

**Федеральное государственное автономное образовательное учреждение
высшего образования**

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Московский институт электроники и математики им. Тихонова

Департамент электронной инженерии

ОТЧЕТ

О ПРАКТИЧЕСКОЙ РАБОТЕ №6

по дисциплине «Программные и аппаратные средства защиты информации»

«Перебор паролей с помощью программных средств»

Студент гр. БИБ201

Шадрунов Алексей

Дата выполнения: 18 июня 2023 г.

Преподаватель:

Перов А. А.

«__» _____ 2023 г.

Москва, 2023

Содержание

1	Цель работы	3
2	Ход работы	3
2.1	Дамп паролей	3
2.2	Подбор паролей	8
2.3	Анализ	12
2.3.1	Влияние метода шифрования	12
2.3.2	Влияние длины пароля	12
2.3.3	Наличие спецсимволов	12
3	Выводы о проделанной работе	13

1 Цель работы

Цель: изучить принципы формирования паролей в операционных системах Windows начиная с Windows Server 2012 и Windows 8.1. Также изучить процесс создания надежных паролей, парольных фраз и политики паролей.

2 Ход работы

2.1 Дамп паролей

Мы будем взламывать пароли, хранящиеся в виде LM и NT хэшей. Для этого на виртуальной машине с Windows XP создадим нескольких пользователей (alex — администратор, user — ограниченный пользователь). После этого скачаем утилиту pwdump6 (Рисунок 1).

