Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Уральский федеральный университет имени первого Президента России Б.Н. Ельцина» (УрФУ)

Институт радиоэлектроники и информационных технологий – РтФ

Школа профессионального и академического образования

Отчет по дисциплине   
«Криптографические методы защиты информации»

Лабораторная работа №1  
«Создание ключей с помощью Kleopatra. Создание корневого и заверенного сертификатов. Обмен зашифрованными данными.»

Студенты: Клоченко И.Е.

Преподаватель: Соколов И.П.

Группа: РИ-300024

Екатеринбург

2023

**Оглавление**

[1. Установка *GPG*4*Win* 3](#_Toc133265985)

[2. Создание ключей 3](#_Toc133265986)

[3. Шифрование файлов 7](#_Toc133265987)

[4. Обмен зашифрованными файлами с напарником 9](#_Toc133265988)

[5. Обмен ключами 10](#_Toc133265989)

[6. Обмен зашифрованными данными 13](#_Toc133265990)

[Ответы на контрольные вопросы 16](#_Toc133265991)

[Вывод 21](#_Toc133265992)

**Цель работы:**

1. Ознакомиться со сведениями о программе *Kleopatra*
2. Ознакомиться с составом программных средств, входящих в систему *Kleopatra*, изучить справку о системе.
3. Создать криптографические ключи с помощью программы. Изучить порядок создания ключей шифрования в системе *Kleopatra*. Создание корневого и заверенного сертификатов.
4. Изучить способы шифрования и дешифрования файлов с помощью программы *Kleopatra*. Узнать, как происходит обмен ключами.

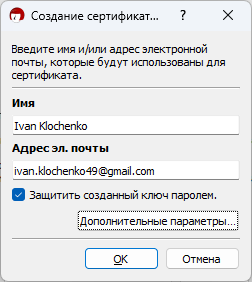
**Ход работы:**

## Установка *GPG*4*Win*

Устанавливаем ПО *GPG*4*Win* 4.1.0.

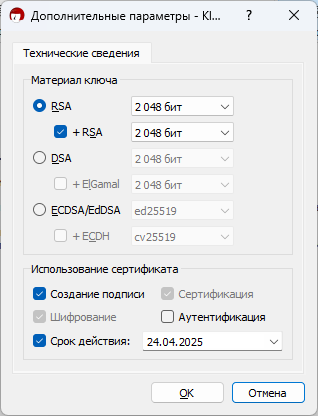
## Создание ключей

* 1. Запускаем утилиту *Kleopatra*.*exe*. Открывается окно управления ключами.
  2. Выбираем пункт создания обычной пары. Открывается окно создания сертификата. Вводим имя пользователя и адрес электронной почты в отведенные под них формы.



**Рисунок 1**. Создание сертификата

Переходим в дополнительные параметры.



**Рисунок 2**. Дополнительные параметры создания пары ключей

Здесь можно выбрать тип алгоритма шифрования, настроить использование сертификата, например создать подпись, включить идентификацию, выставить дату действительности сертификата.

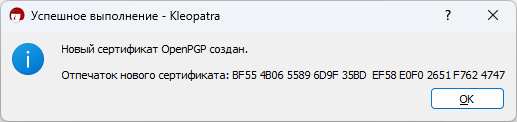
Остановимся на алгоритмах. У них есть отличия: они используют разные алгоритмические подходы для шифрования, как ни странно. *RSA* – использование двух простых больших чисел. *DSA* использует дискретные алгоритмы. *ECDSA*/*EdDSA* использует алгоритм шифрования, основанных на эллиптических кривых; из предложенных алгоритмов считается более современным и безопасным чем его «коллеги».

У каждого алгоритма можно выставить длину ключей в байтах. Выставим 2048, как указано в методических указаниях.

Кроме того, можно установить гибридную схему шифрования (чек-боксы под основными алгоритмами) для сочетания ассиметричного и симметричного шифрований, что в свою очередь обеспечит как безопасность, так и скорость работы.

* 1. После подтверждения нас встречает окно ввода пароля (указали дополнительным параметром при создании сертификата).

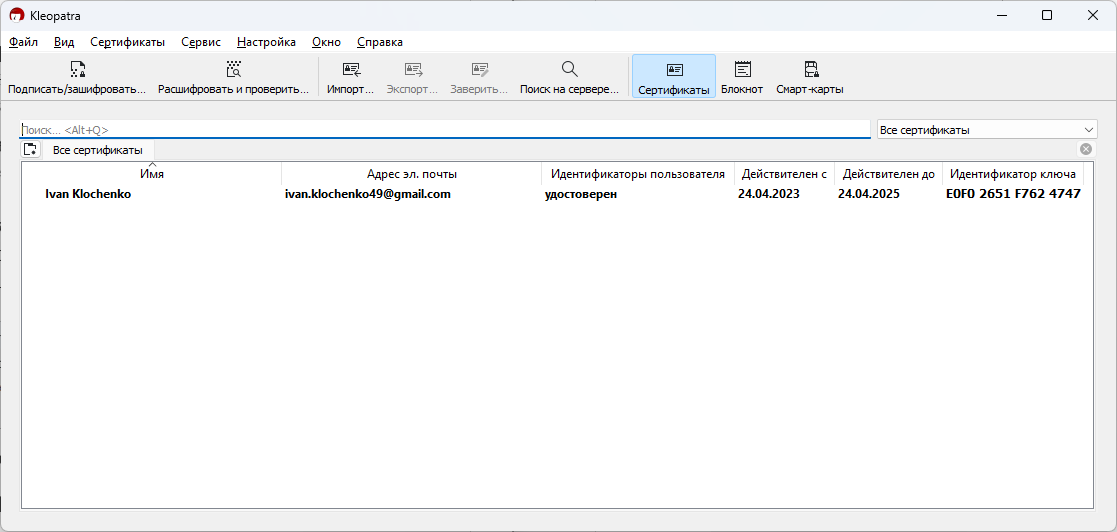
Вводим и подтверждаем повторным вводом фразу-пароль. Нажимаем «ОК». Появляется диалоговое окно с процессом создания сертификата и после этого появляется новое диалоговое окно с успешным завершением процесса создания сертификата.



