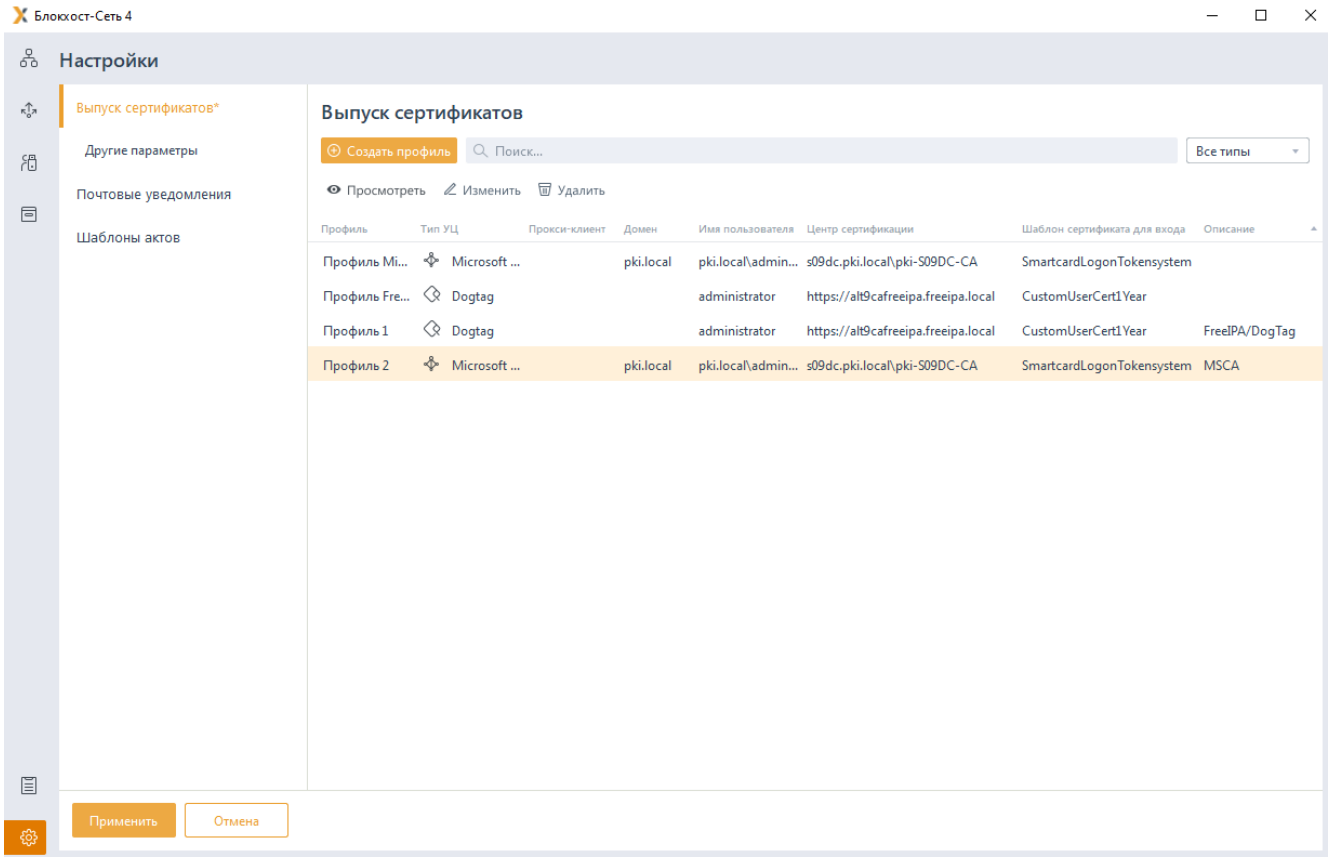


Рисунок 3.11 – Раздел ***Настройки***

Для каждого созданного профиля в окне ***Выпуск сертификатов*** отображается информация:

– *Профиль* – наименование профиля;

– *Тип УЦ* – центр сертификации выпускающий сертификат (Microsoft CA/DogTag);

– *Прокси*-клиент – клиентская рабочая станция, выбранная для взаимодействия с центром сертификации;

– *Домен* – наименование домена;

– *Имя пользователя* – наименование сервисной учетной записи;
– *Центр сертификации* – адрес центра сертификации;
– *Шаблон сертификата* – шаблон сертификата для входа;
– *Описание* – подробное описание профиля.

При необходимости, любой из созданных профилей возможно изменить или удалить с помощью соответствующих кнопок в верхней части вкладки (рисунок 3.11).

3.3.1 Создание профиля FreeIPA/DogTag

Для создания профиля по выпуску сертификатов центра сертификации DogTag нажмите на кнопку Создать профиль и выберите ***FreeIPA/DogTag*** (рисунок 3.12).

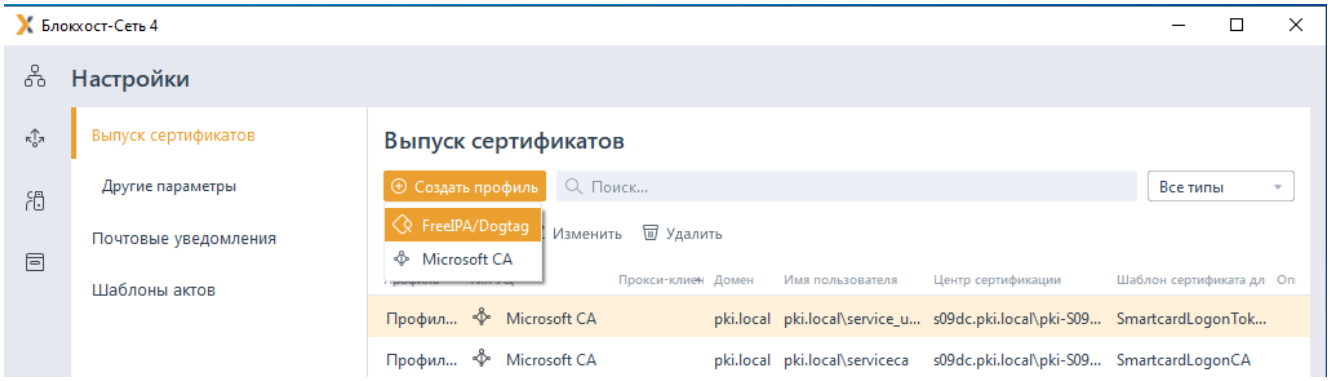


Рисунок 3.12 – Выбор центра сертификации

В появившемся окне (рисунок 3.13) введите наименование профиля, его описание, при необходимости, и параметры сервисной учетной записи (имя пользователя и пароль) в соответствующих полях.

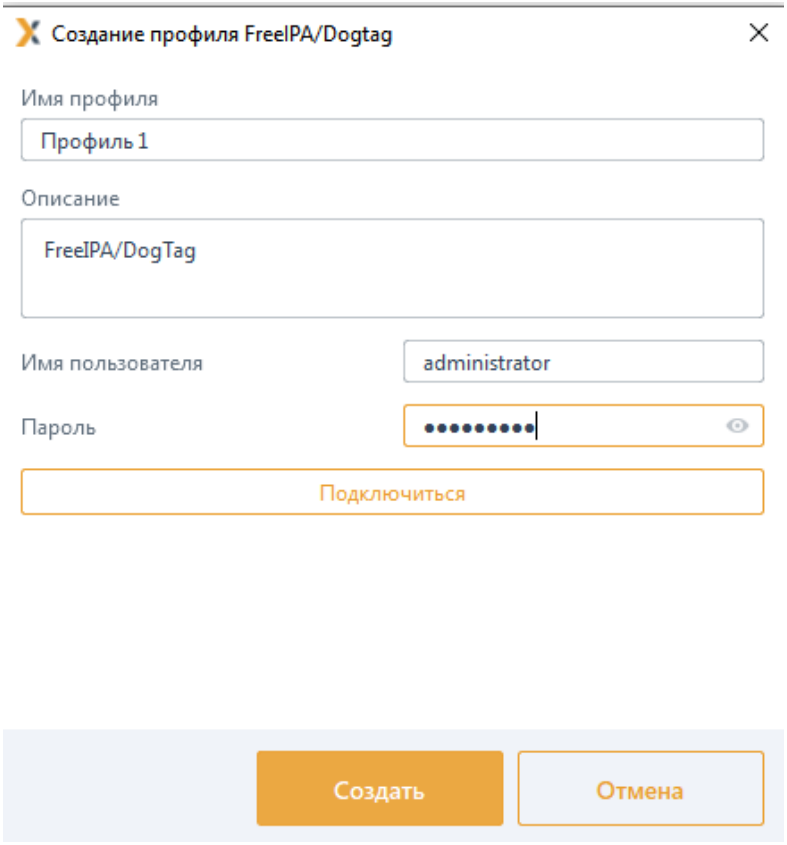


Рисунок 3.13 – Создание профиля FreeIPA/DogTag

Нажмите кнопку ***Подключиться*** (рисунок 3.13).

При возникновении ошибки при подключении убедитесь, что для вводимой сервисной учетной записи или для группы, в которую учетная запись входит, правилами клиентской политики безопасности не установлен запрет на аутентификацию ОС (подробное описание приведено в документе «СЗИ от НСД «Блокхост-Сеть 4». Руководство администратора безопасности. Часть 1. Управление политиками»).

После успешного подключения добавьте адрес центра сертификации в поле ***Центр сертификации*** и по кнопке ***Выбрать сертификат Enrollment agent*** укажите путь к сертификату агента регистрации, дающий право выпускать пользовательские сертификаты, установленный в личное хранилище оснастки **Сертификаты.** (рисунок 3.14).