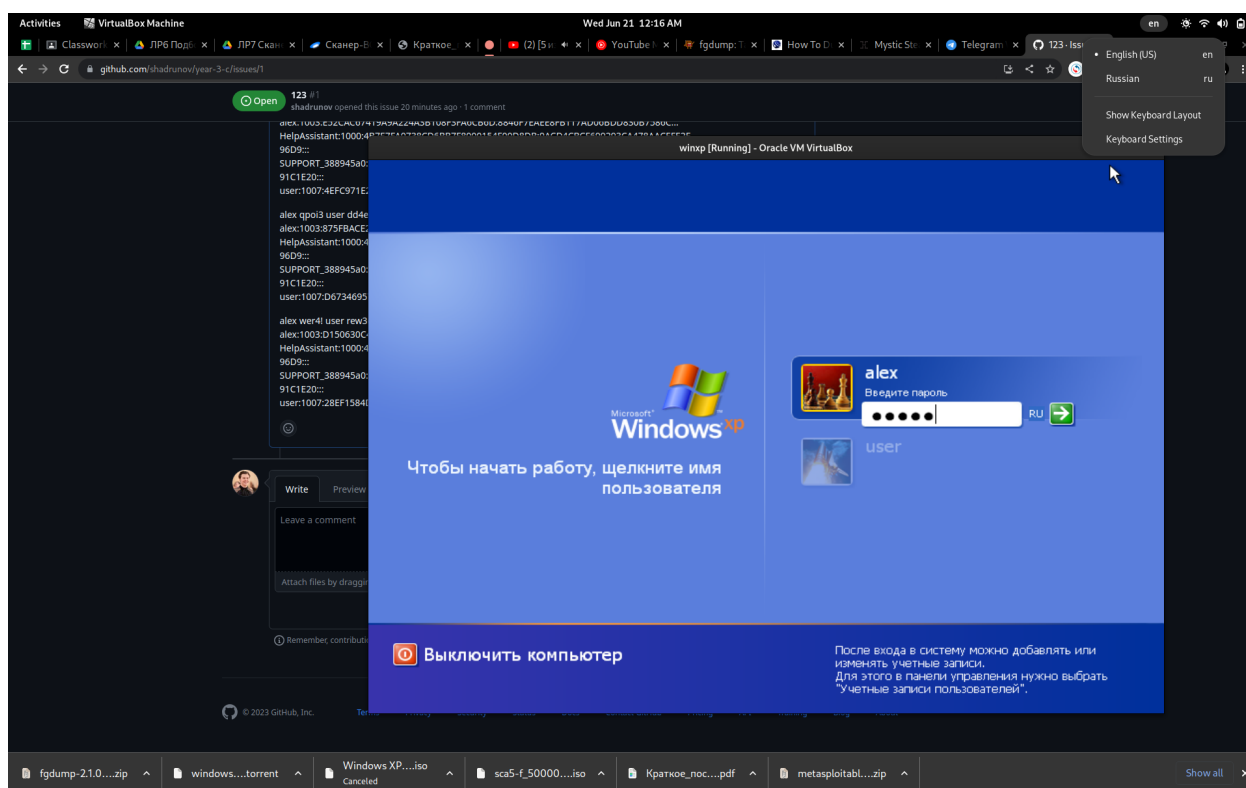


Рисунок 1 – Создание пользователей

По умолчанию в системе пароли хранятся в виде обеих хэшей. С помощью команды `pwdump.exe localhost` сделаем дамп паролей всех пользователей (Рисунок 2).

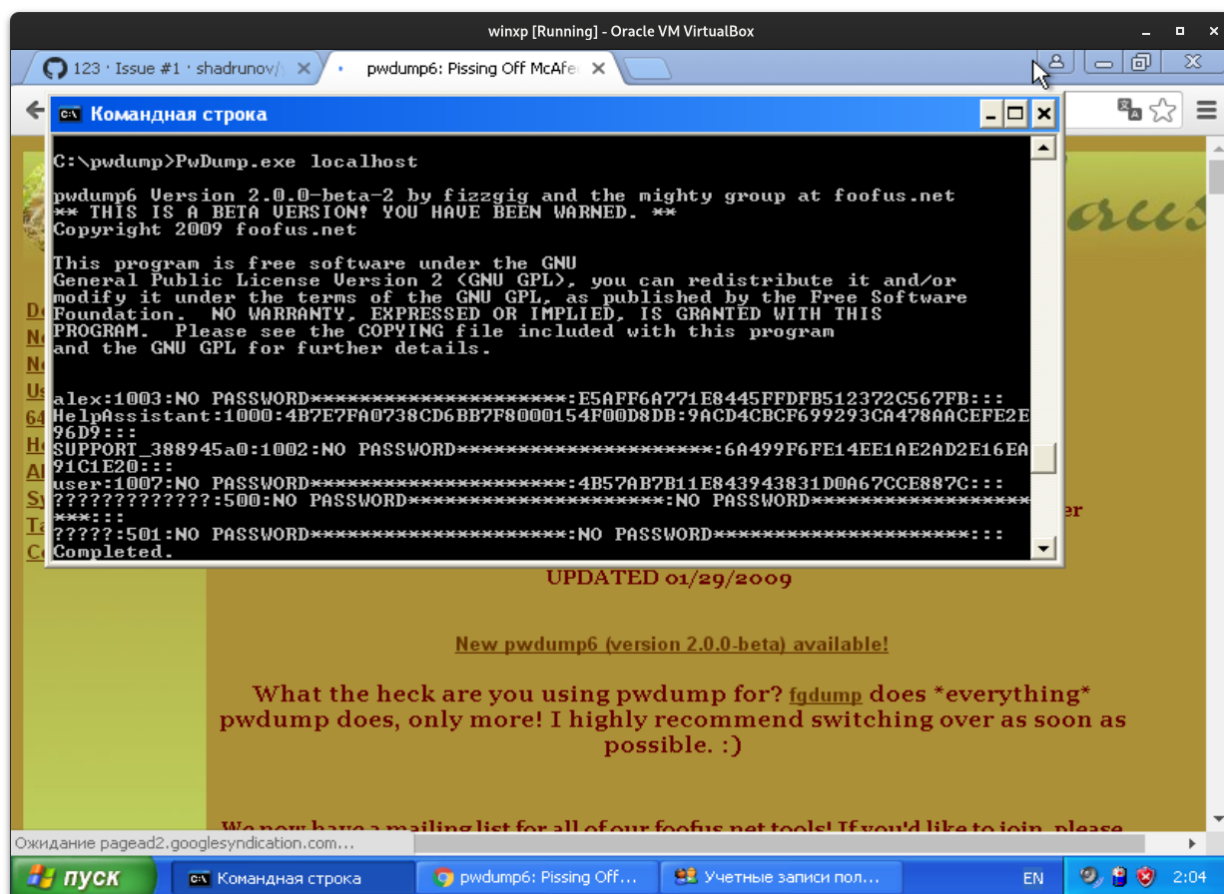


Рисунок 2 – Дамп LM и NT хэшей

Будем менять пароли по схеме: числовой (до 6 символов), числовой (6-10 символов), буквенный (до 6 символов), словарное слово (до 6 символов), буквенный (6-10 символов), словарное слово (6-10 символов), смешанный (буквы и цифры до 6 символов), смешанный (буквы, цифры, спецсимволы до 6 символов). Затем снова будем делать дампы.