**Рисунок 3**. Успешное выполнение – Kleopatra

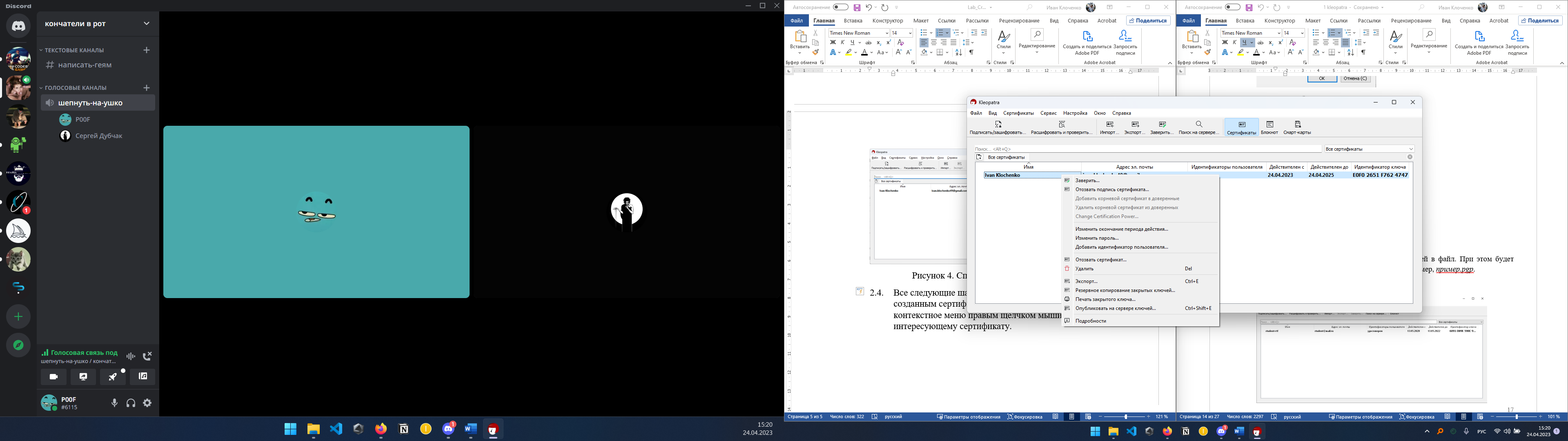
Если не устанавливать пароль на закрытый ключ, кто угодно может, получив доступ к рабочему месту, получить все возможности, которые открыты при владении закрытым ключом: дешифрование, шифрование, подпись файлов, что, безусловно, небезопасно.

За этим окном следует другое – в котором мы можем увидеть наш созданный сертификат.



**Рисунок 4**. Список сертификатов

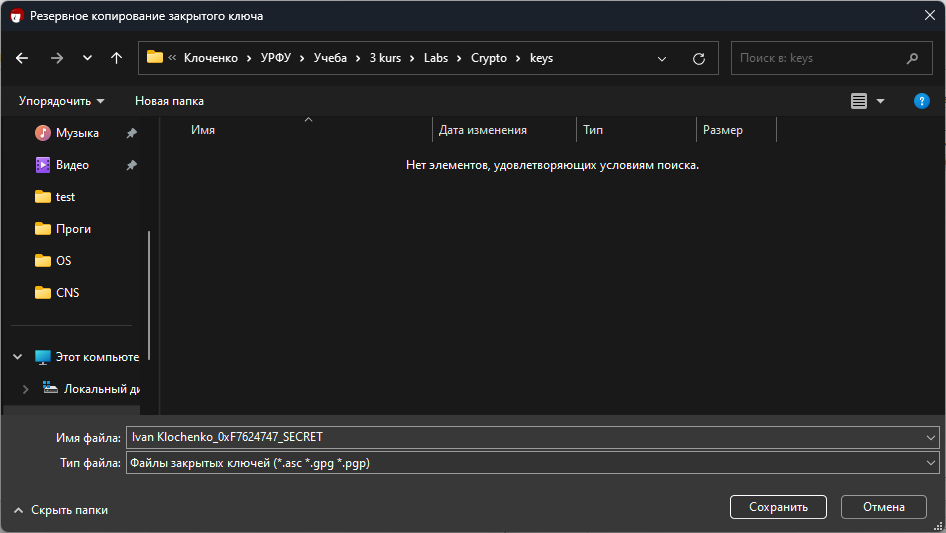
* 1. Все следующие шаги, которые можно выполнить с созданным сертификатом, можно увидеть, вызвав контекстное меню правым щелчком мыши по интересующему сертификату.