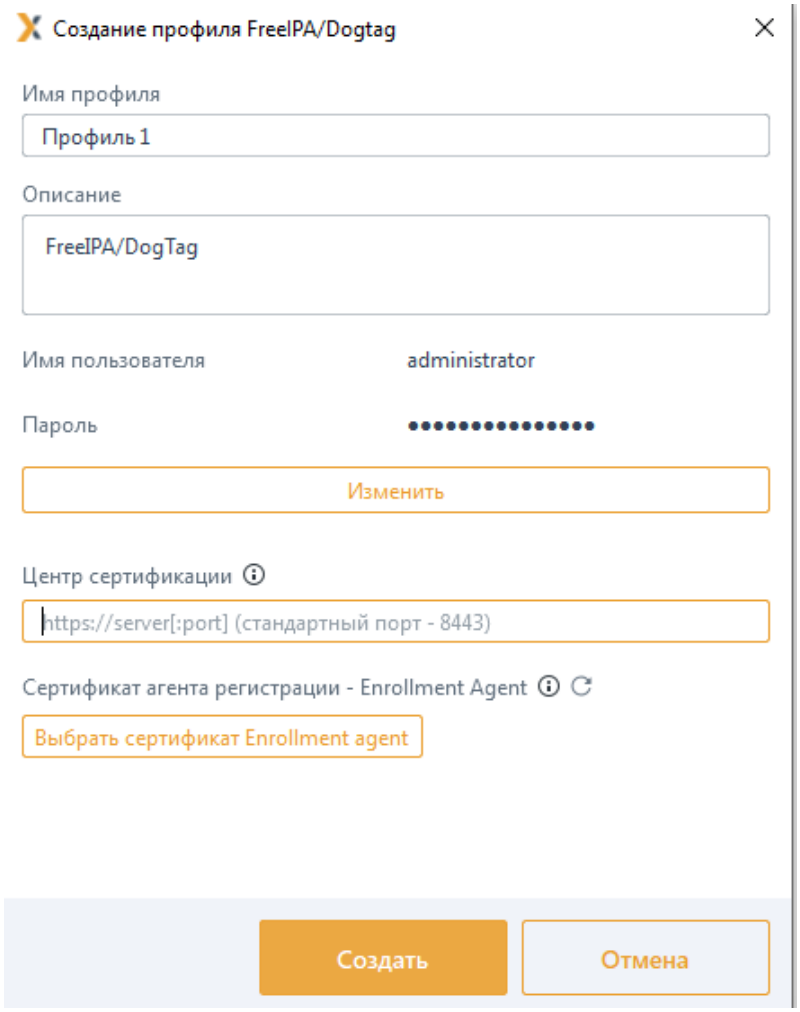


Рисунок 3.14 – Ввод центра сертификации и выбор сертификата агента регистрации

В поле ***Шаблон сертификата для входа*** (рисунок 3.15) выберите из списка шаблон сертификата пользователя, созданный в центре регистрации (например, ***CustomUserCert1Year***, созданный в разделе ***Создание профиля для выпуска сертификата пользователя***) и нажмите ***Применить***.

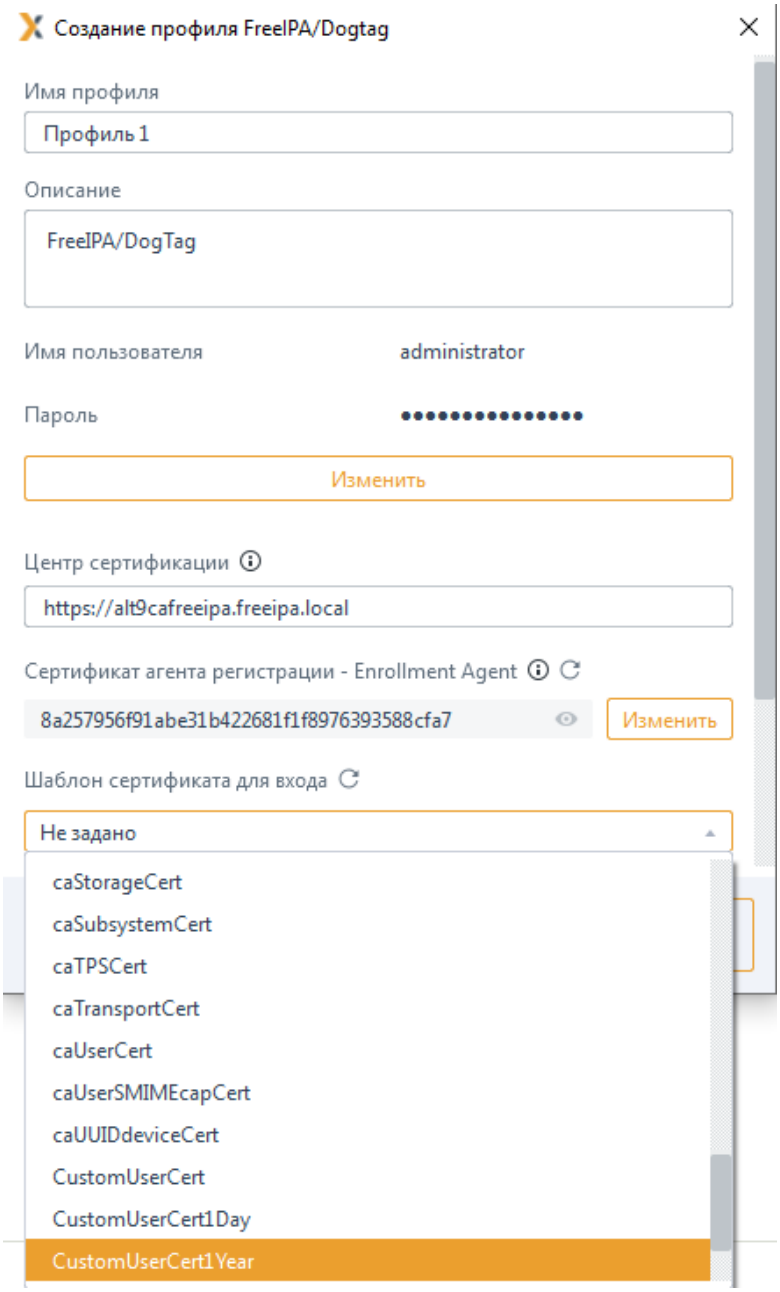


Рисунок 3.15 – Выбор шаблона сертификата для входа

После нажатия кнопки ***Создать*** профиль отобразится в списке и будет доступен к выбору при выпуске токена пользователю для входа по сертификату (рисунок 3.16).

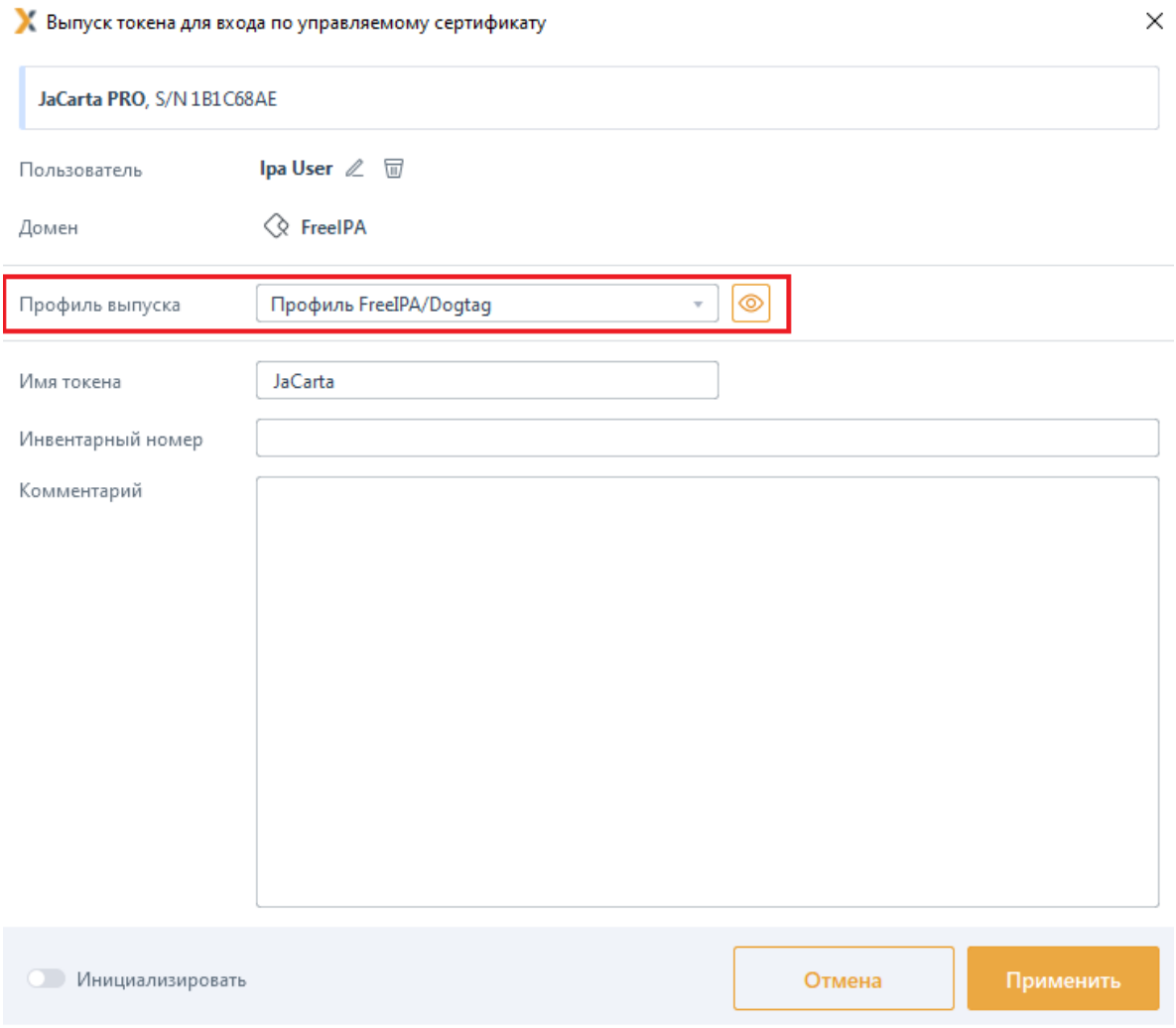


Рисунок 3.16 – Отображение профиля при выпуске токена

4 Особенности при выпуске токена средствами подсистемы

При выпуске токена средствами подсистемы жизненного цикла токенов, с записью на него сертификата для входа пользователя или при использовании токена для входа по паролю, на токен записывается информация служебного характера (рисунок 4.1).