Также попробуем отключить хранение LM хэшей. Для этого отредактируем реестр, как показано на рисунках 3-4.

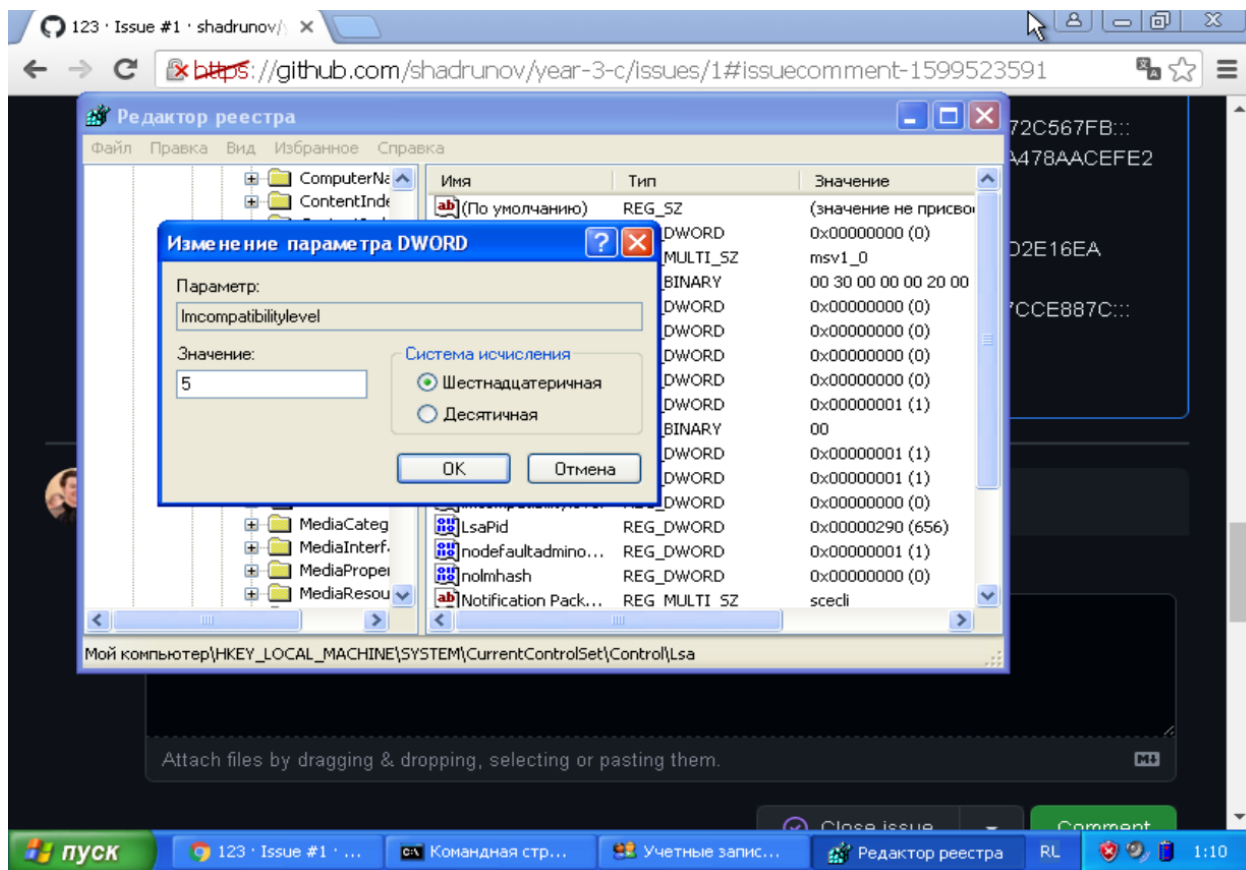


Рисунок 3 – regedit

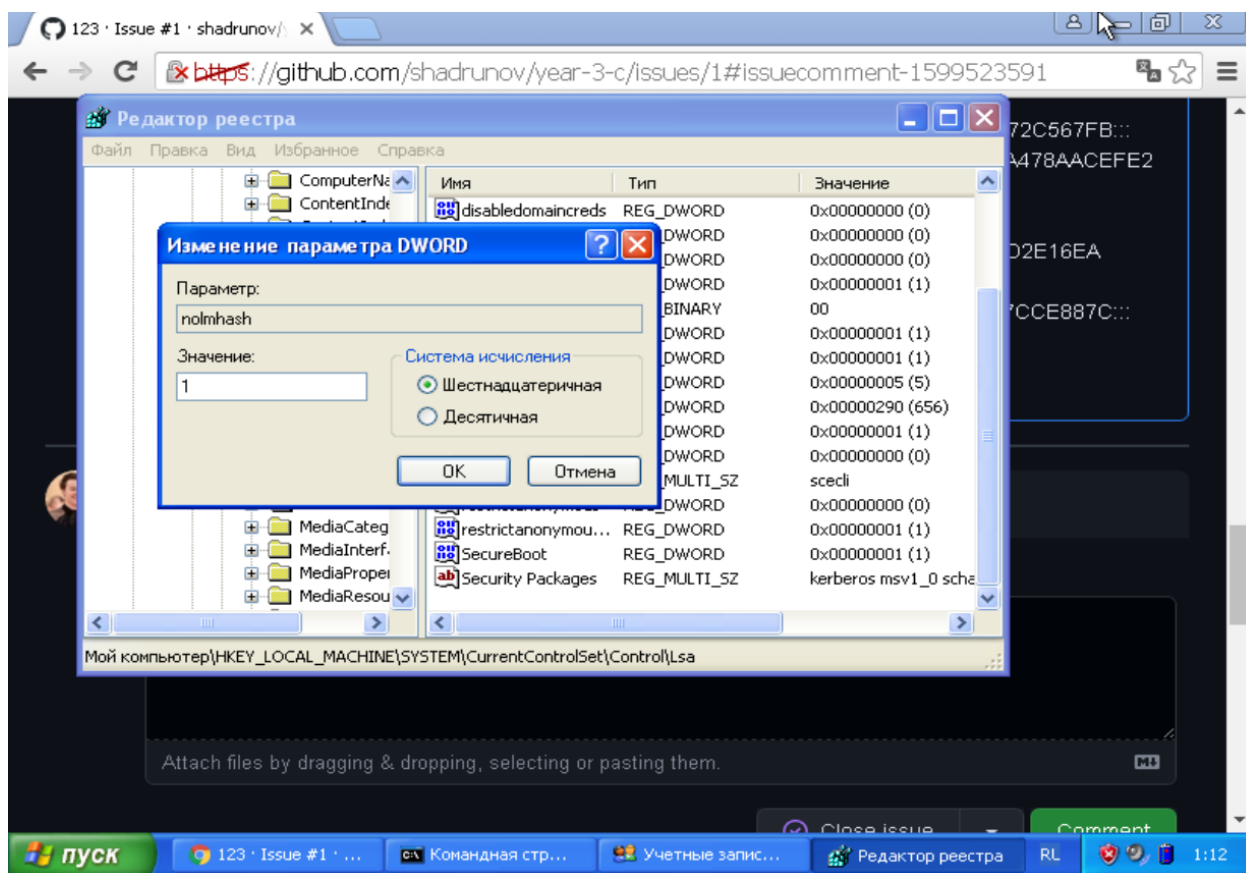
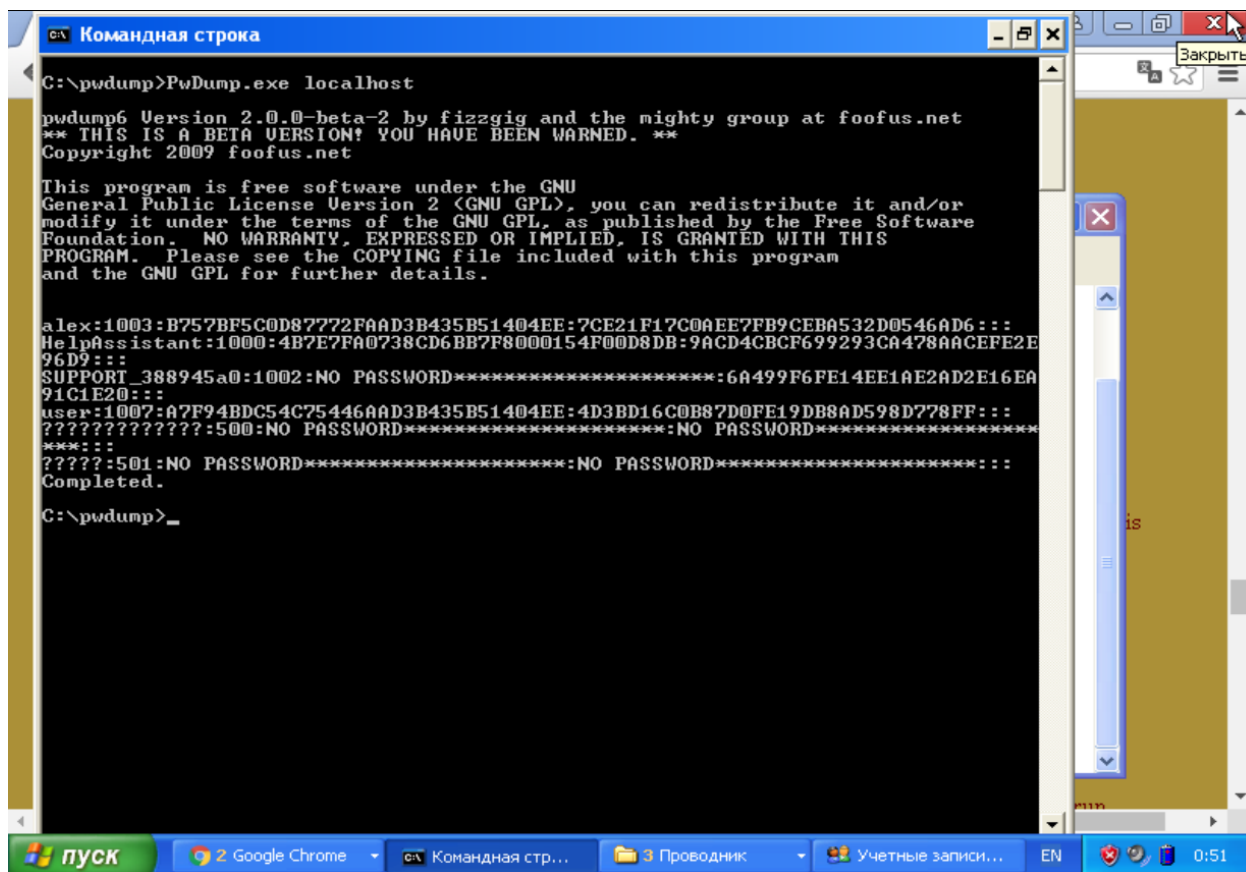