**Рисунок 5**. Возможности работы с сертификатом

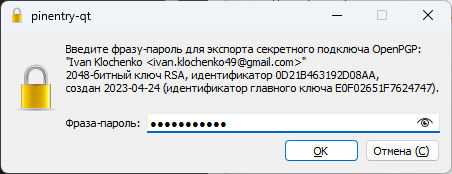
* 1. Попробуем сохранить резервную копию ключей в файл.

Нас встречает окно с выбором места сохранения ключей и указанием имени файла, в который будут сохранена резервная копия ключей.



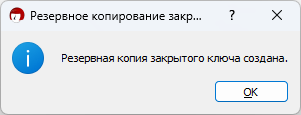
**Рисунок 6**. Сохранение резервной копии ключей в файл

Далее нас встречает диалоговое окно, которое просит ввести фразу-пароль для продолжения:



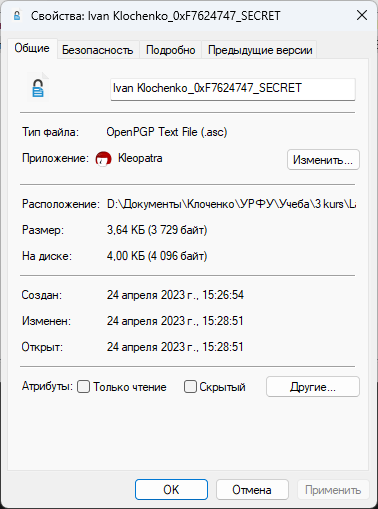
**Рисунок 7**. Подтверждение паролем

После этого встречаем окно с сообщением об успехе сохранения резервной копии ключей в файл.



**Рисунок 8**. Успешное создание резервной копии закрытого ключа

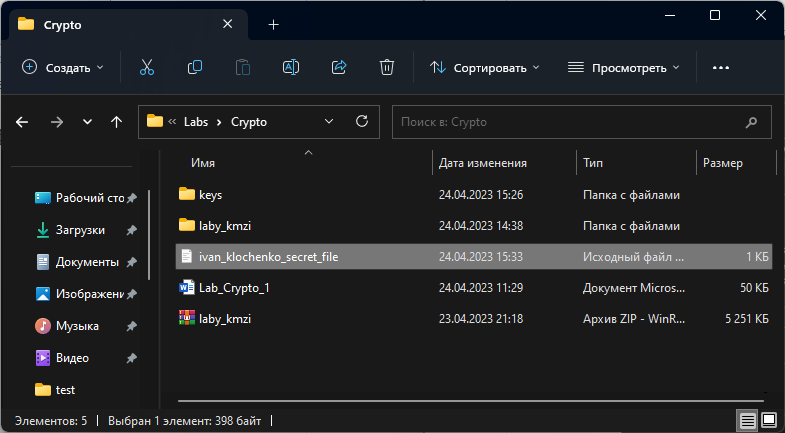
Удостоверимся в создании резервной копии. Был создан файл с расширением закрытого ключа .*asc*.



**Рисунок 9**. Файл резервной копии ключа

## Шифрование файлов

* 1. Выберем файл для зашифровки.

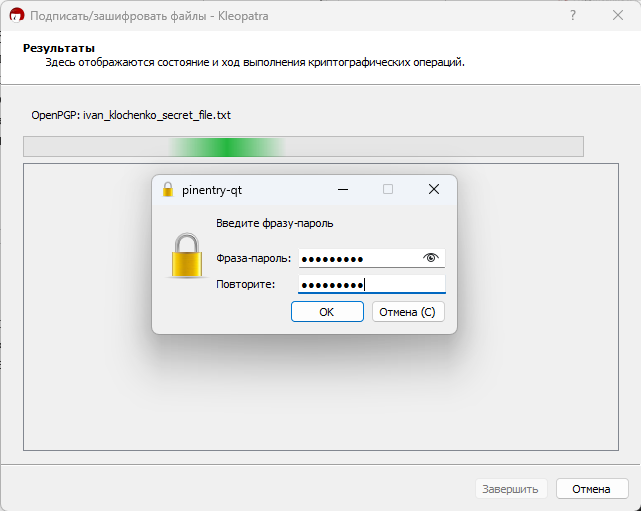


**Рисунок 10**. Выбор файла для зашифровки

Правой кнопкой мыши по файлу вызываем контекстное меню и выбираем «Подписать и Зашифровать».

* 1. Изучим параметры шифрования. Если не выбрать пункт шифрования с паролем, то шифрование и дешифрование будет производиться без контрольной фразы-пароля, что означает следующее: если вы, например, каким-то образом заполучили сертификат лица, выполнившего шифрование файла, то вы можете без проблем расшифровать этот файл, ссылаясь лишь на доверенный сертификат. Если же во время шифрования файл будет установлена контрольная фраза-пароль, то каждый раз при дешифровании файла, она будет запрашиваться, что повышает степень безопасности файла и информации в нем. Использование фразы-пароля отражено в журнале аудита.