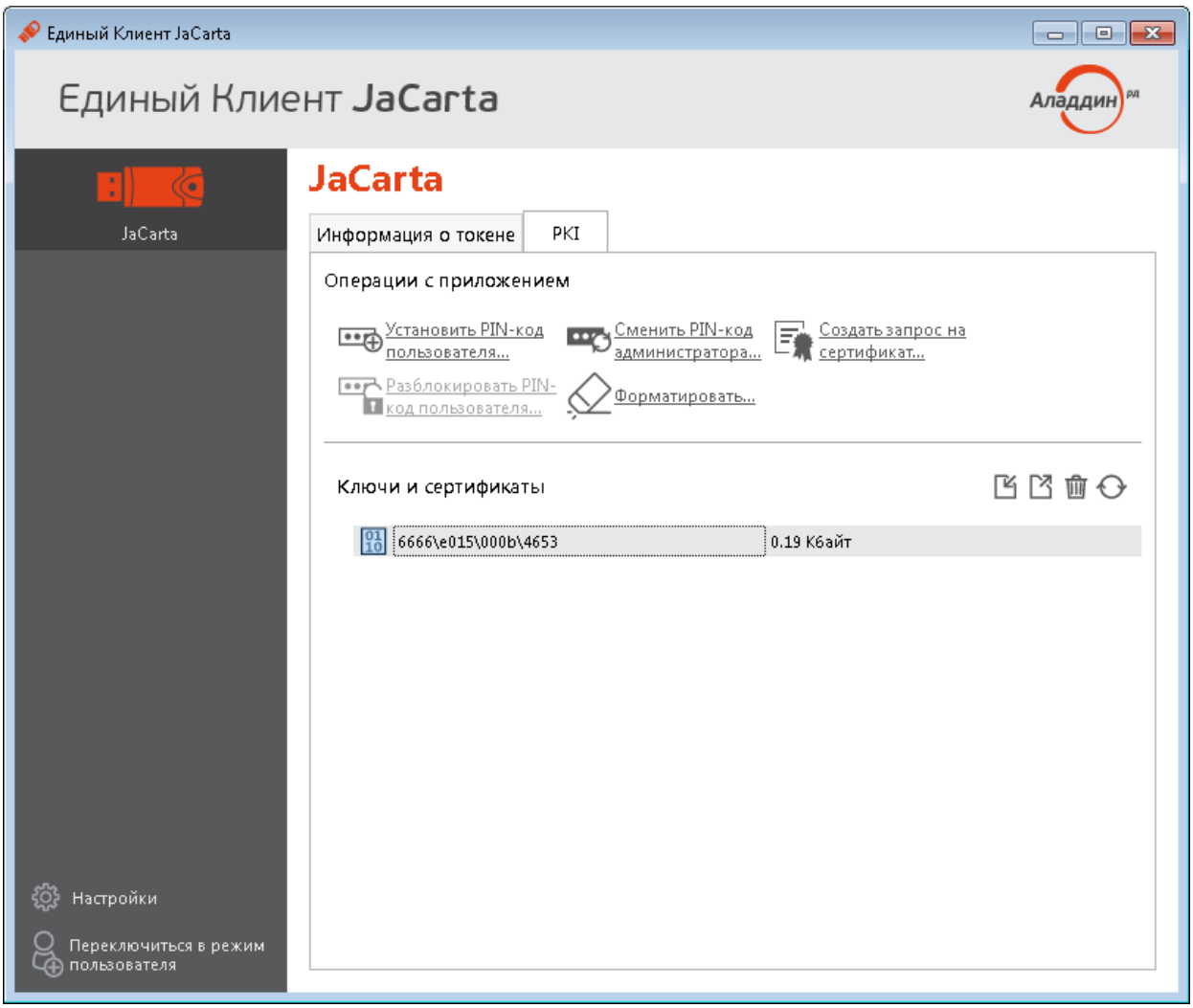


Рисунок 4.1 – Запись служебной информации при выпуске токена

Служебная информация на токене не подлежит удалению или редактированию. Указанные действия приведут к некорректной работе подсистемы с используемым токеном, и ошибкам при выполнении операций с ним.

Дальнейшие действия администратора по работе с подсистемой управления жизненным циклом токенов описаны в документе «СЗИ от НСД «Блокхост-Сеть 4». Руководство администратора безопасности. Часть 5. Управление токенами».

5 Настройка входа по сертификатам ЦС DogTag в ОС Linux

Перед выполнением настроек входа по сертификатам на рабочие станции под управлением ОС семейства Linux необходимо убедиться, что клиентская рабочая станция входит в домен FreeIPA, и что на рабочей станции установлена клиентская часть СЗИ от НСД «Блокхост-Сеть 4».

При установке клиента управления на ОС Astra Linux версии 1.6 необходимо убедиться в установке обновлений Update 6. Подробная инструкция по выполнению обновления до версии №20200722SE16 Update 6 приведена на сайте справочного центра Astra Linux https://wiki.astralinux.ru/pages/viewpage.action?pageId=103025136.

Установка клиентской части СЗИ на рабочие станции под управлением ОС семейства Linux описана в документе «СЗИ от НСД «Блокхост-Сеть 4». Руководство по инсталляции в ОС Linux».

5.1 Приоритет проверок средств аутентификации при входе

При входе пользователя в ОС семейства Linux действует следующая очередность проверок предъявляемых пользователем средств аутентификации по убыванию приоритета: вход по сертификату, вход по БВПП, вход по паролю:

1) При предъявлении пользователем токена, в первую очередь выполняется проверка на наличие на устройстве сертификата для входа:

   если сертификат для входа присутствует на токене (вне зависимости от истечения срока действия сертификата, наличия возможности безопасного входа по паролю), будет осуществлена попытка входа по сертификату.

2) При отсутствии на предъявляемом токене сертификата для входа, выполняется проверка на наличие на устройстве возможности безопасного входа по паролю:

   если безопасный вход по паролю возможен для пользователя, будет осуществлена попытка входа по паролю на токене.

3) В случае невозможности входа по сертификату и по паролю на токене, пользователь может воспользоваться входом по паролю без предъявления токена, при этом предъявляемое для входа устройство необходимо извлечь.

5.2 Настройка входа по сертификатам в ОС Альт

5.2.1 Ввод рабочей станции под управлением ОС Альт в домен FreeIPA

Перед вводом рабочей станции в домен, необходимо предварительно убедиться в **правильности настроек сети** (домен и все вводимые в него рабочие станции находятся в одной виртуальной подсети) и **в отсутствии расхождения по времени** между доменом и рабочей станцией, которая вводится в домен.

Ввод рабочей станции под управлением ОС Альт в домен FreeIPA может быть выполнен только одним способом с использованием инструментария sssd.

Подробное описание ввода в домен FreeIPA рабочей станции под управлением ОС Альт доступно на сайте справочного центра Альт https://www.altlinux.org/FreeIPA.

Для ввода в домен рабочей станции в ОС Альт необходимо последовательно выполнить следующие действия:

1) Скачать и установить все необходимые для работы пакеты:

```
sudo apt-get install task-auth-ad-sssd samba-client sssd-ad
sudo apt-get install task-auth-freeipa
```

2) Если используется домен, оканчивающийся на «.local», сразу выполнить следующую команду:

```
sudo chkconfig avahi-daemon off
sudo reboot
```

3) Задать имя клиентского компьютера следующей командой:

```
hostnamectl set-hostname <имя клиентского компьютера в нижнем регистре>.freeipa.local
```

при этом, имя клиентского компьютера обязательно должно содержать ***freeipa.local***.

4) В файле /etc/hosts указать:

```
10.72.9.11    alt9cafreeipa.freeipa.local alt9cafreeipa
10.72.9.11    freeipa.local
```

5) В *Центре управления системой* перейти в раздел ***Пользователи*** → ***Аутентификация***, выбрать пункт ***Домен FreeIPA*** и заполнить поля (рисунок 5.1).

