Методические указания по выполнению практической работы "Защита информации в электронных документах путем шифрования и формирования электронной подписи" (для пользователей Linux)

Для создания и использования электронной подписи мы будем использовать приложение gpa.

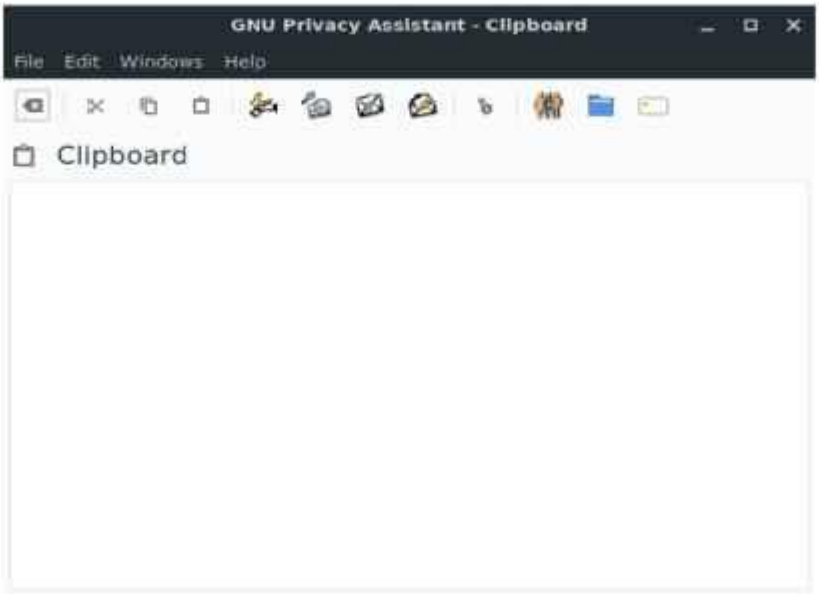

Рис 1. Окно приложения gpa

Для его установки, необходимо в терминале ввести команду:

sudo apt-get install gpa

Запустите установленное приложение.
Нажмите кнопку “Open the keyring editor” 

