Уязвимость системы защиты данных DPAPI в Win2K, Win2K3, Windows Server 2008, Windows Server 2012

Passcape Software

1 Описание уязвимости

1.1 Суть проблемы

В одной из наших предыдущих статей мы описывали принцип работы системы DPAPI, где отмечали ее исключительную надежность, функциональность и стойкость ко взлому. Совсем недавно обнаружилось, что некоторые DPAPI блобы имеют проблему с расшифровкой. После выяснения причины возникающей проблемы, была выявлена довольно интересная брешь в системе безопасности DPAPI во всех серверных системах, начиная от Win2K и заканчивая Windows Server 2012.

Если вкратце, то суть ее сводится к следующему: по умолчанию, Мастер ключи системы безопасности DPAPI всех пользователей домена с правами интерактивного входа в систему, за исключением встроенных учетных записей, создаются в режиме совместимости с Windows 2000, поэтому для расшифровки данных, зашифрованных при помощи DPAPI, знание пароля владельца не требуется.

1.2 Что затрагивает

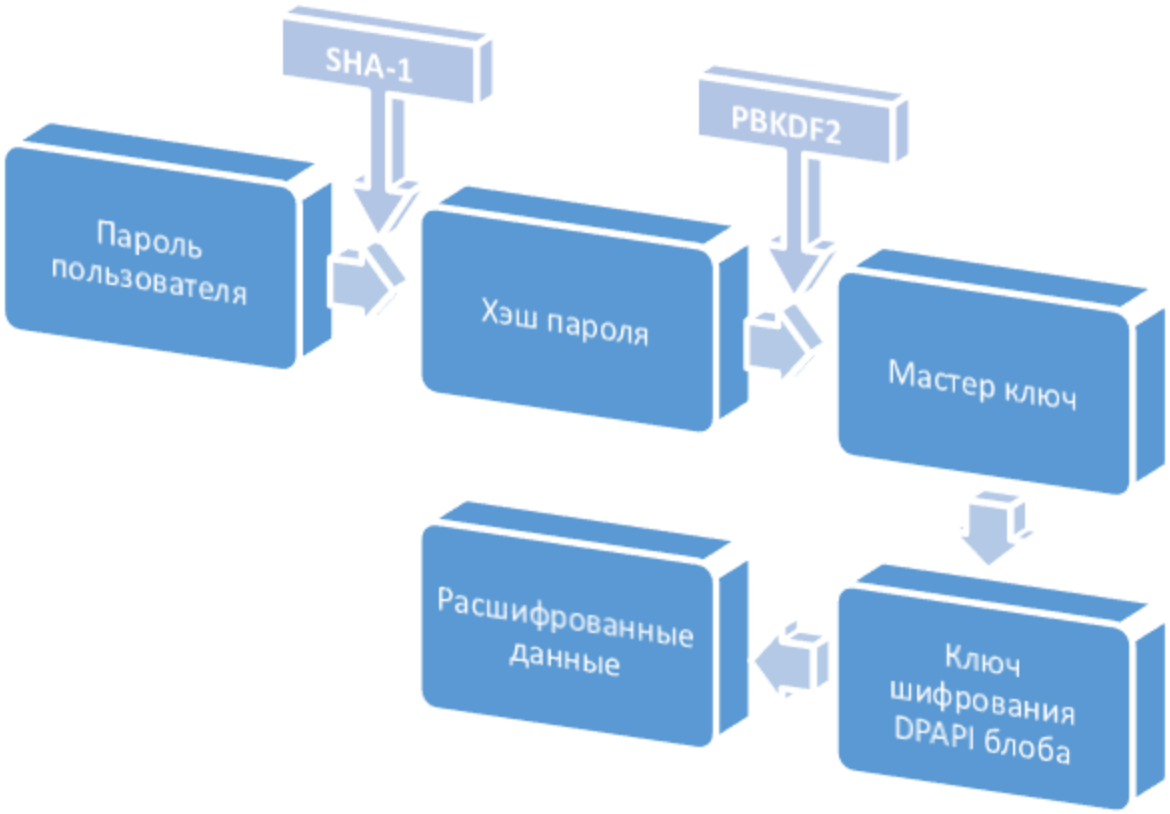
Это означает, что любой Администратор или любой пользователь, имеющий физический доступ к серверу, может расшифровать следующие персональные данные учетных записей уязвимых пользователей:

- Пароли и данные авто-заполнения форм в популярных браузерах Internet Explorer, Google Chrome, Opera Browser
- Пароли учетных записей почты в Outlook, Windows Mail, Windows Mail, и т.д.
- Пароли учетных записей встроенного менеджера FTP
- Пароли доступа к общим папкам и ресурсам
- Пароли и ключи учетных записей беспроводной сети
- Ключи шифрования в Windows CardSpace
- Ключи шифрования Windows Vault
- Пароли соединений удаленного доступа к рабочему столу
- Пароли .NET Passport
- Приватные ключи Системы Шифрования Файлов (EFS)
- Ключи шифрования почты S-MIME
- Сертификаты пользователя
- Приватные данные в Internet Information Services
- EAP/TLS и 802.1x аутентификация
- Сетевые пароли в Credential Manager
- Персональные данные любого приложения, программно защищенные при помощи API функции CryptProtectData. Например, Skype, Windows Rights Management Services, Windows Media, MSN messenger, Google Talk и др.

2 Технические детали

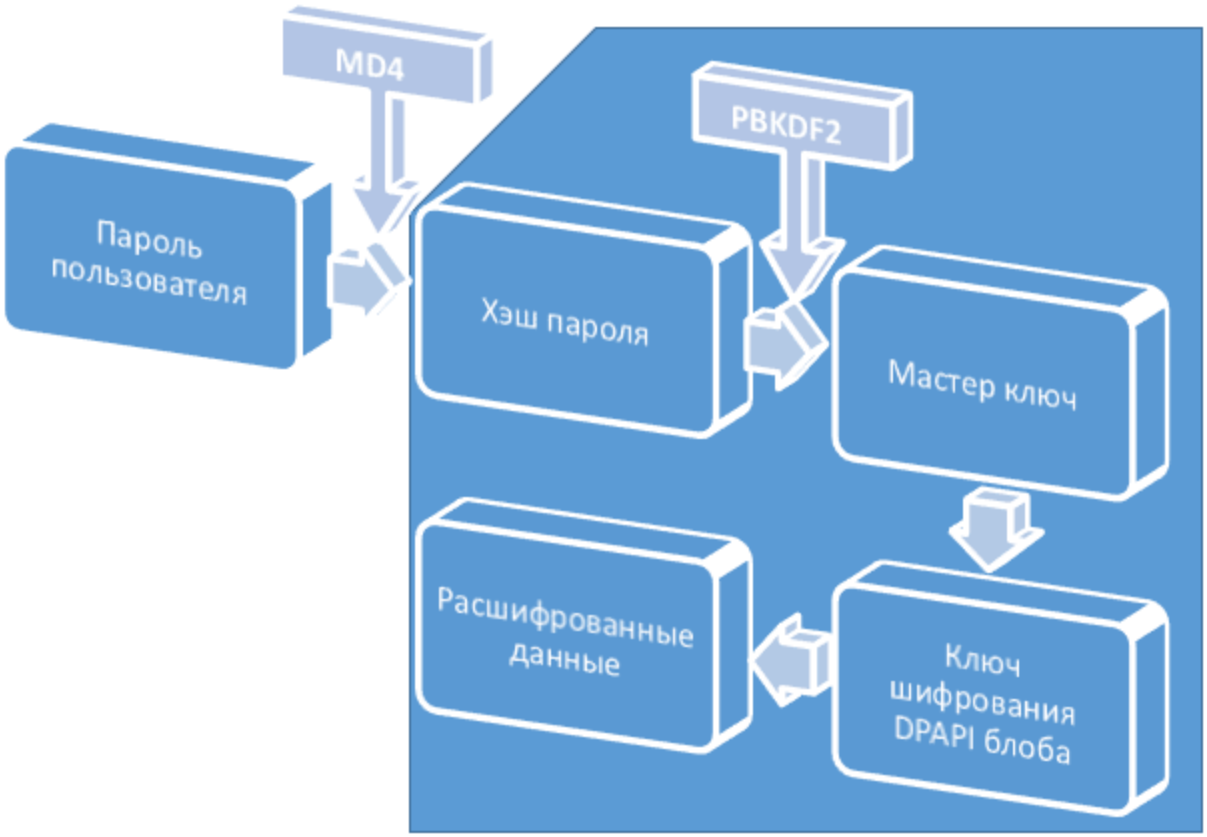
2.1 Шифрование DPAPI в ОС Windows XP +

Процесс расшифровки приватных данных, зашифрованных с помощью DPAPI, выглядит следующим образом (некоторые детали опущены): на начальном этапе логон пароля владельца данных пропускается через SHA-1, чтобы получить хэш пароля, хэш пароля и SID владельца затем скармливаются в функцию PBKDF2. На выходе получается prekey, который участвует в расшифровке Мастер Ключа. Расшифрованный Мастер ключ, в свою очередь, используется для дешифровки самих DPAPI блобов. Вот как это выглядит на схеме:



2.2 Шифрование DPAPI в Windows 2000

Первая реализация DPAPI, используемая в Windows 2000, имела другие алгоритмы шифрования Мастер ключа. Но не это делало ее чрезвычайно уязвимой, а то, что для получения хэша пароля пользователя вместо хэш-функции SHA-1 использовалась MD4. Вот как это выглядит:

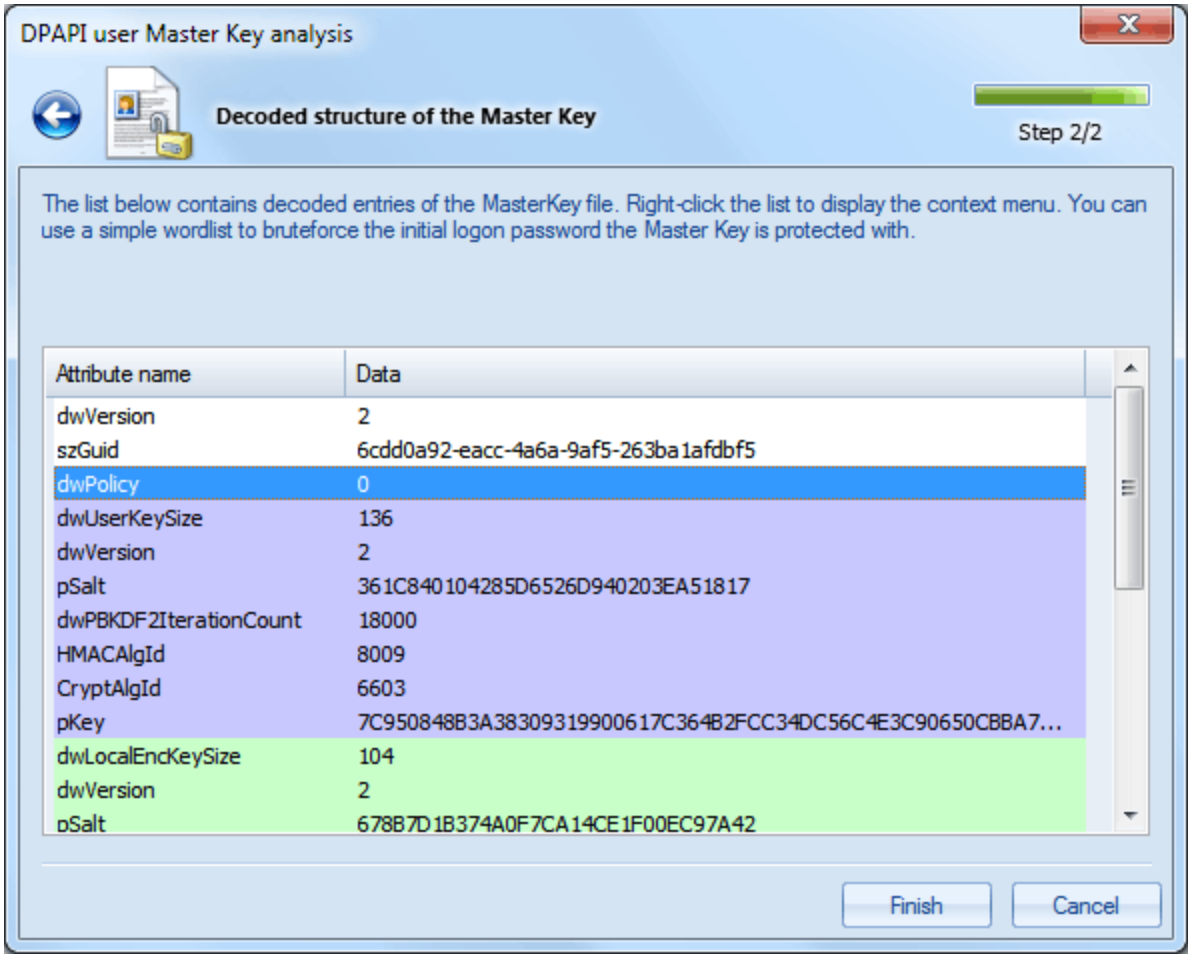


Забавно, но эта же функция хэширования используется и для проверки паролей пользователей, хэши которых хранятся либо в реестре SAM, либо в Active Directory (в серверных ОС). Таким образом, для расшифровки Мастер ключа знания текстового пароля пользователя необязательно. Достаточно взять из SAM или из NTDS.DIT готовый MD4 хэш соответствующего пользователя и использовать его в качестве входных данных. Все остальное нам известно.

2.3 Шифрование DPAPI в Windows 2003, 2008, 2012

Какой алгоритм используется при расшифровки Мастер ключа DPAPI, SHA-1 или MD4, задается в служебном заголовке. Бит 4 поля dwPolicy означает, что в данном Мастер

ключе используется алгоритм SHA-1. У пользователей с правами интерактивного входа в систему, этот флаг по-умолчанию не установлен. Соответственно, для расшифровки приватных данных, знание пароля на вход не требуется.



3 Как воспроизвести проблему

3.1 Создание интерактивного пользователя в домене Windows Server 2012

Перейдем от теории к практике и попробуем добавить нового пользователя в домене Windows Server 2012, создадим какой-нибудь DPAPI секрет и попробуем расшифровать его в режиме offline без пароля владельца.

Открываем консоль «Active Directory Users and Computers» и создаем нового пользователя домена с именем Test. Даем ему право интерактивного входа в систему. Для этого можно просто добавить его к группе локальных администраторов.

3.2 Создание Мастер Ключа и секрета DPAPI для нового пользователя

Теперь нам надо выйти из системы и залогиниться под вновь созданной учетной записью. Новая учетная запись пока не имеет Мастер ключа. Он будет создан при первом вызове функции CryptProtectData. Мы ускорим этот процесс, форсировав вызов соответствующей функции. Для этого у нас есть одноименная утилита CryptProtectData.exe, которая просто вызывает API функцию CryptProtectData с данными командной строки (исходный код утилиты доступен на сайте).