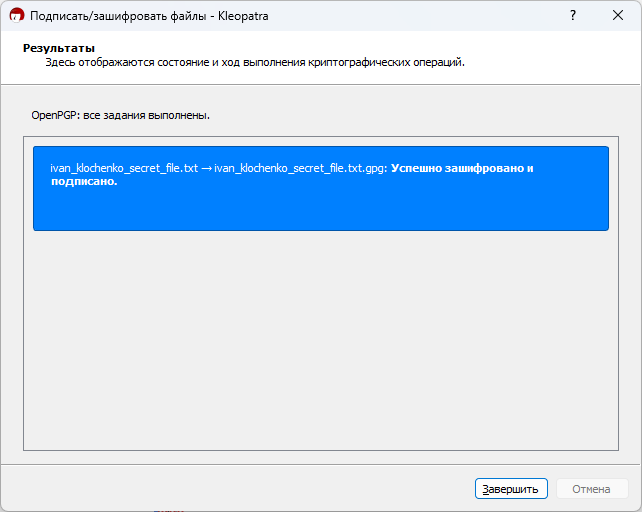
Зашифруем файл, используя пароль. Сначала утилита попросит нас ввести фразу-пароль (опциональный пароль, который мы указали на предыдущем шаге).



**Рисунок 11**. Ввод фразы-пароля для шифрования файла

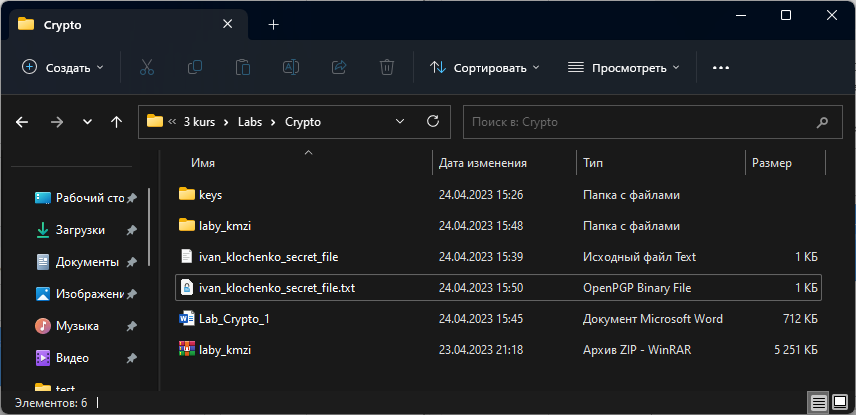
Далее утилита просит ввести фразу пароль для сертификата.

После успешного подтверждения следует окно, сообщающее нам, что файл успешно подписан и зашифрован, и что у зашифрованной и подписанной версии файла новый тип – *OpenPGP* *Binary* *File*.



**Рисунок 12**. Успешные подпись и шифрование файла

Нажимаем «Завершить» и проверяем.



**Рисунок 13**. Зашифрованный и подписанный файл

## Обмен зашифрованными файлами с напарником

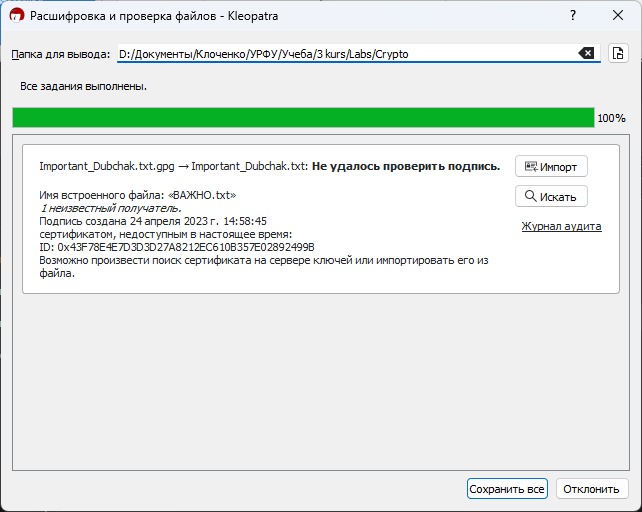
В качестве напарника был выбран одногруппник Дубчак Сергей.

* 1. Обменяемся зашифрованным файлом с напарником.
  2. Пробуем расшифровать полученный от напарника файл. Важно отметить, что во время шифрования отправитель (я) и получатель (Сергей) шифровали свои файлы с паролем.

Файлы успешно открылись расшифровались только после обмена фразами-паролями. Вывод: при перехвате фразы-пароля, которая была установлена при шифровании файла, файл может быть успешно расшифрован злоумышленником.

Если же пароль не сообщать, расшифровка файла невозможна.