Появится окно для работы с цифровыми ключами шифрования

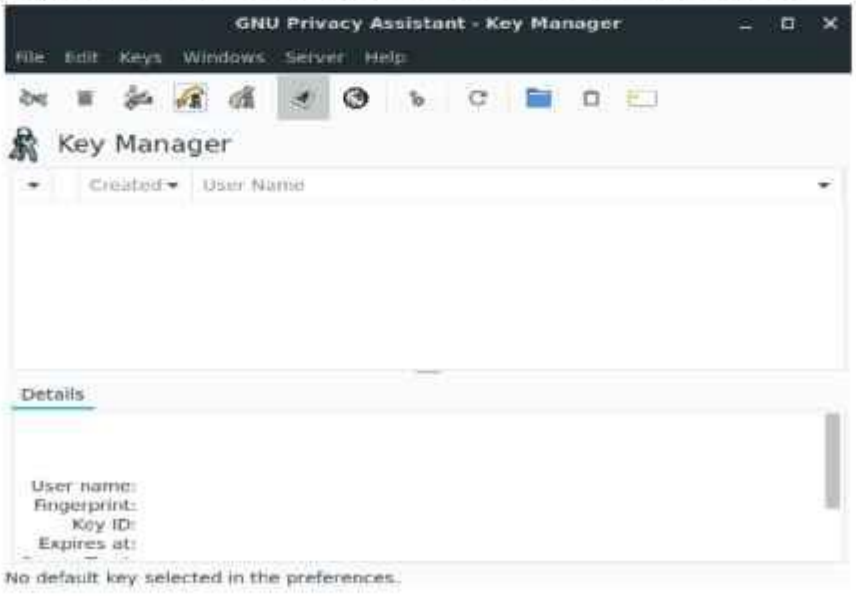

Рис 2. Окно для работы с ключами шифрования

Нажмите кнопку Keys – New key
Появится генератор ключей

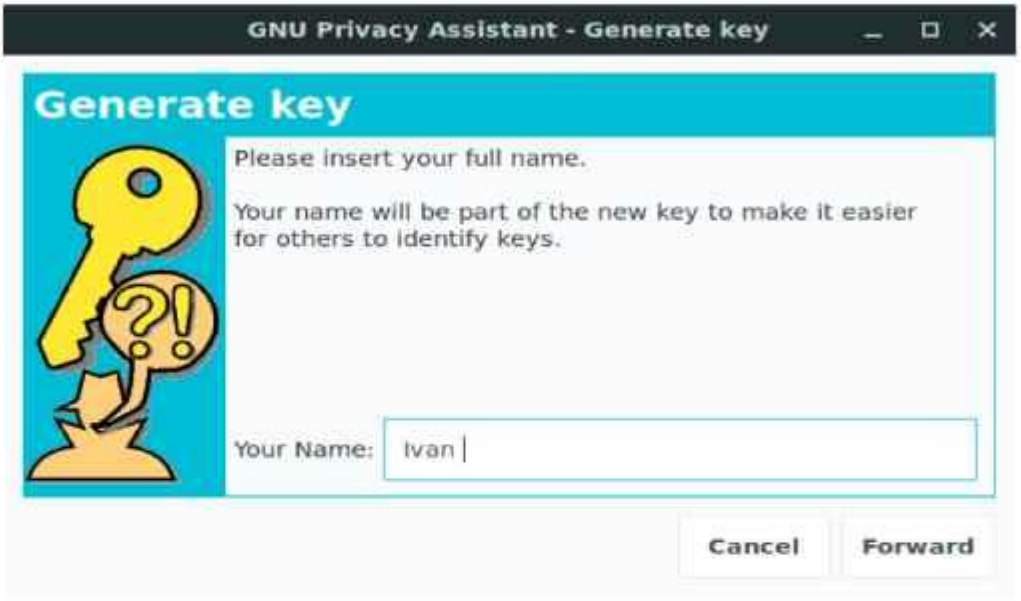

Рис 3. Окно генератора ключей

Введите своё имя (или то, что хотите выдать за него)

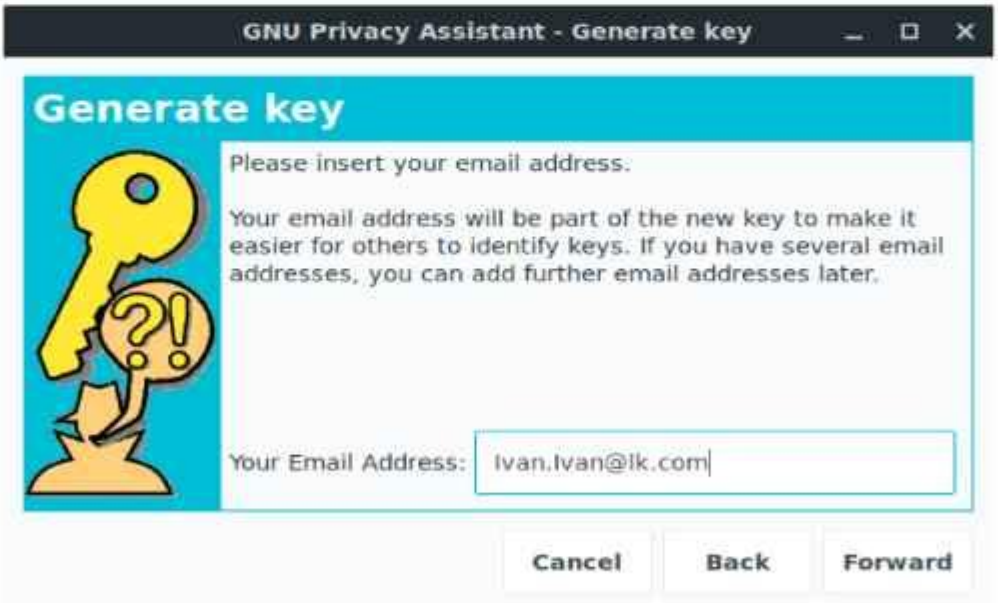

Рис 4. Окно ввода

Введите свой Email (или опять же то, что хотите выдать за него)

Рис 5. Окно с предложением сделать резервную копию ключа

Далее появится окно которое предложит ввести фразу-пароль для защиты ключа. Введите фразу-пароль и подтвердите её. После этого можно будет сохранить файл ключа в произвольном месте на компьютере.