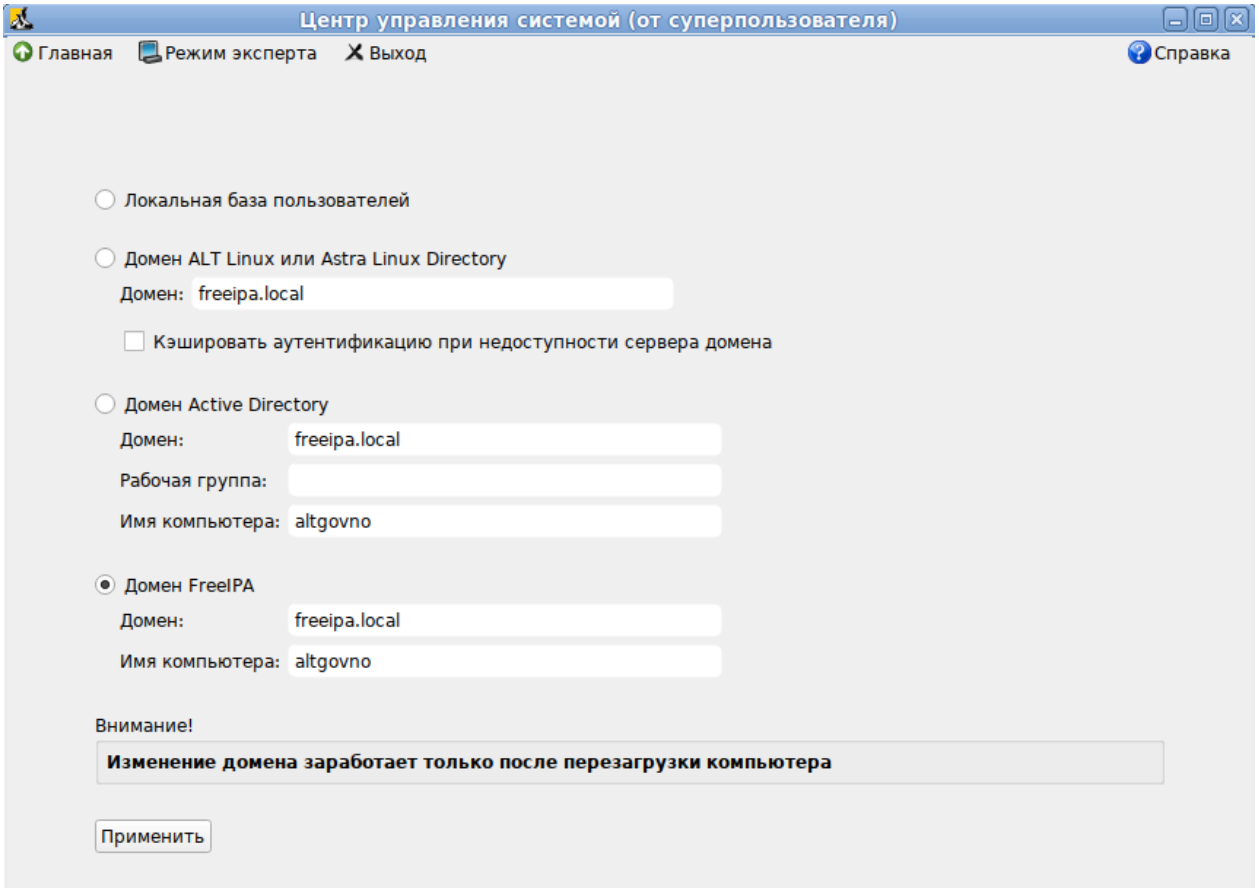


Рисунок 5.1 – Заполнение полей для ввода в домен FreeIPA

6) Нажать кнопку ***Применить***.

5.2.2 Общий порядок настройки для входа по сертификатам в ОС Альт

Для возможности аутентификации пользователя на клиентской рабочей станции под управлением СЗИ от НСД «Блокхост-Сеть 4» с использованием цифрового сертификата в ОС Альт необходимо выполнить следующие действия на клиентской рабочей станции:

1) Установить корневые сертификаты домена FreeIPA.
2) Настроить Policy Kit.
3) Установить пакеты для работы с токенами.
4) Проверить работу PC/SC интерфейса.
5) Проверить работу библиотек вендоров PKCS#11
6) Настроить и проверить работу службы Kerberos.
7) Настроить и проверить работу службу SSSD.
8) Включить запрос двойного ввода ПИН-кода.

5.2.3 Экспорт корневого сертификата FreeIPA

Для установки промежуточного и доверенного корневого сертификатов необходимо выполнить следующие действия:

1) На рабочей станции с установленным центром сертификации DogTag с помощью команды

```
sudo ls /root/
```

найти сертификаты *ipa.cer* и *root.cer* (рисунок 5.2).

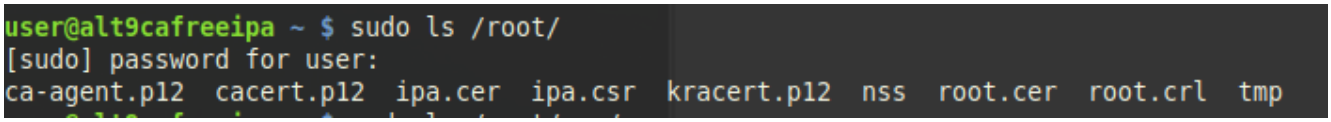


Рисунок 5.2 – Поиск сертификатов

2) Скопировать данные сертификаты в каталог пользователя с помощью команды:

```
sudo cp /root/<имя сертификата>.cer
```

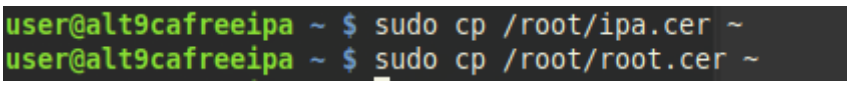


Рисунок 5.3 – Копирование сертификатов в каталог пользователя

3) При необходимости сменить владельца сертификатов с помощью команды

```
sudo chown user:user <имя сертификата>.cer:
```

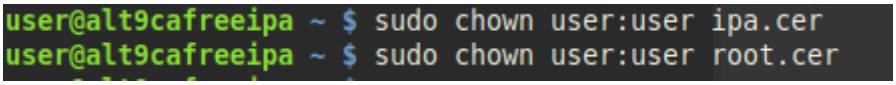


Рисунок 5.4 – Смена владельца сертификатов

4) Записать сертификаты на переносное устройство для установки на клиентской рабочей станции.

5) На клиентской рабочей станции под управлением СЗИ от НСД «Блокхост-Сеть 4» для удобства настройки сделать из файлов *ipa.cer* и *root.cer* один общий файл *ca.pem*:

```
cat ipa.cer > ca.pem

cat root.cer >> ca.pem
```

6) Для сохранения сертификатов создайте каталог */etc/certs/:*

```
sudo mkdir /etc/certs
```

7) Скопируйте в каталог */etc/certs/* сертификат центра сертификации (*ca.pem*):

```
sudo cp /<директория хранения сертификата> /etc/certs
```

5.2.4 Настройка Policy Kit в ОС Альт

При аутентификации пользователя в ОС Альт по паролю на токене или при переходе на двухфакторную аутентификацию с использованием пароля на токене возможна ситуация, когда не выполняется поиск токена.

Данная ситуация означает, что токен был заблокирован средствами ОС Альт. Для разблокировки доступа к токену необходимо настроить Policy Kit.

По умолчанию Policy Kit позволяет работать с токенами только в активной сессии (после входа в ОС) и только локальному пользователю. Для работы с токенами на этапе аутентификации доменных пользователей, необходимо поменять все запреты "*no*" на разрешения "*yes"* в файле ***/usr/share/polkit-1/actions/org.debian.pcsc-lite.policy***.

В итоге конфигурационный файл polkit примет следующий вид:

```
[user@altetobol Рабочий стол]$ cat /usr/share/polkit-1/actions/org.debian.pcsc-
lite.policy
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE policyconfig PUBLIC
 "-//freedesktop//DTD PolicyKit Policy Configuration 1.0//EN"
 "http://www.freedesktop.org/standards/PolicyKit/1.0/policyconfig.dtd">
<policyconfig>
  <vendor>The PCSC-lite Project</vendor>
  <vendor_url>https://pcsclite.apdu.fr/</vendor_url>
<!--  <icon_name>smart-card</icon_name> -->
  <action id="org.debian.pcsc-lite.access_pcsc">
    <description>Access to the PC/SC daemon</description>
    <message>Authentication is required to access the PC/SC daemon</message>
    <defaults>
      <allow_any>yes</allow_any>
      <allow_inactive>yes</allow_inactive>
      <allow_active>yes</allow_active>
    </defaults>
  </action>
  <action id="org.debian.pcsc-lite.access_card">
    <description>Access to the smart card</description>
    <message>Authentication is required to access the smart card</message>
    <defaults>
      <allow_any>yes</allow_any>
      <allow_inactive>yes</allow_inactive>
      <allow_active>yes</allow_active>
    </defaults>
  </action>
</policyconfig>
```

5.2.5 Установка пакетов для работы с токенами

Настройка работы выполняется на обновлённой пакетной базе и последнем ядре из репозитория:

```
apt-get update
apt-get dist-upgrade
update-kernel
reboot
```

Для работы с токенами необходимо установить следующие пакеты:

```
apt-get install librtpkcs11ecp libjcpkcs11 isbc-pkcs11 opensc pcsc-lite-ccid pcsc-lite pcsc-tools gnutls-utils libp11 libp11-kit
```

Если каких-то пакетов в репозитории нет, или с ними выявлены проблемы, необходимо установить их с сайта производителя:

- Актив Рутокен (библиотека librtpkcs11ecp) *https://www.rutoken.ru/support/download/pkcs/*;
- Аладдин JaCarta (библиотека libjcPKCS11-2) *https://www.aladdin-rd.ru/support/downloads/jacarta_client* (или в составе «Единого Клиента JaCarta»);
- ISBC ESMART (библиотека libisbc_pkcs11_main) *https://esmart.ru/download/*;

5.2.6 Проверка работы PC/SC интерфейса

Работу интерфейса PC/SC обеспечивает служба pcscd.service, которая при необходимости запускается через одноимённый программный интерфейс - pcscd.socket.