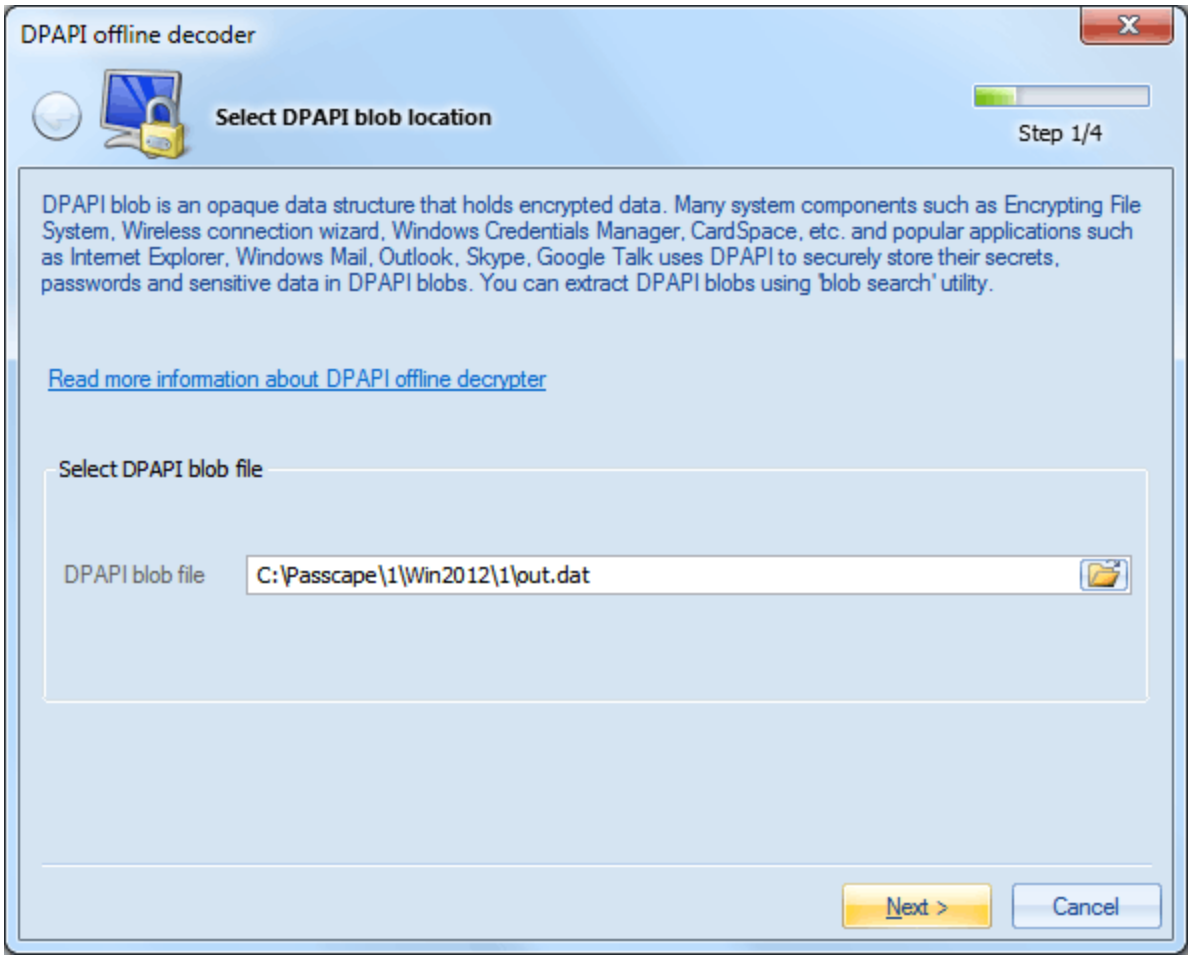
Запустим ее с такими параметрами: **CryptProtectData mysupersecret out.dat**. На выходе получаем файл out.dat с DPAPI блобом, в котором зашифрован наш текст (mysupersecret).

Итак, мастер ключ создался, он находится в каталоге *C:\Users\test\AppData\Roaming\Microsoft\Protect\<SID>\<mk>*
Где *<SID>* - sid владельца. Он еще нам понадобится, запоминаем его, либо копируем весь каталог целиком.
<mk> - имя Мастер ключа DPAPI. К примеру, *6cdd0a92-eacc-4a6a-9af5-263ba1afdbf5*