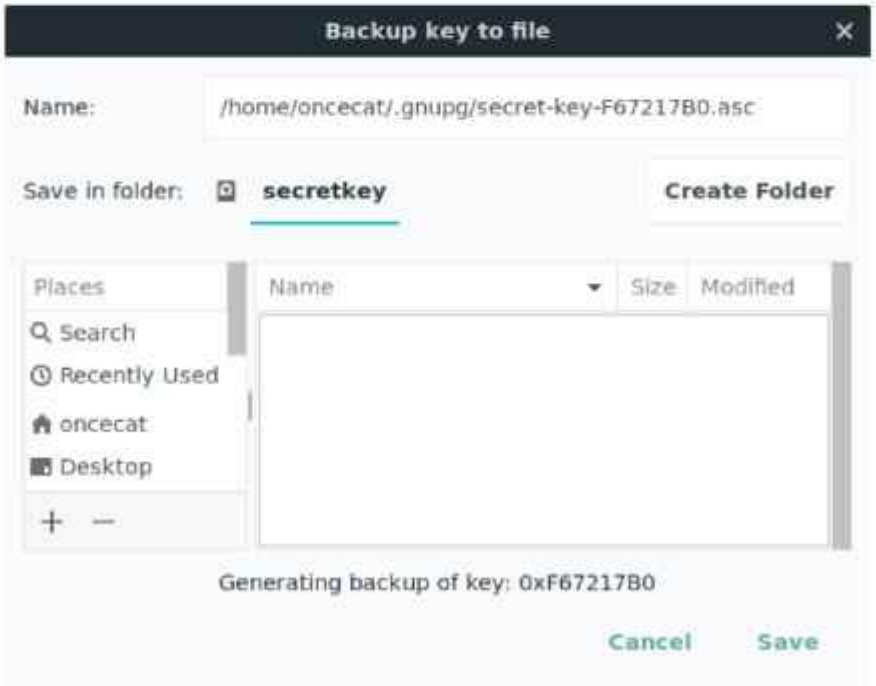

Рис 6. Окно сохранения файла ключа

Рис 7. Окно об успешном создании ключа

После этого, в окне работы с ключами появится строчка содержащая введёные вами данные

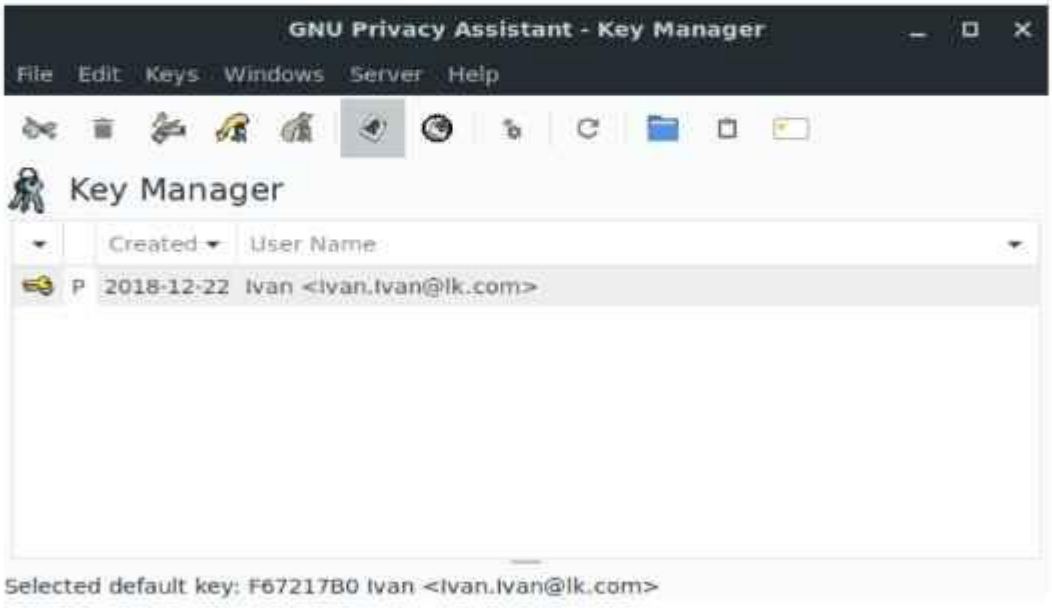

Рис 8. Отображение сертификата после создания пары ключей

Для того, чтобы осуществить шифрование и подпись файлов для передачи другим лицам, необходимо обменяться открытыми ключами . Для этого

необходимо передать файл-ключ по какому-либо каналу связи (эл.почта, соцсети, файлообменники и тд).