Рисунок 4 – regedit

После этого LM хэш больше не выводится в дампе, только NT (Рисунок 5).



```
C:\pvdump>PwDump.exe localhost

pvdump6 Version 2.0.0-beta-2 by fizzgig and the mighty group at foofus.net
** THIS IS A BETA VERSION! YOU HAVE BEEN WARNED. **
Copyright 2009 foofus.net

This program is free software under the GNU
General Public License Version 2 (GNU GPL), you can redistribute it and/or
modify it under the terms of the GNU GPL, as published by the Free Software
Foundation. NO WARRANTY, EXPRESSED OR IMPLIED, IS GRANTED WITH THIS
PROGRAM. Please see the COPYING file included with this program
and the GNU GPL for further details.

alex:1003:B757BF5C0D87772FAAD3B435B51404EE:7CE21F17C0AEE7FB9CEBA532D0546AD6:::
HelpAssistant:1000:4B7E7FA0738CD6BB7F8000154F00D8DB:9ACD4CBCF699293CA478AAACEFE2E
96D9:::
SUPPORT_388945a0:1002:NO PASSWORD*****:6A499F6FE14EE1AE2AD2E16EA
91C1E20:::
user:1007:A7F94BDC54C75446AAD3B435B51404EE:4D3BD16C0B87D0FE19DB8AD598D778FF:::
?????????:500:NO PASSWORD*****:NO PASSWORD*****:
*****:
?????:501:NO PASSWORD*****:NO PASSWORD*****:
Completed.

C:\pvdump>
```

Рисунок 5 – Дамп только NT хэшей

Результаты всех дампов приведены в таблице 1.