Во время расшифровки файла, было получено следующее сообщение:

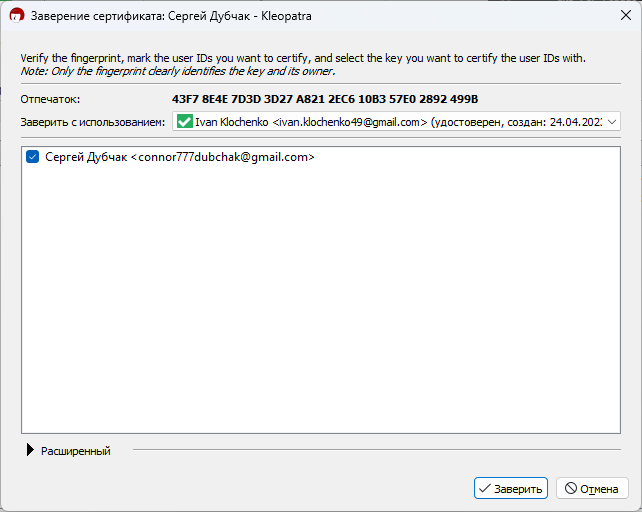


**Рисунок 14**. Сообщение об отсутствии сертификата

Это сообщение было получено в силу отсутствия заверенного сертификата (публичного ключа) отправителя.

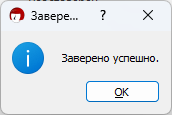
## Обмен ключами

* 1. Обмениваемся с напарником публичными ключами (файл с публичным ключом имеет суффикс *\_public*.*asc*).
  2. Добавляем открытый ключ напарника список сертификатов. После импорта сертификата, программа предлагает его заверить (изначально сертификат импортируется с не удостоверенным идентификатором пользователя).



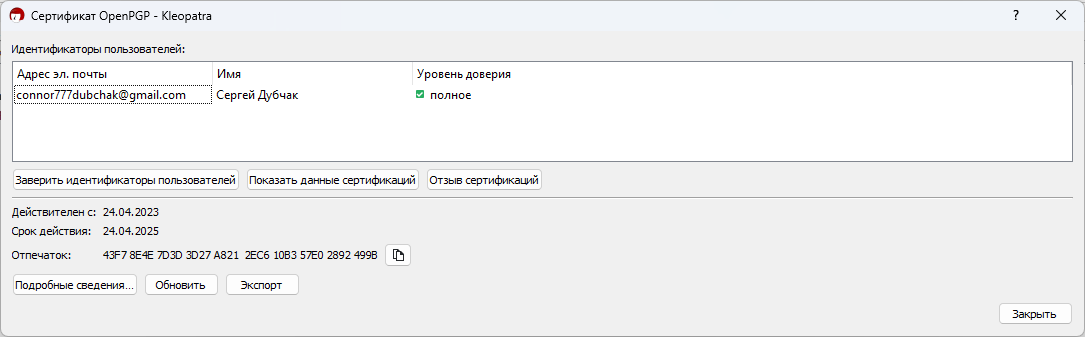
**Рисунок 15**. Заверение сертификата напарника

Нажимаем «Заверить» и после этого нас встречает форма с подтверждением фразы-пароля для разблокировки нашего секретного ключа *OpenPGP*. Вводим пароль и получаем диалоговое окно с сообщением об успешном заверении сертификата.



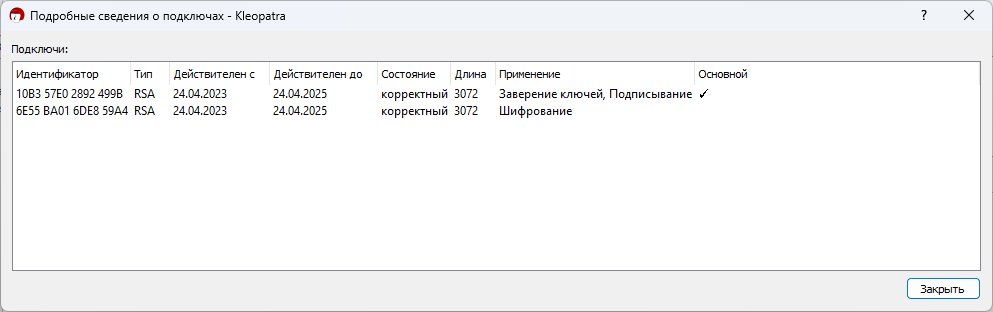
**Рисунок 16**. Успешное заверение сертификата

* 1. Дважды кликаем по импортированному сертификату пользователя чтобы посмотреть подробности. Уровень доверия у сертификата: полное.



**Рисунок 17**. Подробности импортированного сертификата

Переходим в «Подробные сведения о подключах». Здесь можно получить информацию о идентификаторе, типе, времени действительности, состоянии, длине, применении, а также, является ли ключ основным.