Данный файл можно отправить другому лицу, чтобы:
* Вам могли отправить файл, зашифрованный по вашему открытому ключу
* Получатель имел возможность проверить вашу электронную подпись

Импорт стороннего открытого ключа (сертификата)

Допустим, вы получили по какому-либо каналу связи файл, содержащий открытый ключ другого лица. Для того, чтобы его можно было использовать для шифрования файлов или проверки электронной подписи – необходимо импортировать его в программу gpa.

Для этого нажмите кнопку Import keys 

Выберите файл содержащий открытый ключ другого лица и откройте его

Появится окно уведомляющее об успехе импорта

Рис 9. Уведомление об импорте (содержимое может отличаться)
Теперь импортированный ключ появился в данном окне

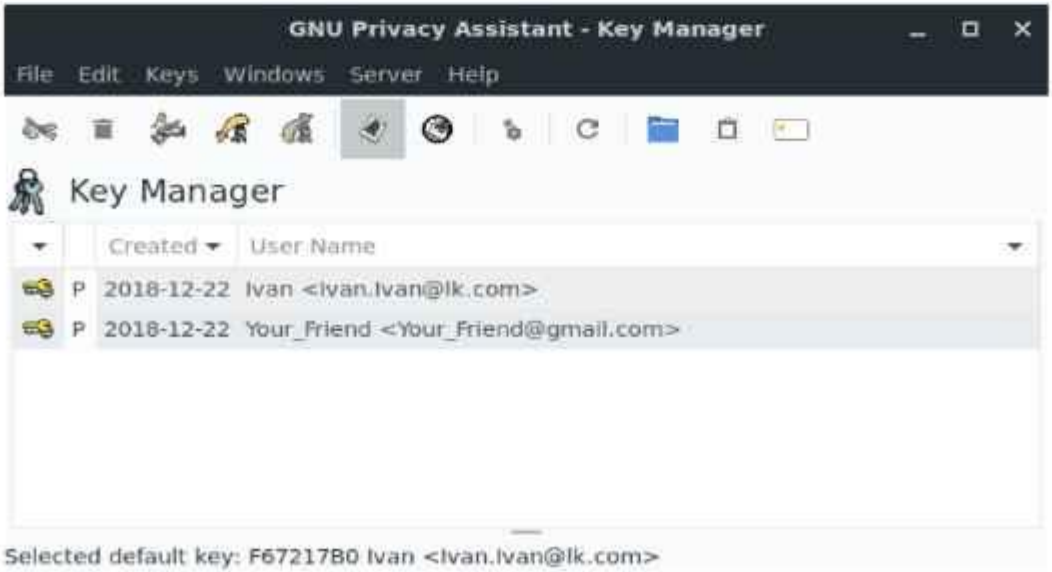

Рис 10. Результат импорта ключа

Теперь, чтобы расшифровать чужой файл, необходимо нажать кнопку 

Появится окно, в котором можно выбрать файл для расшифровки. Сделать это можно выбрав File-Open

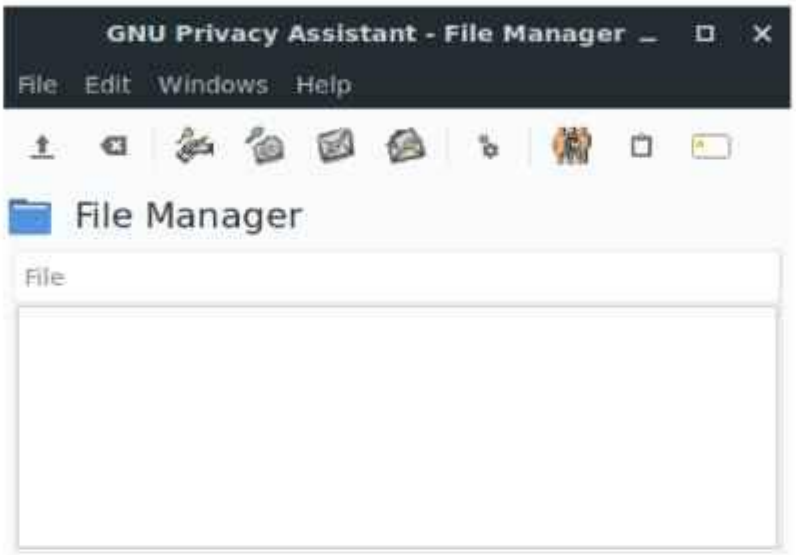

Рис 11. Окно файлового менеджера gpa

В следующем окне выбираем зашифрованный файл (он будет иметь двойное расширение) в данном примере, был зашифрован документ с расширением docx. В зашифрованном виде расширение у нашего файла будет .docx.asc