Для включения pcscd.socket необходимо воспользоваться командой:

```
systemctl enable --now pcscd.socket
```

Для проверки запуск службы необходимо выполнить команду:

```
systemctl status pcscd.socket
  pcscd.socket - PC/SC Smart Card Daemon Activation Socket
  Loaded: loaded (/lib/systemd/system/pcscd.socket; enabled; vendor preset: enabled)
  Active: active (running) since Tue 2022-05-17 17:38:36 MSK; 26min ago
  Listen: /run/pcscd/pcscd.comm (Stream)
  CGroup: /system.slice/pcscd.socket мая 17 17:38:36 alt82purefreeipa2.freeipa.local
systemd[1]: Listening on PC/SC Smart Card Daemon Activation Socket
```

5.2.7 Проверка работы библиотек вендоров PKCS#11

Для проверки работы библиотек вендоров PKCS#11 необходимо воспользоваться командой:

- для ESMART Token:

```
$ pkcs11-tool --module /usr/lib64/libisbc_pkcs11_main.so --list-token-slots
```

- для Rutoken ECP:

```
$ pkcs11-tool --module /usr/lib64/librtpkcs11ecp.so --list-token-slots
```

- для JaCarta:

```
$ pkcs11-tool --module /usr/lib64/libjcPKCS11-2.so --list-token-slots
```

5.2.8 Настройка и проверка работы службы Kerberos

Для настройки *kerberos* в */etc/krb5.conf.* Секцию [libdefaults] необходимо дополнить следующими строками:

```
sudo pluma /etc/krb5.conf
   pkinit_anchors = DIR:/etc/certs/
   pkinit_kdc_hostname = s09dc.pki.local
   pkinit_eku_checking = kpServerAuth
   pkinit_cert_match = <ISSUER>.*pki-S09DC-CA
   canonicalize = True
```

где:

- *pkinit_anchors* директория расположения сертификата;
- *pkinit_kdc_hostname* название домена;
- *pkinit_cert_match* название корневого сертификата.

Параметр *canonicalize = True* может влиять на смену пароля доменному пользователю по требованию домена и БВПП. Если смены пароля не происходит, и система возвращает ошибку ***KDC reply did not match expectations***, строку нужно закомментировать.

Для проверки работоспособности службы Kerberos и получения билета Kerberos необходимо воспользоваться командой:

- для ESMART Token:

```
kinit -X X509_user_identity=PKCS11:libisbc_pkcs11_main.so <имя пользователя>
   esmart_64 PIN:
klist
   Ticket cache: KEYRING:persistent:500:krb_ccache_iXXIg94
   Default principal: <имя пользователя>@TEST.ALT
   Valid starting Expires Service principal
   20.06.2022 19:33:36 21.06.2022 05:33:36 krbtgt/TEST.ALT@TEST.ALT
   renew until 27.06.2022 19:33:26
```

- для Rutoken ECP:

```
kinit -X X509_user_identity=PKCS11:librtpkcs11ecp.so <имя пользователя>
   RuTokenECP PIN:
klist
   Ticket cache: KEYRING:persistent:500:500
   Default principal: <имя пользователя>@TEST.ALT
   Valid starting Expires Service principal
   20.06.2022 19:32:54 21.06.2022 05:32:54 krbtgt/TEST.ALT@TEST.ALT
   renew until 27.06.2022 19:32:46
```

- для JaCarta:

```
kinit -X X509_user_identity=PKCS11:libjcPKCS11-2.so <имя пользователя>
   jacarta PIN:
klist
   Ticket cache: KEYRING:persistent:500:krb_ccache_ppoB6Qg
   Default principal: <имя пользователя>@TEST.ALT
   Valid starting Expires Service principal
   20.06.2022 19:34:46 21.06.2022 05:34:46 krbtgt/TEST.ALT@TEST.ALT
   renew until 27.06.2022 19:34:26
```

При возникновении ошибок при проверке службы Kerberos, для более полной информации по процессу PKINIT необходимо добавить опцию вывода отладочной информации в консоль — KRB5_TRACE=/dev/stdout.

Команда получения билета Kerberos с отладкой:

```
KRB5_TRACE=/dev/stdout kinit -X X509_user_identity=PKCS11:libjcPKCS11-2.so <имя пользователя>
```

Если токен двухапплетный, то дополнительно нужно указать его имя:

```
kinit -X X509_user_identity=PKCS11:module_name=libjcPKCS11-2.so:token=JaCarta <имя пользователя>
```

Для удаления билета Kerberos выполнить команду:

```
kdestroy
```

5.2.9 Настройка и проверка службы SSSD

Необходимо учитывать, что библиотек не должно быть несколько. Нужно выбрать только один токен для работы и добавить для него библиотеку.

Для добавления библиотеки в список необходимо воспользоваться командой:

- для Rutoken ECP:

```
sudo echo "module: /usr/lib64/librtpkcs11ecp.so">/etc/pkcs11/modules/rutokenecp.module
```

- для JaCarta:

```
sudo echo "module: /usr/lib64/libjcPKCS11-2.so">/etc/pkcs11/modules/jacarta.module
```

Для проверки работы sssd с токенами необходимо воспользоваться командой:

- для ESMART Token:

```
/usr/libexec/sssd/p11_child --ca_db=/etc/certs/ca.cer -pre
  esmart_64
  /usr/lib64/pkcs11/libisbc_pkcs11_main.so
  6C652D52545F4A435F45535F323034382D663164626165643 82D313563642D34332D3038343637le-
RT_JC_ES_2048-f1dbaed8-15cd-43-08467
MIIGozCCBYugAwIBAgITJwAAAA+7dZWQmcp+kAAAAAAADzANBgkqhkiG9w0BAQsFADBJMRMwEQYKCZImiZPgqCa6bW
GgNqxgCQQpv7ld0N3IzFmTI=
```

- для Rutoken ECP:

```
/usr/libexec/sssd/p11_child --ca_db=/etc/certs/ca.cer -pre
  RuTokenECP
  /usr/lib64/pkcs11/librtpkcs11ecp.so
  6C652D52545F4A435F45535F323034382D6363383134323133 2D653232302D34362D30353632355F451e-
RT_JC_ES_2048-cc814213-e220-46-
05625_EMIIGozCCBYugAwIBAgITJwAAAA1nYbE+RFds5wAAAAAADTANBgkqhkiG9w0BAQsFADBJMRMwEQYKCZImiZP
0y0AxXzGs4Ex/slygrEGCAO3MJuN3k=
```

- для JaCarta:

Для проверки работы службы sssd с токеном JaCarta-2 SE необходимо дополнительно указать слот при помощи схемы pkcs11:<URI>, которая поддерживается в sssd.
С полным перечнем параметров pkcs11:<URI> можно ознакомиться в RFC7512 (*https://datatracker.ietf.org/doc/html/rfc7512#section-2.1*).

Создать символическую ссылку в каталоге */usr/lib64/pkcs11/* на библиотеку *libjckt2.so* (чтобы не было ошибок с первым слотом JaCarta GOST 2.0; исправлено с версии 2.7.4-alt6 пакета libjcpkcs11):

```
cd /usr/lib64/pkcs11/
ln -s ../libjckt2.so

/usr/libexec/sssd/p11_child  --ca_db=/etc/certs/ca.cer  --pre  --uri=pkcs11:token=jacarta-
slot-laser
  jacarta-slot-laser
  /usr/lib64/pkcs11/libjcPKCS11-2.so
  6C652D52545F4A435F45535F323034382D393335616430393 32D653866662D34632D3237373131le-
RT_JC_ES_2048-935ad093-e8ff-4c-
```

```
27711MIIGozCCBYugAwIBAgITJwAAAA4vyUvSfioOvAAAAAAADjANBgkqhkiG9w0BAQsFADBJMRMwEQYKCZImiZPP2
6XcAxrmGfjfT53PYqC30uk+W7S7Q=
```

Если сертификат не найден, посмотреть диагностику, указав дополнительные параметры:

```
--debug-level=10 --logger=stderr
```

Получить идентификатор токена:

```
sudo modutil -dbdir /etc/pki/nssdb -add "Rutoken PKCS11" -libfile librtpkcs11ecp.so
sudo modutil -dbdir /etc/pki/nssdb -list
```

Для настройки службы SSSD необходимо отредактировать файл ***/etc/sssd/sssd.conf*** командой:

```
sudo pluma /etc/sssd/sssd.conf
```

В секцию [pam] добавить строки:

```
  pam_cert_auth = True
  pam_p11_allowed_services = +mate-screensaver, +lightdm, +kf5-screenlocker, +sddm
  pam_cert_db_path = /etc/certs/ca.cer
```

где:

pam_cert_db_path – название вашего сертификата.

Если на токене несколько апплетов, нужно указать с каким именно работать. Для этого необходимо прописать p11_uri токена:

```
  p11_uri
= pkcs11:token=NSS%20Generic%20Crypto%20Services;manufacturer=Mozilla%20Foundation;serial=
0000000000000000;model=NSS%203
```

Так как аутентификация по токену занимает больше времени, чем по паролю (поиск сертификата на токене, проверка сертификата по протоколу OCSP и т.п.) необходимо увеличить таймауты для Kerberos и для p11_child, добавив следующие параметры:

```
  [domain/test.alt]
  ...
  krb5_auth_timeout = 60
  [pam]
  ...
  p11_child_timeout = 60
```

После изменения конфигурации необходимо перезапустить службу SSSD, очистив при этом кэш sssd:

```
systemctl stop sssd && sss_cache -E && systemctl start sssd
```

Проверить доступность службы SSSD для библиотеки:

```
p11-kit list-modules
  p11-kit-trust: p11-kit-trust.so
      library-description: PKCS#11 Kit Trust Module
      library-manufacturer: PKCS#11 Kit
      library-version: 0.23
      token: System Trust
          manufacturer: PKCS#11 Kit
          model: p11-kit-trust
          serial-number: 1
          hardware-version: 0.23
          flags:
                 write-protected
                 token-initialized
      token: Default Trust
          manufacturer: PKCS#11 Kit
          model: p11-kit-trust
          serial-number: 1
          hardware-version: 0.23
          flags:
                 write-protected
                 token-initialized
  isbc: libisbc_pkcs11_main.so
      library-description:
      library-manufacturer: ISBC
      library-version: 1.0
  jcpkcs11: libjcPKCS11-2.so
      library-description: JaCarta PKCS#11 module
      library-manufacturer: Aladdin R.D.
      library-version: 2.4
  rutokenecp: librtpkcs11ecp.so
      library-description: Rutoken ECP PKCS #11 library
      library-manufacturer: Aktiv Co.
      library-version: 2.3
```

Если после настройки входа по сертификату при входе пользователя видим ошибку ***generic failure: GSSAPI Error: Unspecified GSS failure. Minor code may provide more information (Server not found in Kerberos database)*** или ***Нет связи с сервером*** необходимо добавить в */etc/hosts* домен:

```
10.72.9.2 s09dc.pki.local pki.local
```

5.3 Настройка входа по сертификатам в Astra Linux

5.3.1 Ввод рабочей станции под управлением ОС Astra Linux в домен FreeIPA

Перед вводом рабочей станции в домен, необходимо предварительно убедиться в **правильности настроек сети** (домен и все вводимые в него рабочие станции находятся в одной виртуальной подсети) и **в отсутствии расхождения по времени** между доменом и рабочей станцией, которая вводится в домен.