**Рисунок 18**. Подробные сведения о подключах

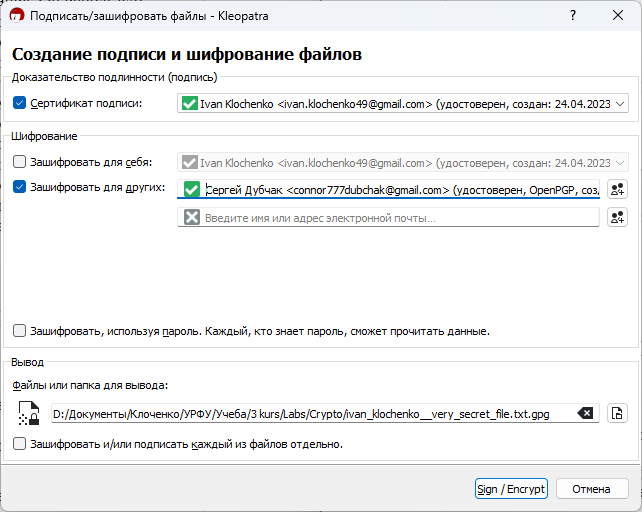
Тип ключа – *RSA*, он обозначает алгоритм шифрования и подписи файла. Это ассиметричный алгоритм, для работы которого необходимы открытый и закрытый ключи. Также в одной из колонок указана длина ключа – 3072 бит. Стоит отметить, что, чем больше длина ключа *RSA*, тем сложнее будет вычисление его факторизации или, иными словами, его будет сложнее взломать.

Преимущества алгоритма *RSA*. Как уже было отмечено выше, *RSA* – это ассиметричный алгоритм шифрования. Он использует открытый и закрытый ключи для шифрования и дешифрования данных или для создания и проверки цифровой подписи. Вторым преимуществом является сложность факторизации больших целых чисел. Сложность факторизации растет экспоненциально при увеличении длины ключа. Третьим преимуществом выступает универсальность данного алгоритма. Его можно использовать для шифрования данных, создания цифровой подписи, аутентификации и обмена ключами. Четвертым преимуществом выступает скорость алгоритма. Это один из самых быстрых ассиметричных алгоритмов шифрования, что очень полезно при требованиях быстрой обработки данных.

Пятым преимуществом можно назвать широкую поддержку алгоритма различными криптографическими библиотеками и ПО. *RSA* – это доступный и удобный алгоритм шифрования.

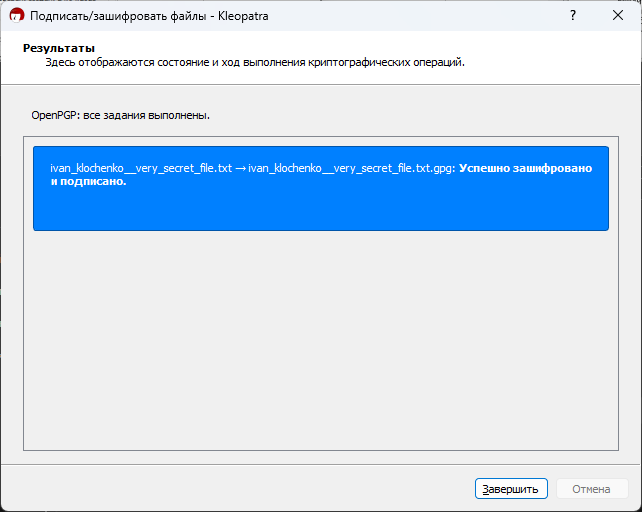
## Обмен зашифрованными данными

* 1. Создадим новый текстовый файл с произвольным названием и произвольным содержанием.
  2. Используем для сохранения кодировку *UTF*-8. Если использовать другую, может поменяться отображение содержимого при дешифровании файла.
  3. Подпишем и зашифруем файл.
  4. Во время шифрования уберем галочку «Шифровать для себя», поставим галочку «Шифровать для других» и отметим напарника в качестве лица, для которого мы будем шифровать файл.



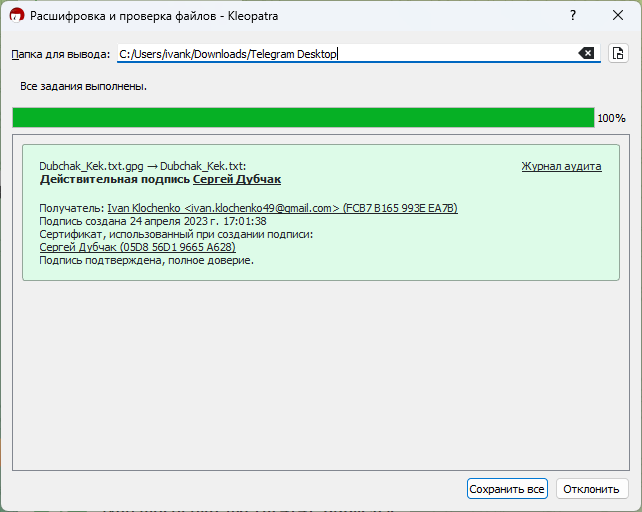
**Рисунок 19**. Шифрование файла для напарника

Запускаем процедуру шифрования и подписи файла, игнорируя предупреждение о том, что мы не сможем расшифровать данные. Вводим фразу пароль для разблокировки секретного ключа *OpenPGP*. Проверяем:



**Рисунок 20**. Шифрование файла для напарника

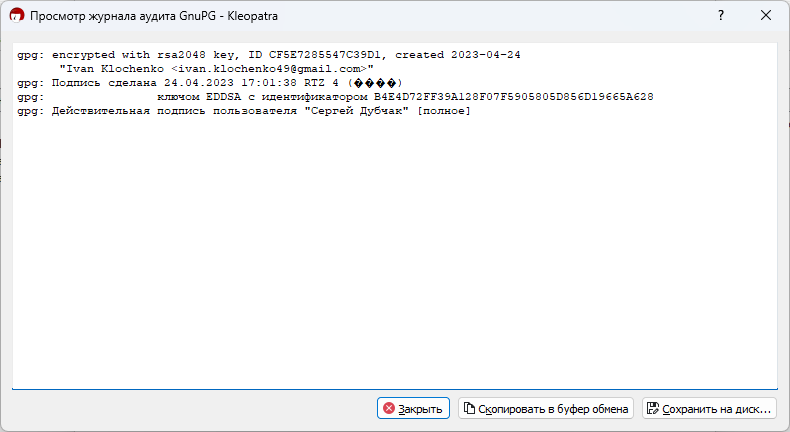
* 1. Отправляем файл напарнику в обмен на аналогичный файл от него. Нажимаем «Расшифровать и проверить файл». Вводим фразу-пароль, с помощью которого был зашифрован исходный файл. Получаем следующее оповещение.



**Рисунок 21**. Успешное дешифрование и проверка подписи файла напарника.

Стоит отметить, что если превысить количество установленных попыток ввода фразы-пароля для расшифровки файла (3 попытки), будет выведено сообщение «*Bad* *passphrase*», после чего придется пробовать расшифровать снова и у вас снова будет 3 попытки.

* 1. Проверим журнал аудита.



**Рисунок 22**. Журнал аудита после дешифрования файла

В этом журнале содержится информация о типе шифрования, длине ключа шифрования, идентификаторе ключа, дате шифрования и имени владельца. Также имеется информация о подписи файла, ключе подписи, идентификаторе подписи и владельце подписи с указанием уровня доверия.

Опасность перехвата кодов идентификаторов ключей заключается в идентификации и аутентификации пользователей при шифровании или дешифровании сообщений, а также проверке подписей сообщений злоумышленником. Если злоумышленник получит эти коды, он может использовать их для подделки сообщений, подписывания сообщений от имени других пользователей, а также для расшифровки сообщений, зашифрованными ключами, коды идентификаторов которых были перехвачены. Следовательно, злоумышленник может получить доступ к конфиденциальной информации, которая была зашифрована с помощью этих ключей.

## Ответы на контрольные вопросы

* 1. В: Для чего предназначена программа *Gpg*4*win* и *Kleopatra*?

О: *Gpg*4*Win* – набор криптографических инструментов для защиты конфиденциальной информации в ОС *Windows*. Включает в себя *OpenPGP* – стандарт протокола шифрования, который обеспечивает конфиденциальность, путем аутентификации и цифровой подписи данных.

*Kleopatra* – графический интерфейс для *Gpg*4*Win*, который позволяет подписывать и шифровать файлы, а также, создавать, хранить и управлять сертификатами и ключами шифрования.

* 1. В: Как с помощью программы *Kleopatra* создать ключи?

О: Чтобы создать ключи с помощью утилиты *Kleopatra*, нужно воспользоваться графическим интерфейсом утилиты: выбрать «Файл – Новая пара ключей *OpenPGP* …»; ввести имя и адрес электронной почты; (опционально) включить защиту паролем (на следующем этапе ввести и подтвердить пароль); (опционально) в дополнительных параметрах выбрать алгоритм шифрования, схему шифрования (обычное шифрование или гибридное шифрование) и длины ключей шифрования; выставить при необходимости аутентификацию.

* 1. В: Для чего применяются открытый (публичный) и закрытый (секретный) ключи?

О: Открытый, или публичный, ключ используется для шифрования данных сообщения; может быть известен получателю или всем. Закрытый, или секретный, ключ используется для дешифрования данных сообщения; известен только владельцу.

* 1. В: Как и для чего осуществляется шифрование файлов?

О: Шифрование файлов осуществляется при помощи программного обеспечения или иного метода шифрования, в основе которых лежат алгоритмы шифрования, например *RSA*, *DES*, *AES*, *DSA*.

Шифрование осуществляется по ряду причин: защита конфиденциальных документов, личная безопасность, государственная безопасность, безопасность переписки и т.д.

* 1. В: Как изменить свой пароль? В каких случаях это необходимо?

О: Пароль (если здесь имеется ввиду пароль созданного сертификата – пары ключей) можно изменить, вызвав для нужного сертификата контекстное меню и выбрав «Изменить пароль». После этого нужно будет ввести предыдущий пароль, а затем ввести и подтвердить новый.

Изменение пароля необходимо в случаях, когда, например, текущий пароль был скомпрометирован, и, чтобы конфиденциальная информация осталась таковой, пароль следует изменить.

* 1. В: Назовите способы шифрования текстовых файлов с помощью *Gpg*4*win*.

О: Есть несколько способов шифрования с помощью *Gpg*4*Win*, укажем некоторые из них:

* Шифрование с помощью симметричного ключа (шифрование через *ZIP* архив с паролем)
* Шифрование с помощью ассиметричного ключа (создание пары ключей – открытый и закрытый – для зашифровки и дешифровки файлов, например, с помощью алгоритма *RSA*)
* Шифрование с помощью расширений для отдельных сервисов, например, использование *GpgOL* для *Microsoft* *Outlook* для шифрования и дешифрования электронных писем.
* Использование командной строки и соответствующего синтаксиса для шифрования и дешифрования файлов.
* Использование плагина *GpgEX* для шифрования файлов Проводника ОС *Windows*.
  1. В: Как проверяется достоверность ключа?