Рис 12. Выбран файл для дешифровки

Для расшифровки файла нужно нажать File-Decrypt , после этого программа попросит вас ввести фразу-пароль. В случае успеха вы получите такое окно, оно сообщает что файл расшифрован и электронная подпись верна.

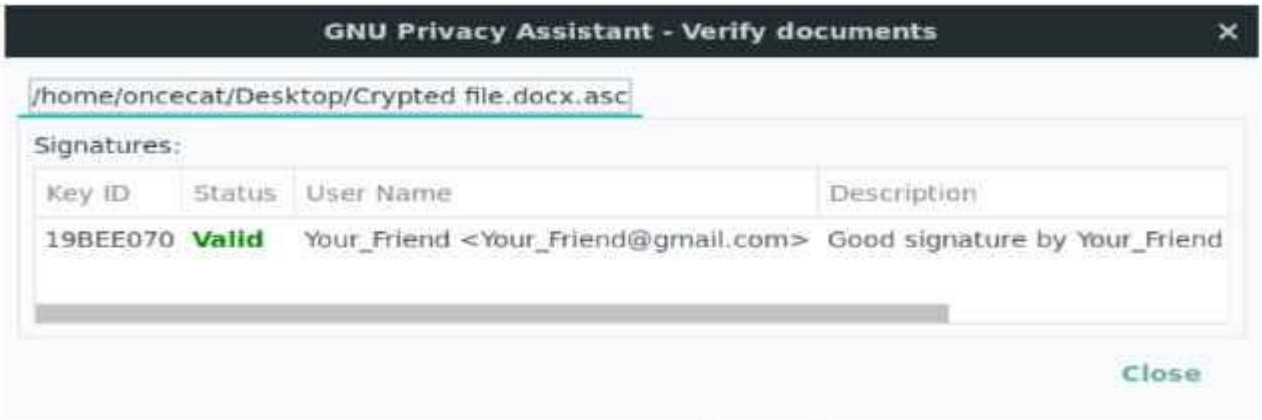

Рис 13. Успешная расшифровка файла

В случае введения неверного пароля – у вас будет еще 2 попытки, если вы провалите и их – окно программы закроется.

Теперь у нас есть расшифрованный файл полученный от другого человека.

В случае если мы хотим отправить зашифрованный или подписанный файл, нужно выполнить следующее:

В главном окне программы открываем файловый менеджер 

Выбираем файл который необходимо зашифровать с помощью File-Open Затем, File-Encrypt , появится следующее окно

Рис 14. Окно шифрования/подписи файла

Разберём содержимое этого окна:

Галочка “Sign” позволит подписать файл выбранной электронной подписью
Галочка “Armor” позволяет зашифровать файл фразой-паролем

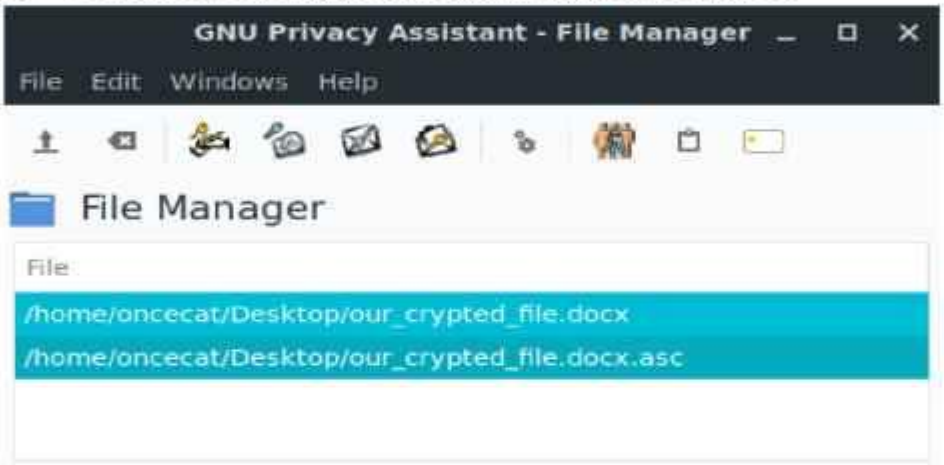

Рис 15. Результат шифрования/подписи файла

Теперь можно отправлять зашифрованный файл и фразу-пароль .