Подробное описание ввода в домен FreeIPA рабочей станции под управлением ОС Astra Linux доступно на сайте справочного центра Astra Linux https://wiki.astralinux.ru/pages/viewpage.action?pageId=60359750.

Для ввода в домен рабочей станции в ОС Astra Linux необходимо последовательно выполнить следующие действия:

1) Задать имя клиентского компьютера следующей командой:

```
hostnamectl set-hostname <имя клиентского компьютера в нижнем регистре>.freeipa.local
```

при этом, имя клиентского компьютера обязательно должно содержать ***freeipa.local***.

2) В файле /etc/hosts указать:

```
127.0.1.1    <имя клиентского компьютера в нижнем регистре>.freeipa.local pc34
10.72.9.11    alt9cafreeipa.freeipa.local alt9cafreeipa
10.72.9.11    freeipa.local
```

3) Проверить доступность клиентского компьютера командой:

```
ping alt9cafreeipa.freeipa.local
```

4) Установить графический инструмент **fly-admin-freeipa-client** для ввода рабочей станции в домен:

```
sudo apt install fly-admin-freeipa-client
```

5) После установки пакет будет доступен в графическом меню ***Пуск → Панель управления → Сеть → Настройка FreeIPA client Fly***.

6) Для ввода рабочей станции в ОС Astra Linux в домен Active Directory необходимо выполнить запуск инструмента, указать имя домена, имя и пароль администратора и нажать кнопку ***Подключиться.***

5.3.2 Общий порядок настройки для входа по сертификатам в ОС Astra Linux

Для возможности аутентификации пользователя на клиентской рабочей станции под управлением СЗИ от НСД «Блокхост-Сеть 4» с использованием цифрового сертификата в ОС Astra Linux необходимо выполнить следующие действия на клиентской рабочей

станции:

1) Установить корневые сертификаты домена FreeIPA.

2) Установить пакеты для работы с токенами.

3) Настроить и проверить работу службы Kerberos.

5.3.3 Экспорт корневого сертификата FreeIPA

Для установки промежуточного и доверенного корневого сертификатов необходимо выполнить следующие действия:

1) На рабочей станции с установленным центром сертификации DogTag с помощью команды

```
sudo ls /root/
```

найти сертификаты *ipa.cer* и *root.cer* (рисунок 5.5).

```
user@alt9cafreeipa ~ $ sudo ls /root/
[sudo] password for user:
ca-agent.p12  cacert.p12  ipa.cer  ipa.csr  kracert.p12  nss  root.cer  root.crl  tmp
```

Рисунок 5.5 – Поиск сертификатов

2) Скопировать данные сертификаты в каталог пользователя с помощью команды:

```
sudo cp /root/<имя сертификата>.cer
```

```
user@alt9cafreeipa ~ $ sudo cp /root/ipa.cer ~
user@alt9cafreeipa ~ $ sudo cp /root/root.cer ~
```

Рисунок 5.6 – Копирование сертификатов в каталог пользователя

3) При необходимости сменить владельца сертификатов с помощью команды

```
sudo chown user:user <имя сертификата>.cer:
```

```
user@alt9cafreeipa ~ $ sudo chown user:user ipa.cer
user@alt9cafreeipa ~ $ sudo chown user:user root.cer
```

Рисунок 5.7 – Смена владельца сертификатов

4) Записать сертификаты на переносное устройство для установки на клиентской рабочей станции.

5) На клиентской рабочей станции под управлением СЗИ от НСД «Блокхост-Сеть 4» для удобства настройки сделать из файлов *ipa.cer* и *root.cer* один общий файл *ca.pem*:

```
cat ipa.cer > ca.pem
```

```
cat root.cer >> ca.pem
```

6) Для сохранения сертификатов создайте каталог */etc/certs/:*

```
sudo mkdir /etc/certs
```

7) Скопируйте в каталог */etc/certs/* сертификат центра сертификации (*ca.pem*):

```
sudo cp /<директория хранения сертификата> /etc/certs
```

5.3.4 Установка пакетов для работы с токенами

Для работы с токенами необходимо установить следующие пакеты:

```
$ sudo apt install libccid pcscd libpcsclite1 krb5-pkinit libsasl2-modules-gssapi-mit
```

где:

- *libccid* драйвер PC/SC для считывателей смарт-карт;
- *pcscd* обеспечение доступа к смарт-картам через PC/SC;
- *libpcsclite1* связующее ПО доступа к смарт-картам, использующее PC/SC;
- *krb5-pkinit* модуль расширения Kerberos для аутентификации по сертификатам;
- *libsasl2-modules-gssapi-mit* - подключаемые модули аутентификации.

Если каких-то пакетов в репозитории нет или с ними выявлены проблемы, необходимо установить их с сайта производителя:

- Актив Рутокен (библиотека librtpkcs11ecp) *https://www.rutoken.ru/support/download/pkcs/*;
- Аладдин JaCarta (библиотека libjcPKCS11-2) *https://www.aladdin-rd.ru/support/downloads/jacarta_client* (или в составе «Единого Клиента JaCarta»);
- ISBC ESMART (библиотека libisbc_pkcs11_main) *https://esmart.ru/download/*;

5.3.5 Настройка и проверка работы службы Kerberos

Для настройки *kerberos* в */etc/krb5.conf.* Секцию [libdefaults] необходимо дополнить следующими строками:

```
sudo vi /etc/krb5.conf
   default_realm = PKI.LOCAL
   pkinit_identities = PKCS11:module_name=/usr/lib/librtpkcs11ecp.so
   pkinit_eku_checking = kpServerAuth
   pkinit_anchors = DIR:/etc/certs/
```

```
pkinit_kdc_hostname = S09DC.PKI.LOCAL
```

где:

- *pkinit_identities* библиотека используемого токена (только для входа по сертификату, для входа по БВПП эту строку необходимо удалить);
- *pkinit_anchors* директория расположения сертификата;
- *pkinit_kdc_hostname* название домена.

Подробнее настройка pki_init описана https://web.mit.edu/kerberos/krb5-1.12/doc/admin/pkinit.html.

При использовании двухапплетного токена необходимо дополнительно указывать имя апплета например:

```
pkinit_identities = PKCS11:module_name=/usr/lib/libjcPKCS11-2.so:token=JaCarta
```

Для проверки работоспособности службы Kerberos и получения билета Kerberos необходимо воспользоваться командой:

- для ESMART Token:

```
kinit -X X509_user_identity=PKCS11:libisbc_pkcs11_main.so <имя пользователя>
   esmart_64 PIN:
klist
   Ticket cache: KEYRING:persistent:500:krb_ccache_iXXIg94
   Default principal: <имя пользователя>@TEST.ALT
   Valid starting Expires Service principal
   20.06.2022 19:33:36 21.06.2022 05:33:36 krbtgt/TEST.ALT@TEST.ALT
   renew until 27.06.2022 19:33:26
```

- для Rutoken ECP:

```
kinit -X X509_user_identity=PKCS11:librtpkcs11ecp.so <имя пользователя>
   RuTokenECP PIN:
klist
   Ticket cache: KEYRING:persistent:500:500
   Default principal: <имя пользователя>@TEST.ALT
   Valid starting Expires Service principal
   20.06.2022 19:32:54 21.06.2022 05:32:54 krbtgt/TEST.ALT@TEST.ALT
   renew until 27.06.2022 19:32:46
```

- для JaCarta:

```
kinit -X X509_user_identity=PKCS11:libjcPKCS11-2.so <имя пользователя>
   jacarta PIN:
```

```
klist
   Ticket cache: KEYRING:persistent:500:krb_ccache_ppoB6Qg
   Default principal: <имя пользователя>@TEST.ALT
   Valid starting Expires Service principal
   20.06.2022 19:34:46 21.06.2022 05:34:46 krbtgt/TEST.ALT@TEST.ALT
   renew until 27.06.2022 19:34:26
```

При возникновении ошибок при проверке службы Kerberos, для более полной информации по процессу PKINIT добавьте опцию вывода отладочной информации в консоль — KRB5_TRACE=/dev/stdout.

Команда получения билета Kerberos с отладкой:

```
KRB5_TRACE=/dev/stdout kinit -X X509_user_identity=PKCS11:libjcPKCS11-2.so <имя пользователя>
```

Для удаления билета Kerberos необходимо выполнить команду:

```
kdestroy
```

Для синхронизации времени с доменом выполнить:

```
sudo ntpdate s09dc.pki.local
```

Если после настройки входа по сертификату при входе пользователя видим ошибку ***generic failure: GSSAPI Error: Unspecified GSS failure. Minor code may provide more information (Server not found in Kerberos database)*** или ***Нет связи с сервером*** необходимо добавить в */etc/hosts* домен:

```
10.72.9.2 s09dc.pki.local pki.local
```

5.4 Настройка входа по сертификатам в РЕД ОС

5.4.1 Ввод рабочей станции под управлением РЕД ОС в домен FreeIPA

Перед вводом рабочей станции в домен, необходимо предварительно убедиться в **правильности настроек сети** (домен и все вводимые в него рабочие станции находятся в одной виртуальной подсети) и **в отсутствии расхождения по времени** между доменом и рабочей станцией, которая вводится в домен.