Таблица 1 – NT и LM хэши

LM

alex

1234	B757BF5C0D87772FAAD3B435B51404EE
1234567	0182BD0BD4444BF8AAD3B435B51404EE
abcd	E165F0192EF85EBBAAD3B435B51404EE
qwerty	598DDCE2660D3193AAD3B435B51404EE
password	E52CAC67419A9A224A3B108F3FA6CB6D
qpoi3	875FBACE27462DB2AAD3B435B51404EE
wer4	D150630C44CEB678AAD3B435B51404EE

user

5678	A7F94BDC54C75446AAD3B435B51404EE
7654321	9EE0D521C88B2C76AAD3B435B51404EE
efgh	6391AAA063992CA8AAD3B435B51404EE
pass	B267DF22CB945E3EAAAD3B435B51404EE
windows	4EFC971E2C6A11F0AAD3B435B51404EE
dd4er	D6734695717A0410AAD3B435B51404EE
rew3@	28EF1584D1C91091AAD3B435B51404EE

NT

alex

1234	7CE21F17C0AEE7FB9CEBA532D0546AD6
1234567	328727B81CA05805A68EF26ACB252039
abcd	EB4FF39B74B0CBCE20A4F62DBD1E3585
qwerty	2D20D252A479F485CDF5E171D93985BF
password	8846F7EAAEE8FB117AD06BDD830B7586C
qpoi3	E6707CAD017AA910A75D5B2386DB5B87
wer4	E5AFF6A771E8445FFDFB512372C567FB

user

5678	4D3BD16C0B87D0FE19DB8AD598D778FF
7654321	8BB80565A55DEAA6E1847DC1BC3505FA
efgh	A0B95A38B5E679942183EF613FCB3C18
pass	36AA83BDCAB3C9FDAF321CA42A31C3FC
windows	A2345375A47A92754E2505132ACA194B
dd4er	0DE51E26F132968A78E39A36BF5D1C53
rew3@	4B57AB7B11E843943831D0A67CCE887C

2.2 Подбор паролей

Теперь нужно подобрать пароли, которые отвечают полученным хэшам. Для этого воспользуемся утилитой hashcat. Аргументы, которые нам понадобятся:

- -m 3000 — тип хэша. 3000 означает LM, 1000 — NTLM (NT);
- -a 3 — тип атаки. 3 означает атаку по маске, 0 — атака по словарю;
- -i — перебирать все возможные длины паролей;
- --outfile — выходной файл;
- --outfile-format=1,2,5,6 — формат выходного файла. 5 и 6 добавляют вывод метки времени, 1 и 2 — хэш и пароль;
- --potfile-path — путь к файлу кэша, куда hashcat помещает взломанные хэши;
- маска — мы будем использовать маску ?a?a?a?a?a?a для взлома LM, так как длина пароля в этом хэше не превышает 7 символов и мы знаем, что он состоит из букв, цифр и символов.

На рисунках 6-7 показан запуск hashcat для взлома первого хэша.

```
[alex@alex-nb 7]$ hashcat -m 3000 -a 3 -i --outfile lm_alex.out --potfile-path=lm_alex.pot --outfile-format=1,2,5,6 B757BF5C0D87772FAAD3B435B51404EE ?a?a?a?a?a?a
hashcat (v6.2.6) starting

OpenCL API (OpenCL 2.1 LINUX) - Platform #1 [Intel(R) Corporation]
=====
* Device #1: 12th Gen Intel(R) Core(TM) i5-1235U, 7816/15696 MB (1962 MB allocatable)
, 12MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 7

Hashes: 2 digests; 2 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates

Optimizers applied:
* Zero-Byte
* Not-Iterated
* Single-Salt
* Brute-Force

Watchdog: Temperature abort trigger set to 90c

INFO: Removed hash found as empty hash.

Host memory required for this attack: 3 MB
```