О: При импорте ключа в *Gpg*4*win* программа отображает информацию о ключе, включая его идентификатор и отпечаток. Пользователь, получивший ключ, может проверить эту информацию, связавшись с владельцем ключа по другим каналам связи, чтобы подтвердить достоверность. Кроме того, пользователь может запросить сертификат подтверждения ключа у других пользователей, которые уже подтвердили его достоверность. Этот сертификат может использоваться для установления доверия к ключу. В *Gpg*4*win* используется система *web*-*of*-*trust*, которая позволяет пользователям проверять достоверность ключей на основе доверия, которое они имеют к другим пользователям, которые уже проверили ключ.

* 1. В: Как удалить устаревшие ключи?

О: Если ключ устарел и его необходимо удалить это производится следующим образом: вызывается контекстное меню ключа, который необходимо удалить, выбирается пункт «Удалить»; после этого следуя интуитивно понятным указаниям диалоговых окон, производится удаление пары ключей.

* 1. В: Как дешифровать закрытый файл?

О: Находясь в основном окне утилиты *Kleopatra*, нажимаем «Расшифровать и проверить файлы», выбираем нужный нам файл и вводим фразу-пароль (если она установлена) для файла, после чего утилита дешифрует файл и спросит, куда сохранить дешифрованный файл. Также она обязательно выведет сообщение о действительности подписи отправителя с остальной информацией о зашифрованном файле.

* 1. В: Какова длина ключа, использованного в данной работе?

О: Длина ключа в данной работе: 2048 бит.

* 1. В: Перечислите способы обмена открытыми ключами.

О: Были освоены следующие способы обмена открытыми ключами:

* Путем экспорта ключа в файл при последующей передаче этого файла по каналам связи получателю. Получатель должен импортировать этот ключ в программе.
* Путем передачи текста ключа в сообщении (с учетом «шапки» и «подвала» передаваемого ключа). Получатель должен при получении текста ключа импортировать текст ключа в программе.
  1. В: Что такое идентификатор пользователя?

О: Идентификатор пользователя в *Gpg*4*win* – это уникальный текстовый идентификатор, который используется для связывания открытого ключа с реальным человеком или организацией. В нашей работе это были имя пользователя и его электронная почта. Идентификатор пользователя помогает облегчить удостоверение личности владельца ключа и помогает другим пользователям связываться с ним для установления безопасного обмена сообщениями и (или) файлами.

* 1. В: Какие компоненты включает программа *Gpg*4*win*?

О: Программа *Gpg*4*win* включает в себя следующие компоненты:

* *GnuPG* – ядро *Gpg*4*win*
* *Kleopatra* – графический интерфейс пользователя для работы с открытым и закрытым ключами и создания цифровых подписей
* *GpgOL* – расширение для *Microsoft* *Outlook*
* *GpgEX* – плагин для *Windows* *Explorer* (проводник).
* *Claws* *Mail* – почтовый клиент, который поддерживает шифрование и подпись сообщений
  1. В: Какие алгоритмы шифрования используются в программе *Kleopatra*?

О: В утилите *Kleopatra* используются следующие алгоритмы шифрования:

* + 1. Ассиметричные алгоритмы шифрования:
* *RSA*
* *DSA*
* *ECDSA*/*EdDSA*
  + 1. Симметричные алгоритмы шифрования:
* *AES*
* *Twofish*
* *Blowfish*
* *Camelia*
* *IDEA*
* 3*DES*
* *ElGamal*
* *CAST*5
  + 1. Хэш-функции:
* *MD*5
* *SHA*-1
* *SHA*-2
* *RIPE*-*MD*160
* *TIGER*
  + 1. Алгоритмы сжатия (и шифрования)
* *ZIP*
* *ZLIB*
* *BZIP*2
  + 1. Алгоритмы подписи:
* *DSA*
* *RSA*
* *ECDSA*/*EdDSA*

## Вывод

В ходе лабораторной работы были освоены и практически закреплены навыки шифрования, дешифрования и электронной подписи файлов в программе *Gpg*4*win*, утилите *Kleopatra*.

Создана пара ключей: открытый и закрытый; изучены подробные сведения о них, такие, как идентификатор, длина, алгоритм шифрования.

Проведено шифрование с последующей отправкой напарнику файла алгоритмом *RSA* с длиной ключа 2048 и фразой-паролем.

В ходе того же действия в обмен получен и дешифрован файл напарника.

Произведен обмен открытыми ключами с их последующим заверением.

Произведена отправка файла, который зашифрован для конкретного пользователя в (в ходе работы им выступал напарник) с последующим дешифрованием.

Кроме того, изучен дополнительный способ передачи открытых ключей – передача текста ключа в сообщении с его последующим импортом.

Закреплены знания об алгоритмах шифрования: *RSA*, *DSA*, *ECDSA*/*EdDSA*.

Даны ответы на контрольные вопросы в конце лабораторной работы.