Подробное описание ввода в домен FreeIPA рабочей станции под управлением РЕД ОС доступно на сайте справочного центра РЕД ОС https://redos.red-soft.ru/base/arm/arm-domen/redos-in-ipa/.

Для ввода в домен рабочей станции в РЕД ОС необходимо последовательно выполнить следующие действия:

1) Отключить SELINUX в файле **/etc/selinux/config**:

```
SELINUX=disabled
```

После изменения перезагрузить ОС.

2) Задать имя клиентского компьютера следующей командой:

```
hostnamectl set-hostname <Имя компьютера>.<домен>
```

при этом, если домен **freeipa.local**, то имя клиентского компьютера обязательно должно содержать **client.freeipa.local**.

3) В файле /etc/hosts указать:

```
10.72.9.11    <Имя компьютера>.<домен> <Имя компьютера>
10.72.9.11    <домен>
```

4) Проверить доступность клиентского компьютера командой:

```
ping <Имя компьютера>.<домен>
```

5) Установить ipa-client:

```
dnf -y install ipa-client
```

6) Присоединить клиента к домену IPA:

```
ipa-client-install --domain freeipa.local --server alt9cafreeipa.freeipa.local --force-join
--mkhomedir
```

7) Согласитесь, введя *yes,* для продолжения с фиксированными значениями:

```
Proceed with fixed values and no DNS discovery? [no]:
```

8) Нажмите *Enter* в следующем сообщении:

```
Do you want to configure chrony with NTP server or pool address? [no]:
```

9) Согласитесь с продолжением настройки системы:

```
Client hostname: redos2.freeipa.local
Realm: FREEIPA.LOCAL
DNS Domain: freeipa.local
IPA Server: alt9cafreeipa.freeipa.local
BaseDN: dc=freeipa,dc=local

Continue to configure the system with these values? [no]:
```

10) Введите имя администратора (возможно использовать администратора по умолчанию IPA, который был создан при установке сервера) и пароль администратора:

```
User authorized to enroll computers: admin
```

11) При успешной установке появится сообщение:

```
Client configuration complete.
The ipa-client-install command was successful
```

5.4.2 Общий порядок настройки для входа по сертификатам в РЕД ОС

Для возможности аутентификации пользователя на клиентской рабочей станции под управлением СЗИ от НСД «Блокхост-Сеть 4» с использованием цифрового сертификата в ОС Альт необходимо выполнить следующие действия на клиентской рабочей станции:

1) Установить корневые сертификаты домена FreeIPA.
2) Настроить Policy Kit.
3) Установить пакеты для работы с токенами.
4) Проверить работу PC/SC интерфейса.
5) Проверить работу библиотек вендоров PKCS#11
6) Настроить и проверить работу службы Kerberos.
7) Настроить и проверить работу службу SSSD.

5.4.3 Экспорт корневого сертификата FreeIPA

Для установки промежуточного и доверенного корневого сертификатов необходимо выполнить следующие действия:

1) На рабочей станции с установленным центром сертификации DogTag с помощью команды

```
sudo ls /root/
```

найти сертификаты *ipa.cer* и *root.cer* (рисунок 5.2).

```
user@alt9cafreeipa ~ $ sudo ls /root/
[sudo] password for user:
ca-agent.p12  cacert.p12  ipa.cer  ipa.csr  kracert.p12  nss  root.cer  root.crl  tmp
```

Рисунок 5.8 – Поиск сертификатов

2) Скопировать данные сертификаты в каталог пользователя с помощью команды:

```
sudo cp /root/<имя сертификата>.cer
```

```
user@alt9cafreeipa ~ $ sudo cp /root/ipa.cer ~
user@alt9cafreeipa ~ $ sudo cp /root/root.cer ~
```

Рисунок 5.9 – Копирование сертификатов в каталог пользователя

3) При необходимости сменить владельца сертификатов с помощью команды

```
sudo chown user:user <имя сертификата>.cer:
```

```
user@alt9cafreeipa ~ $ sudo chown user:user ipa.cer
user@alt9cafreeipa ~ $ sudo chown user:user root.cer
```

Рисунок 5.10 – Смена владельца сертификатов

4) Записать сертификаты на переносное устройство для установки на клиентской рабочей станции.

5) На клиентской рабочей станции под управлением СЗИ от НСД «Блокхост-Сеть 4» для удобства настройки сделать из файлов *ipa.cer* и *root.cer* один общий файл *ca.pem*:

```
cat ipa.cer > ca.pem

cat root.cer >> ca.pem
```

6) Для сохранения сертификатов создайте каталог */etc/certs/:*

```
sudo mkdir /etc/certs
```

7) Скопируйте в каталог */etc/certs/* сертификат центра сертификации (*ca.pem*):

```
sudo cp /<директория хранения сертификата> /etc/certs
```

5.4.4 Настройка Policy Kit в РЕД ОС

При аутентификации пользователя в РЕД ОС по паролю на токене или при переходе на двухфакторную аутентификацию с использованием пароля на токене возможна ситуация, когда не выполняется поиск токена.

Данная ситуация означает, что токен был заблокирован средствами РЕД ОС. Для разблокировки доступа к токену необходимо настроить Policy Kit.

По умолчанию Policy Kit позволяет работать с токенами только в активной сессии (после входа в ОС) и только локальному пользователю. Для работы с токенами на этапе аутентификации доменных пользователей, необходимо поменять все запреты «*no*» на разрешения «*yes*» в файле ***/usr/share/polkit-1/actions/org.debian.pcsc-lite.policy***.

В итоге конфигурационный файл polkit примет следующий вид:

```
[user@altetobol Рабочий стол]$ cat /usr/share/polkit-1/actions/org.debian.pcsc-
lite.policy
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE policyconfig PUBLIC
 "-//freedesktop//DTD PolicyKit Policy Configuration 1.0//EN"
 "http://www.freedesktop.org/standards/PolicyKit/1.0/policyconfig.dtd">
<policyconfig>
  <vendor>The PCSC-lite Project</vendor>
  <vendor_url>https://pcsclite.apdu.fr/</vendor_url>
<!--  <icon_name>smart-card</icon_name> -->
  <action id="org.debian.pcsc-lite.access_pcsc">
    <description>Access to the PC/SC daemon</description>
    <message>Authentication is required to access the PC/SC daemon</message>
    <defaults>
      <allow_any>yes</allow_any>
      <allow_inactive>yes</allow_inactive>
      <allow_active>yes</allow_active>
    </defaults>
  </action>
  <action id="org.debian.pcsc-lite.access_card">
    <description>Access to the smart card</description>
    <message>Authentication is required to access the smart card</message>
    <defaults>
      <allow_any>yes</allow_any>
      <allow_inactive>yes</allow_inactive>
      <allow_active>yes</allow_active>
    </defaults>
  </action>
</policyconfig>
```

5.4.5 Установка пакетов для работы с токенами

Для работы с токенами необходимо установить следующие пакеты:

```
dnf install opensc pcsc-lite-ccid pcsc-lite pcsc-tools gnutls-utils libp11
```

Если каких-то пакетов в репозитории нет, или с ними выявлены проблемы, необходимо установить их с сайта производителя:

- Актив Рутокен (библиотека librtpkcs11ecp) *https://www.rutoken.ru/support/download/pkcs/*;
- Аладдин JaCarta (библиотека libjcPKCS11-2) *https://www.aladdin-rd.ru/support/downloads/jacarta_client* (или в составе «Единого Клиента JaCarta»);
- ISBC ESMART (библиотека libisbc_pkcs11_main) *https://esmart.ru/download/*;

5.4.6 Проверка работы PC/SC интерфейса

Работу интерфейса PC/SC обеспечивает служба pcscd.service, которая при необходимости запускается через одноимённый программный интерфейс - pcscd.socket.

Для включения pcscd.socket необходимо воспользоваться командой:

```
systemctl enable --now pcscd.socket
```

Для проверки запуск службы необходимо выполнить команду:

```
systemctl status pcscd.socket
   pcscd.socket - PC/SC Smart Card Daemon Activation Socket
   Loaded: loaded (/lib/systemd/system/pcscd.socket; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2022-05-17 17:38:36 MSK; 26min ago
   Listen: /run/pcscd/pcscd.comm (Stream)
   CGroup:  /system.slice/pcscd.socket  мая  17  17:38:36  alt82purefreeipa2.freeipa.local
systemd[1]: Listening on PC/SC Smart Card Daemon Activation Socket
```

Убедиться, что токен виден в выводе утилиты pcsc_scan, запущенной с правами пользователя, и однозначно определяется его модель:

```
pcsc_scan -r
   Using reader plug'n play mechanism
   Scanning present readers...
   0: Aktiv Rutoken ECP 00 00
```

5.4.7 Проверка работы библиотек вендоров PKCS#11

Для проверки работы библиотек вендоров PKCS#11 необходимо воспользоваться командой:

- для ESMART Token:

```
pkcs11-tool --module /usr/lib64/libisbc_pkcs11_main.so --list-token-slots
```

- для Rutoken ECP:

```
pkcs11-tool --module /usr/lib64/librtpkcs11ecp.so --list-token-slots
```

- для JaCarta:

```
pkcs11-tool --module /usr/lib64/libjcPKCS11-2.so --list-token-slots
```

5.4.8 Настройка и проверка работы службы Kerberos

Для настройки *kerberos* в */etc/krb5.conf.* Секцию [libdefaults] необходимо дополнить следующими строками:

```
sudo pluma /etc/krb5.conf
   default_ccache_name = KEYRING:persistent:%{uid}
   pkinit_anchors = DIR:/etc/certs/
   pkinit_kdc_hostname = S09DC.PKI.LOCAL
   pkinit_eku_checking = kpServerAuth
   pkinit_cert_match = <ISSUER>.*pki-S09DC-CA
```

Параметры pkinit_anchors и default_ccache_name, что были по умолчанию -

закомментировать (поставить символ # в начале строки).