Рисунок 6 — hashcat


```

Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 3000 (LM)
Hash.Target.....: aad3b435b51404ee
Time.Started.....: Wed Jun 21 09:46:52 2023 (0 secs)
Time.Estimated...: Wed Jun 21 09:46:52 2023 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Mask.....: ?a?a?a?a [4]
Guess.Queue.....: 4/7 (57.14%)
Speed.#1.....: 147.8 MH/s (1.00ms) @ Accel:256 Loops:1024 Thr:1 Vec:8
Recovered.....: 2/2 (100.00%) Digests (total), 1/2 (50.00%) Digests (new)
Progress.....: 211968/22667121 (0.94%)
Rejected.....: 0/211968 (0.00%)
Restore.Point....: 0/328509 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-69 Iteration:0-1024
Candidate.Engine.: Device Generator
Candidates.#1....: 1234 -> G#$
Hardware.Mon.#1..: Temp: 63c Util: 65%

Started: Wed Jun 21 09:46:50 2023
Stopped: Wed Jun 21 09:46:53 2023
[alex@alex-nb 7]$

```

Рисунок 7 – hashcat

На рисунке 8 виден выходной файл. Видно, что перебор пароля занял меньше 1 секунды.

```

hw > 7 > ≡ lm_alex.out
1 1687330012:0:b757bf5c0d87772f:1234
2

```

Рисунок 8 – hashcat

Для словарных паролей попробуем атаку по словарю (-a 0). Для этого воспользуемся словарём xato-net-10-million-passwords-10000.txt. Результат на рисунках ниже:

```

[alex@alex-nb 7]$ wget https://raw.githubusercontent.com/danielmiessler/SecLists/master/Passwords/xato-net-10-million-passwords-10000.txt
--2023-06-21 15:36:38-- https://raw.githubusercontent.com/danielmiessler/SecLists/master/Passwords/xato-net-10-million-passwords-10000.txt
Loaded CA certificate '/etc/ssl/certs/ca-certificates.crt'
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.109.133, 185.199.110.133, 185.199.108.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.109.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 76497 (75K) [text/plain]
Saving to: 'xato-net-10-million-passwords-10000.txt'

xato-net-10-million-p 100%[=====] 74.70K --.-KB/s in 0.09s

2023-06-21 15:36:39 (808 KB/s) - 'xato-net-10-million-passwords-10000.txt' saved [76497/76497]

```

Рисунок 9 – Скачиваем список паролей

```

[alex@alex-nb 7]$ hashcat -m 3000 -a 0 --outfile lm_alex.out --potfile-path=lm_alex.p
ot --outfile-format=1,2,5,6 4EFC971E2C6A11F0AAD3B435B51404EE xato-net-10-million-pass
words-10000.txt
hashcat (v6.2.6) starting

OpenCL API (OpenCL 2.1 LINUX) - Platform #1 [Intel(R) Corporation]
=====
* Device #1: 12th Gen Intel(R) Core(TM) i5-1235U, 7816/15696 MB (1962 MB allocatable)
, 12MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 7

Hashes: 2 digests; 2 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Not-Iterated
* Single-Salt

Watchdog: Temperature abort trigger set to 90c

INFO: Removed hash found as empty hash.

Host memory required for this attack: 3 MB

Dictionary cache built:
* Filename...: xato-net-10-million-passwords-10000.txt
* Passwords..: 10000
* Bytes.....: 76497
* Keyspace...: 10000
* Runtime....: 0 secs

```

Рисунок 10 – Запускаем утилиту

```

Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 3000 (LM)
Hash.Target.....: 4efc971e2c6a11f0
Time.Started....: Wed Jun 21 15:36:50 2023 (0 secs)
Time.Estimated...: Wed Jun 21 15:36:50 2023 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (xato-net-10-million-passwords-10000.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 8208.3 kH/s (0.63ms) @ Accel:1024 Loops:1 Thr:1 Vec:8
Recovered.....: 2/2 (100.00%) Digests (total), 1/2 (50.00%) Digests (new)
Progress.....: 10000/10000 (100.00%)
Rejected.....: 0/10000 (0.00%)
Restore.Point....: 0/10000 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: 123456 -> BLITZ
Hardware.Mon.#1..: Temp: 51c Util: 11%

Started: Wed Jun 21 15:36:49 2023
Stopped: Wed Jun 21 15:36:51 2023

```

Рисунок 11 – Готово

Повторим процедуру для остальных LM хэшей. Результат в таблице 2.