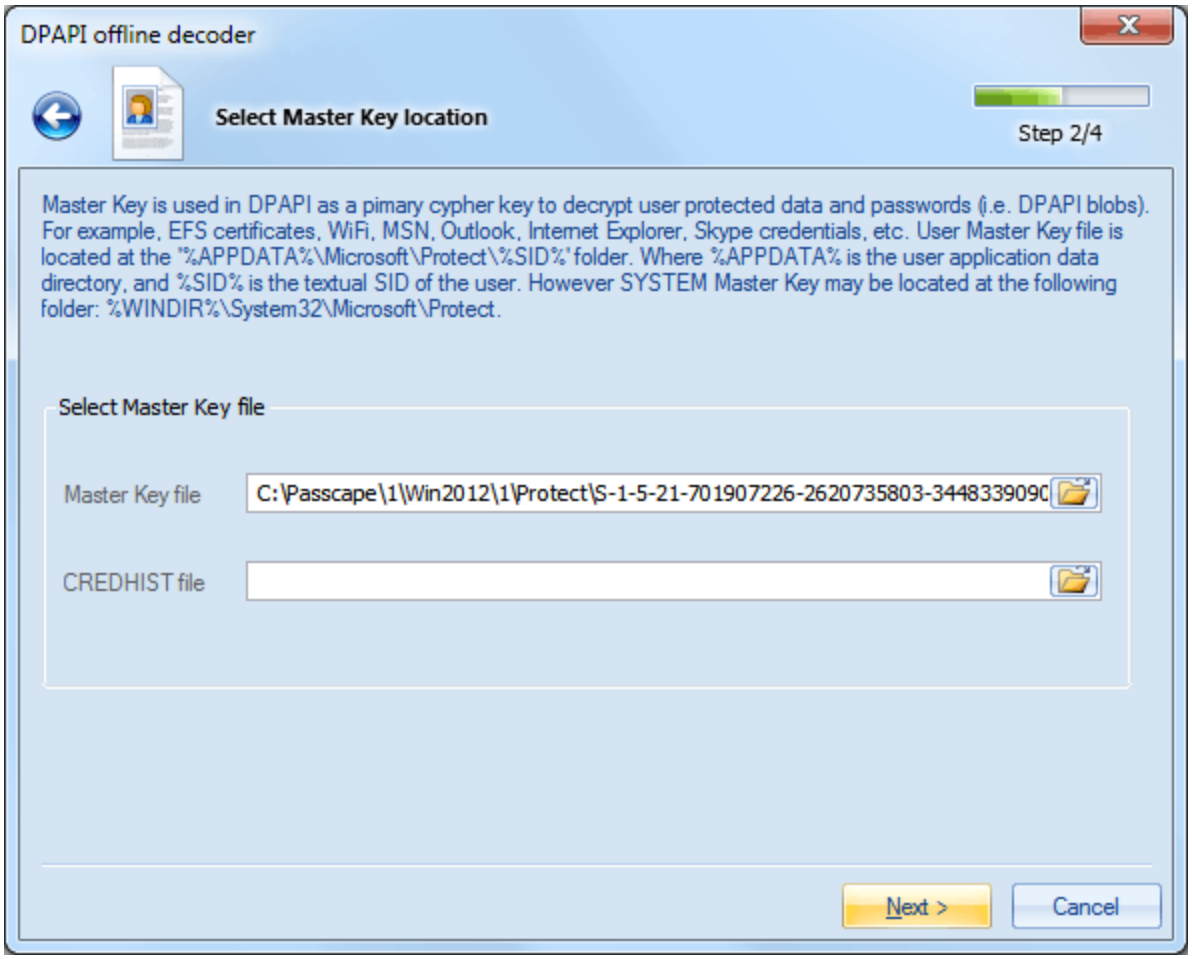
Еще для offline расшифровки out.dat нам понадобится MD4 хэш владельца данных, хранящийся в Active Directory. Для этого с помощью нашей утилиты делаем копию файла NTDS.DIT, а также копируем реестр SYSTEM, необходимый для получения хэша.