Для подписи файла без шифрования , необходимо выбрать File-Sign

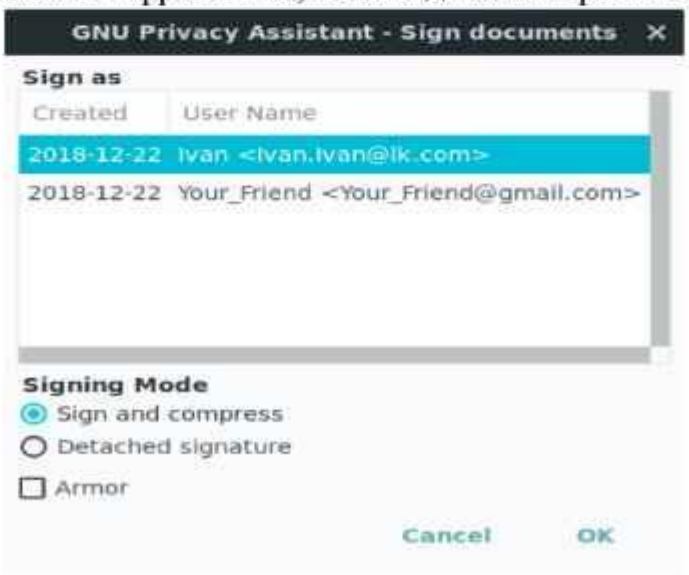

Рис 16. Окно подписи файла

Рассмотрим режимы подписи:
Sign and compress – позволит подписать и сжать файл (будет создан файл с расширениями .docx.gpg)
Detached signature – отдельная подпись (будет создан файл с расширениями .docx.sig)
Armor – позволит дополнительно защитить ваш файл ASCII кодом (это просто способ преобразования необработанных двоичных данных использующий ограниченные символы ASCII)

Подписанные файлы появятся в списке файлового менеджера программы

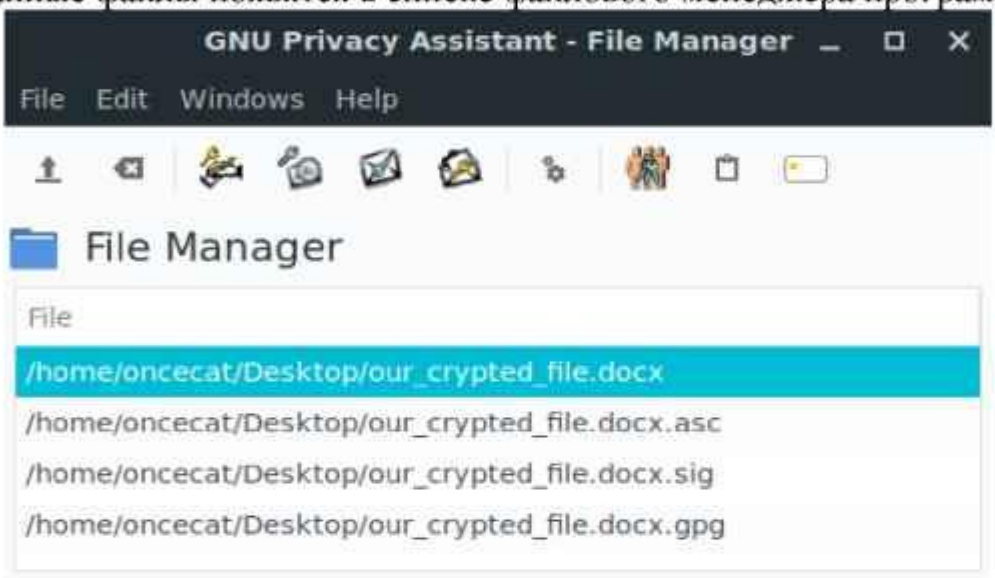

Рис 17. Окно с файлами подписанными разными способами

Для проверки электронной подписи на файле пометим щелчком мыши нужный файл и затем выберем File-Verify .
В случае успеха вы увидете следующее окно:

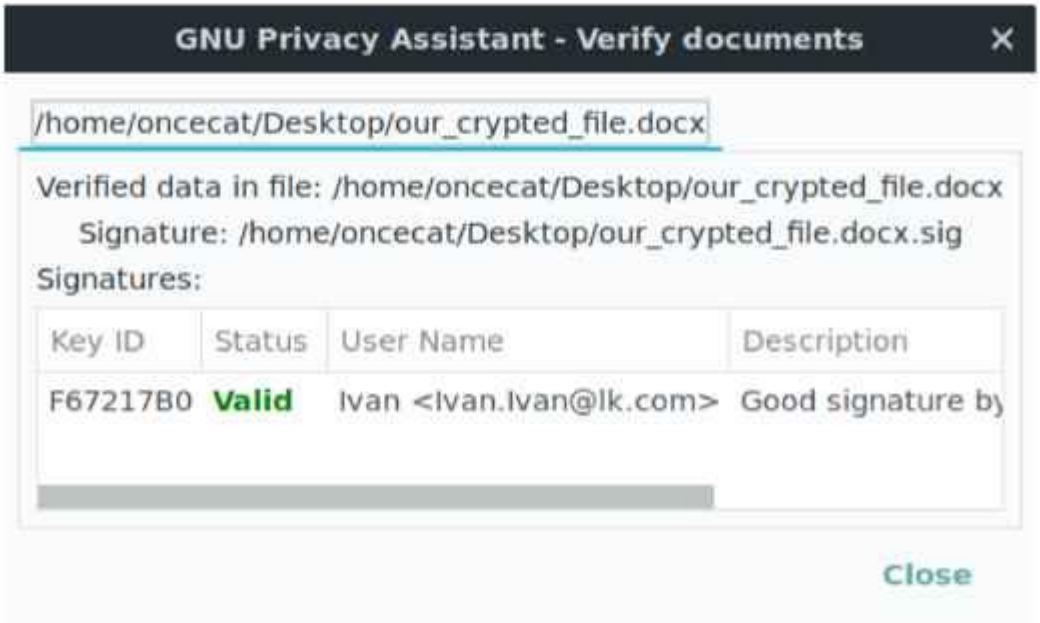

Рис 18. Успешная проверка подписанного файла

Если файл был хоть на символ изменён – приложение оповестит о том, что подпись неверна

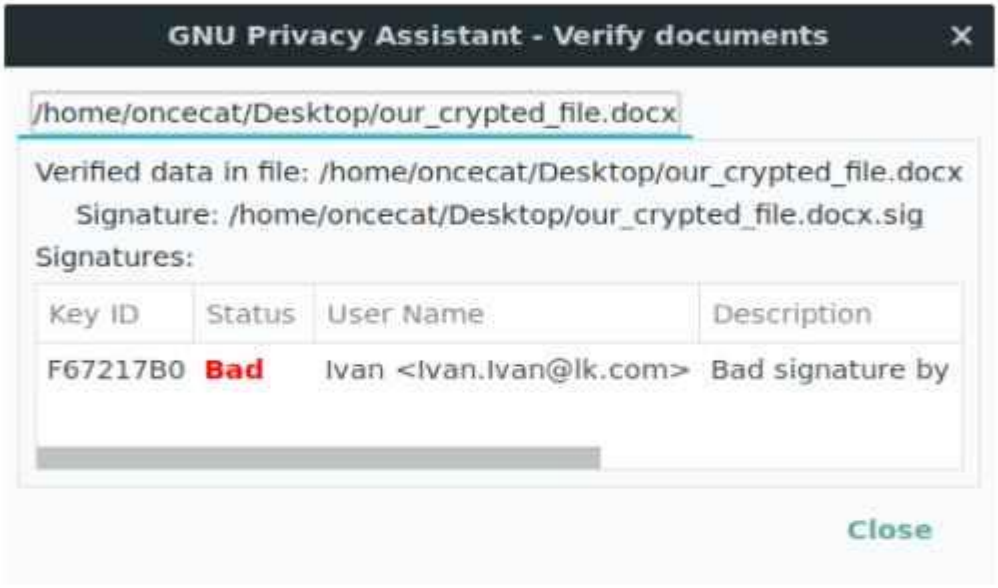

Рис 19. Подписанный файл не соответствует заявленному