где:

- *pkinit_anchors* директория расположения сертификата;
- *pkinit_kdc_hostname* название домена;
- *pkinit_cert_match* название корневого сертификата.

  Параметр *canonicalize = True* может влиять на смену пароля доменному пользователю по требованию домена и БВПП. Если смены пароля не происходит, и система возвращает ошибку ***KDC reply did not match expectations***, строку нужно закомментировать.

Для проверки работоспособности службы Kerberos и получения билета Kerberos необходимо воспользоваться командой:

- для ESMART Token:

```
kinit -X X509_user_identity=PKCS11:libisbc_pkcs11_main.so <имя пользователя>
   esmart_64 PIN:
klist
   Ticket cache: KEYRING:persistent:500:krb_ccache_iXXIg94
   Default principal: <имя пользователя>@TEST.ALT
   Valid starting Expires Service principal
   20.06.2022 19:33:36 21.06.2022 05:33:36 krbtgt/TEST.ALT@TEST.ALT
   renew until 27.06.2022 19:33:26
```

- для Rutoken ECP:

```
kinit -X X509_user_identity=PKCS11:librtpkcs11ecp.so <имя пользователя>
   RuTokenECP PIN:
klist
   Ticket cache: KEYRING:persistent:500:500
   Default principal: <имя пользователя>@TEST.ALT
   Valid starting Expires Service principal
   20.06.2022 19:32:54 21.06.2022 05:32:54 krbtgt/TEST.ALT@TEST.ALT
   renew until 27.06.2022 19:32:46
```

- для JaCarta:

```
kinit -X X509_user_identity=PKCS11:libjcPKCS11-2.so <имя пользователя>
   jacarta PIN:
klist
   Ticket cache: KEYRING:persistent:500:krb_ccache_ppoB6Qg
   Default principal: <имя пользователя>@TEST.ALT
   Valid starting Expires Service principal
```

```
  20.06.2022 19:34:46 21.06.2022 05:34:46 krbtgt/TEST.ALT@TEST.ALT
  renew until 27.06.2022 19:34:26
```

При возникновении ошибок при проверке службы Kerberos, для более полной информации по процессу PKINIT необходимо добавить опцию вывода отладочной информации в консоль — KRB5_TRACE=/dev/stdout.

Команда получения билета Kerberos с отладкой:

```
KRB5_TRACE=/dev/stdout kinit -X X509_user_identity=PKCS11:libjcPKCS11-2.so <имя пользователя>
```

Если токен двухапплетный, то дополнительно нужно указать его имя:

```
kinit -X X509_user_identity=PKCS11:module_name=libjcPKCS11-2.so:token=JaCarta <имя пользователя>
```

Для удаления билета Kerberos выполнить команду:

```
kdestroy
```

5.4.9 Настройка и проверка службы SSSD

Необходимо учитывать, что библиотек не должно быть несколько. Нужно выбрать только один токен для работы и добавить для него библиотеку.

Для добавления библиотеки в список необходимо воспользоваться командой:

- для Rutoken ECP:

```
sudo echo "module: /usr/lib64/librtpkcs11ecp.so">/etc/pkcs11/modules/rutokenecp.module
```

- для JaCarta:

```
sudo echo "module: /usr/lib64/libjcPKCS11-2.so">/etc/pkcs11/modules/jacarta.module
```

Для добавления модуля к p11-kit выполните команду:

```
pluma /usr/share/p11-kit/modules/Rutoken.module
  module:/usr/lib64/librtpkcs11ecp.so
```

Для проверки работы sssd с токенами необходимо воспользоваться командой:

- для ESMART Token:

```
/usr/libexec/sssd/p11_child --ca_db=/etc/certs/ca.cer -pre
  esmart_64
  /usr/lib64/pkcs11/libisbc_pkcs11_main.so
```

```
  6C652D52545F4A435F45535F323034382D663164626165643820313563642D34332D30383436371e-
RT_JC_ES_2048-f1dbaed8-15cd-43-08467
MIIGozCCBYugAwIBAgITJwAAAA+7dZWQmcp+kAAAAAAADzANBgkqhkiG9w0BAQsFADBJMRMwEQYKCZImiZPgqCa6bW
GgNqxgCQQpv7ld0N3IzFmTI=
```

- для Rutoken ECP:

```
/usr/libexec/sssd/p11_child --ca_db=/etc/certs/ca.cer -pre
  RuTokenECP
  /usr/lib64/pkcs11/librtpkcs11ecp.so
  6C652D52545F4A435F45535F323034382D636338313431332D653232302D34362D30353632355F451e-
RT_JC_ES_2048-cc814213-e220-46-
05625_EMIIGozCCBYugAwIBAgITJwAAAA1nYbE+RFds5wAAAAAADTANBgkqhkiG9w0BAQsFADBJMRMwEQYKCZImiZP
0y0AxXzGs4Ex/slygrEGCAO3MJuN3k=
```

- для JaCarta:

  Для проверки работы службы sssd с токеном JaCarta-2 SE необходимо дополнительно указать слот при помощи схемы pkcs11:<URI>, которая поддерживается в sssd.
  С полным перечнем параметров pkcs11:<URI> можно ознакомиться в RFC7512 (*https://datatracker.ietf.org/doc/html/rfc7512#section-2.1*).

Создать символическую ссылку в каталоге */usr/lib64/pkcs11/* на библиотеку *libjckt2.so* (чтобы не было ошибок с первым слотом JaCarta GOST 2.0; исправлено с версии 2.7.4-alt6 пакета libjcpkcs11):

```
cd /usr/lib64/pkcs11/
ln -s ../libjckt2.so
/usr/libexec/sssd/p11_child  --ca_db=/etc/certs/ca.cer  --pre  --uri=pkcs11:token=jacarta-
slot-laser
  jacarta-slot-laser
  /usr/lib64/pkcs11/libjcPKCS11-2.so
  6C652D52545F4A435F45535F323034382D393335616430393932D653866662D34632D32373731316le-
RT_JC_ES_2048-935ad093-e8ff-4c-
27711MIIGozCCBYugAwIBAgITJwAAAA4vyUvSfioOvAAAAAAADjANBgkqhkiG9w0BAQsFADBJMRMwEQYKCZImiZPP2
6XcAxrmGfjfT53PYqC30uk+W7S7Q=
```

Если сертификат не найден, посмотреть диагностику, указав дополнительные параметры:

```
--debug-level=10 --logger=stderr
```

Получить идентификатор токена:

```
sudo modutil -dbdir /etc/pki/nssdb -add "Rutoken PKCS11" -libfile librtpkcs11ecp.so
sudo modutil -dbdir /etc/pki/nssdb -list
```

Для настройки службы SSSD необходимо отредактировать файл ***/etc/sssd/sssd.conf*** командой:

```
sudo pluma /etc/sssd/sssd.conf
```

В секцию [pam] добавить строки:

```
pam_cert_auth = True
pam_p11_allowed_services = +mate-screensaver, +lightdm, +kf5-screenlocker, +sddm
pam_cert_db_path = /etc/certs/ca.cer
```

где:

pam_cert_db_path – название вашего сертификата.

Если на токене несколько апплетов, нужно указать с каким именно работать. Для этого необходимо прописать p11_uri токена:

```
p11_uri
= pkcs11:token=NSS%20Generic%20Crypto%20Services;manufacturer=Mozilla%20Foundation;serial=
0000000000000000;model=NSS%203
```

Так как аутентификация по токену занимает больше времени, чем по паролю (поиск сертификата на токене, проверка сертификата по протоколу OCSP и т.п.) необходимо увеличить таймауты для Kerberos и для p11_child, добавив следующие параметры:

```
[domain/test.alt]
...
krb5_auth_timeout = 60
[pam]
...
p11_child_timeout = 60
```

После изменения конфигурации необходимо перезапустить службу SSSD, очистив при этом кэш sssd:

```
systemctl stop sssd && sss_cache -E && systemctl start sssd
```

Проверить доступность службы SSSD для библиотеки:

```
p11-kit list-modules
p11-kit-trust: p11-kit-trust.so
    library-description: PKCS#11 Kit Trust Module
    library-manufacturer: PKCS#11 Kit
    library-version: 0.23
    token: System Trust
        manufacturer: PKCS#11 Kit
        model: p11-kit-trust
        serial-number: 1
```

```
        hardware-version: 0.23
        flags:
               write-protected
               token-initialized
    token: Default Trust
        manufacturer: PKCS#11 Kit
        model: p11-kit-trust
        serial-number: 1
        hardware-version: 0.23
        flags:
               write-protected
               token-initialized
isbc: libisbc_pkcs11_main.so
    library-description:
    library-manufacturer: ISBC
    library-version: 1.0
jcpkcs11: libjcPKCS11-2.so
    library-description: JaCarta PKCS#11 module
    library-manufacturer: Aladdin R.D.
    library-version: 2.4
rutokenecp: librtpkcs11ecp.so
    library-description: Rutoken ECP PKCS #11 library
    library-manufacturer: Aktiv Co.
    library-version: 2.3
```

Если после настройки входа по сертификату при входе пользователя видим ошибку ***generic failure: GSSAPI Error: Unspecified GSS failure. Minor code may provide more information (Server not found in Kerberos database)*** или ***Нет связи с сервером*** необходимо добавить в */etc/hosts* домен:

```
10.72.9.2 s09dc.pki.local pki.local
```

Перечень сокращений

CA	–	Certification Authority (удостоверяющий центр)
НСД	–	Несанкционированный доступ
ОС	–	Операционная система
СЗИ	–	Средство защиты информации