3.3 Расшифровка секрета DPAPI без знания пароля пользователя

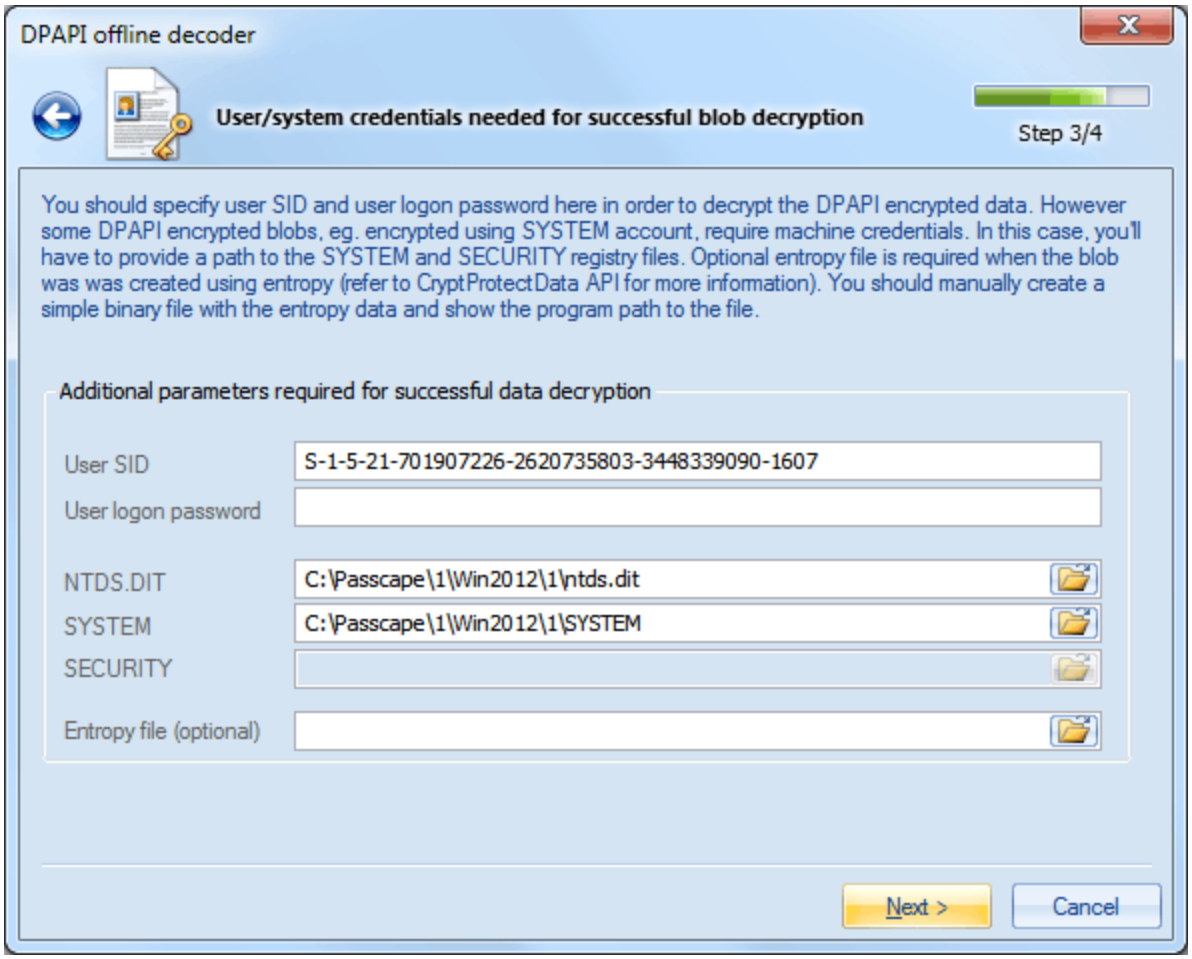
Итак, мы имеем:
- Файл out.dat с зашифрованным секретом, который нам надо расшифровать
- Текстовый SID владельца данных
- Его Мастер ключ
- Хэш владельца (файлы NTDS.DIT и SYSTEM)