Аналогично для NT хэшей используем код 1000 (Таблица 3):

Таблица 2 – LM хэши

LM			
alex		атака маской	атака по словарю
1234	B757BF5C0D87772FAAD3B435B51404EE	0	0
1234567	0182BD0BD4444BF8AAD3B435B51404EE	1944	0
abcd	E165F0192EF85EBBAAD3B435B51404EE	0	0
qwerty	598DDCE2660D3193AAD3B435B51404EE	16	0
password	E52CAC67419A9A224A3B108F3FA6CB6D	1018	0
qpoi3	875FBACE27462DB2AAD3B435B51404EE	5	-
wer4!	D150630C44CEB678AAD3B435B51404EE	5	-
user			
5678	A7F94BDC54C75446AAD3B435B51404EE	0	0
7654321	9EE0D521C88B2C76AAD3B435B51404EE	1116	0
efgh	6391AAA063992CA8AAD3B435B51404EE	0	-
pass	B267DF22CB945E3EAAAD3B435B51404EE	0	0
windows	4EFC971E2C6A11F0AAD3B435B51404EE	~10 часов	0
dd4er	D6734695717A0410AAD3B435B51404EE	0	-
rew3@	28EF1584D1C91091AAD3B435B51404EE	3	-

Таблица 3 – NT хэши

NT			
alex		атака маской	атака по словарю
1234	7CE21F17C0AEE7FB9CEBA532D0546AD6	0	0
1234567	328727B81CA05805A68EF26ACB252039	1817	0
abcd	EB4FF39B74B0CBCE20A4F62DBD1E3585	0	0
qwerty	2D20D252A479F485CDF5E171D93985BF	100	0
password	8846F7EAE8FB117AD06BDD830B7586C	1244	0
qpoi3	E6707CAD017AA910A75D5B2386DB5B87	5	-
wer4!	E5AFF6A771E8445FFDFB512372C567FB	10	-
user			
5678	4D3BD16C0B87D0FE19DB8AD598D778FF	0	0
7654321	8BB80565A55DEAA6E1847DC1BC3505FA	1911	0
efgh	A0B95A38B5E679942183EF613FCB3C18	0	-
pass	36AA83BDCAB3C9FDAF321CA42A31C3FC	0	0
windows	A2345375A47A92754E2505132ACA194B	~10 часов	0
dd4er	0DE51E26F132968A78E39A36BF5D1C53	5	-
rew3@	4B57AB7B11E843943831D0A67CCE887C	5	-

2.3 Анализ

2.3.1 Влияние метода шифрования

В использованных нами методах атаки (по маске и по словарю) тип шифрования не оказывает существенного влияния, так как сложность взлома определяется количеством перебираемых вариантов, а время, затрачиваемое на хэширование, мало и примерно одинаково.

Атака по словарю в обоих случаях работает за малое время, если пароль присутствует в словаре, либо не работает вовсе.

2.3.2 Влияние длины пароля

Длина пароля в случае атаки по маске является определяющим фактором. Чем длиннее пароль, тем сложнее его подобрать. Пароли из 4 символов подбираются очень быстро, пароли из 5 символов — за единицы секунд, из 6 символов — за десятки секунд, из 7 символов — в пределах часа. Так происходит, потому что количество вариантов при добавлении одного символа возрастает в количество раз, равное мощности алфавита.

При подборе по словарю длина пароля не влияет на успех атаки.

2.3.3 Наличие спецсимволов

При переборе по маске наличие спецсимволов делает невозможным отгадывание пароля с простой маской, не учитывающей эти символы. Такая маска включена по умолчанию. Так как мы расширили маску до всех служебных символов на любых позициях, то в нашем случае они не оказывали влияния на успешность атаки.

При подборе по словарю спецсимволы являются хорошей защитой, так как в словаре чаще встречаются обычные слова. Но есть вариант гибридной атаки, когда маска прибавляется к словарным словам.

3 Выводы о проделанной работе

Я изучил принципы формирования паролей в операционных системах Windows начиная с Windows Server 2012 и Windows 8.1. Также изучил процесс создания надежных паролей, парольных фраз и политики паролей, а также освоил утилиту hashcat и pwdump6.