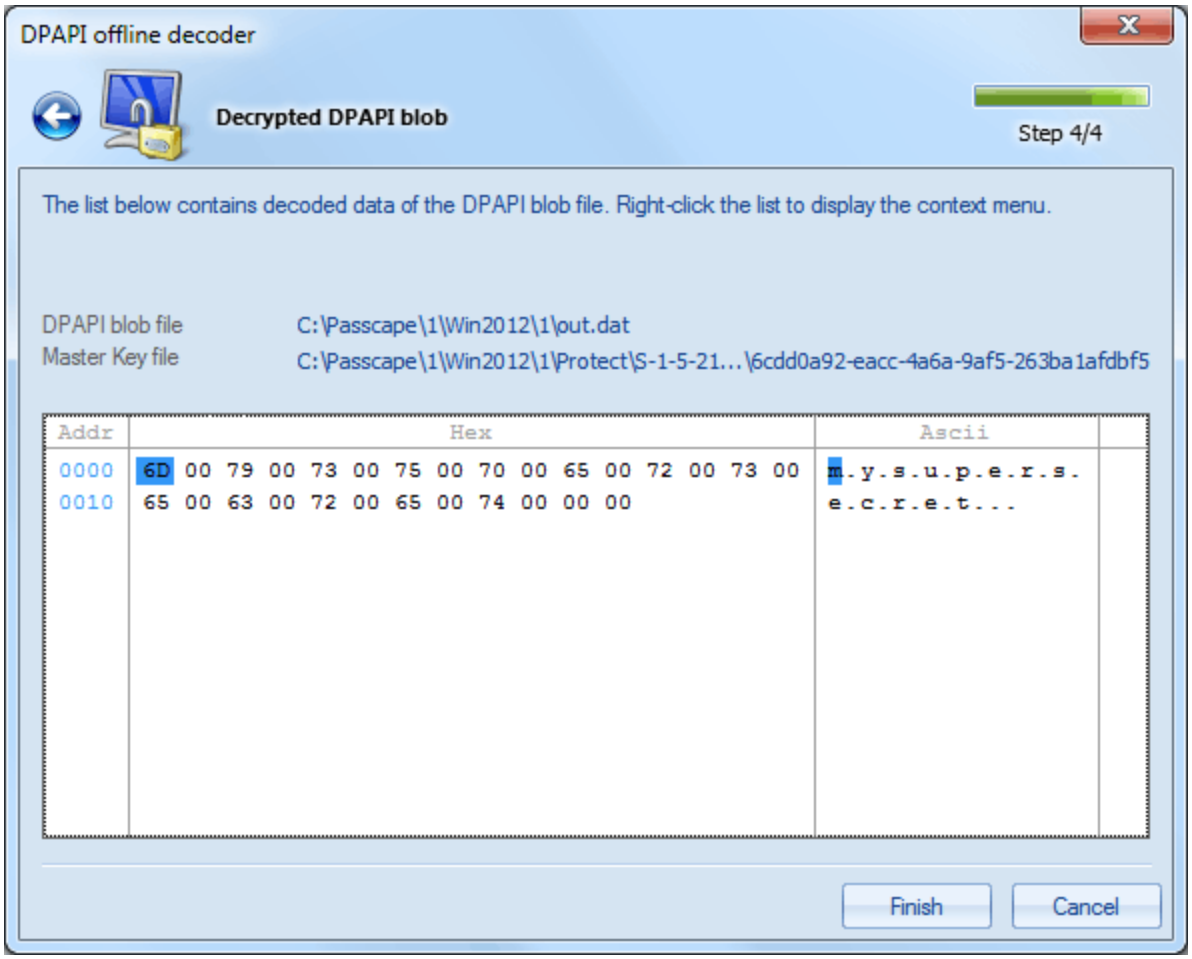
Запускаем утилиту для расшифровки DPAPI и указываем путь к файлу out.dat.



На втором шаге указываем путь к Мастер ключу владельца данных и переходим к следующему шагу. Программа выдает предупреждение, что в данном Мастер ключе обнаружена уязвимость, поэтому возможна расшифровка двумя способами: при помощи пароля пользователя или без него. Нас, конечно, интересует последний.



Поэтому на следующем шаге Мастера вводим SID владельца и пути к файлам NTDS.DIT и SYSTEM, оставляя поле для пароля пустым.



Жмем Далее и получаем наш расшифрованный секрет. Как видим, пароль владельца не понадобился.

4 Заключение

Остается не совсем понятным, связана ли уязвимость интерактивных пользователей с ошибкой или ее не следует рассматривать в таком контексте. Возможно, это особенность реализации DPAPI на серверных ОС. Например, для обеспечения обратной совместимости. С другой стороны, такое кажется маловероятным, поскольку сохранность конфиденциальных данных пользователя не обеспечивается должным образом, а DPAPI как раз является системой защиты информации на основе пароля владельца. Забавно, но получается, что десктопные ПК более устойчивы ко взлому, чем серверные ОС. Так или иначе, системные администраторы должны знать об этой уязвимости. А кто предупрежден, тот вооружен.