Windows Password Recovery

User manual

Copyright (c) 2023 Passcape Software. All rights reserved. Passcape Software

1. Inti	roduction	6
1.1	About the program	7
1.2	Features and benefits	7
1.3	System requirements	8
2. Pro	gram's interface	9
2.1	Overview	10
2.2	Project menu	11
2.2.1	Import	11
2.2.7	1.1 Import local hashes	. 12
2.2.7	1.2 Import hashes from remote computer	. 13
2.2.	1.3 Import hashes from binary files	. 15
2.2. 2.2	1.4 Importing hashes from system restore folders	. 10 18
2.2.	Fxnort	18
2.2.2	New	19
2.2.3	Open	19
2.2.5	Save	19
2.2.6	Save as	19
2.2.7	Close	19
2.2.8	Hash Import Wizard	19
2.2.9	Hardware Configuration Wizard	20
2.2.10	Password Recovery Wizard	21
2.3	Recovery menu	22
2.3.1	Run	22
2.3.2	Continue	23
2.3.3	Stop	23
2.4	Edit menu	23
2.4.1	Edit	23
2.4.2	Add	24
2.4.3	Delete	24
2.4.4	Reset passwords	24
2.4.5	Сору	24
2.4.6	Select	24
2.4.7	Search	25
2.5	Reports Menu	25
2.5.1	Password reports	26
2.5.2	Attack statistics	27
2.5.3	Miscellaneous statistics	28
2.5.4	Account statistics	29

2.	5.5	Password-list analysis	. 31
2.	5.6	Group information	. 32
2.6	Тоо	ls menu	. 33
2.	6.1	Program access	. 34
2	6.2	Pass-o-meter	35
2	6.3	Password Checker	. 36
2	6.4	Hash Generator	. 36
2.	65	Rainbow Tables Generator	38
2.	6.6 6.6	Pascape Rainbow Tables Generator	30
2.	6.7	Wordlist tools	. 37
۷.	2.6.7.1	Create new wordlist by indexing files	. 40
	2.6.7.2	Merge wordlists	42
	2.6.7.3	Wordlist statistics	43
	2.6.7.4	Sort wordlist	45
	2.6.7.5	Convert/compress wordlist	46
	2.6.7.6	Compare wordlists	47
	2.6.7.7	Additional operations	48
	2.6.7.8	Index HDD sensitive areas	49
	2.6.7.9	Extract HTML links	53
2.7	Utils	s menu	. 55
2.	7.1	Backup system files	. 55
2.	7.2	Asterisk Password Revealer	. 57
2.	7.3	Offline Password Remover	. 57
2.	7.4	Forensic tools	. 61
	2.7.4.1	LSA Secrets Dumper	61
	2.7.4.2	Domain Cached Credentials Explorer	65
	2.7.4.3	Active Directory Explorer	68
	2.7.4.4	SAM Explorer	73
	2.7.4.5	DPAPI tools	79
	2.7.4.5.	1 Decrypt DPAPI blob	80
	2.7.4.5.	2 Analyse DPAPI blob	84
	2.7.4.5.	3 Search DPAPI blobs	86
	2.7.4.5.	4 Master Key analysis	87
	2.7.4.5.	5 Dump user credentials history hashes	90
	2.7.4.5.	 Analyse credential history Windows Vault Evalurar 	91
	2.7.4.0	Windows Hello Explorer	94
	2.7.4.7	1 Windows Hollo crodontials	100
	2747	2 Biometric databases	100
	2.7.4.7	3 PIN brute forcer	103
	2.7.4.8	Windows Credentials Explorer	111
2.8	Sett	ings menu	116
2	8 1	General settings	116
۷.	2.8.1.1	General options	117

2.8.	1.2 Attack options	
2.8.	1.3 CPU settings	
2.8.	1.4 GPU settings	
2.8.	1.5 GPU health monitor	
2.8.	1.6 Event notifications	
2.8.2	Attack Settings	123
2.8.	2.1 Preliminary attack	
2.8.	2.2 Artificial intelligence attack	
2.8.	2.3 Fingerprint attack	
2.8.	2.4 Brute-force attack (exhaustive search)	
2.8.	2.5 Dictionary attack	
2.8.	2.6 Mask attack	
2.8.	2.7 Base-word attack	
2.8.	2.8 Combined dictionary attack	
2.8.	2.9 Pass-phrase attack	
2.8.	2.10 Rainbow tables attack	
2.8.	2.11 Hybrid dictionary attack	
2.8.	2.12 Offitte recovery	
2.0.	2.15 Passcape lable allack	
2.0.	2.14 Datch attack	
2.0.	216 GPU: Fingerprint attack	166
2.8.	2.17 GPU: Mask attack	172
2.8.	2.18 GPU: Dictionary-force Attack	
2.8.	2.19 GPU: Hybrid dictionary attack	
29	View menu	193
2.7		100
2.10	Inemes menu	193
2.11	Help menu	
2.12	Hardware Monitor	
3. Wo	orking with the program	195
3.1	Attacking Windows hashes	
3.2	Attack comparison table	
3.3	Recovering passwords from hashes	
3.4	Windows passwords FAQ	
3.5	Windows Password Recovery FAQ	
3.6	GPU FAQ	
3.7	Online dictionaries	
4. Lice	ense and registration	213
4.1	License agreement	
4.2	Registration	

4.3	Limitation of unregistered version	216
4.4	Editions of the program	216
5. Tec	hnical support	219
5.1	Reporting problems	220
5.2	Suggesting features	220
5.3	Contacts	220

Introduction

1 Introduction

1.1 About the program

Welcome to **Windows Password Recovery**, a network security analyzer and Windows password recovery utility. Windows Password Recovery is the only solution that implements the most advanced, patented password recovery technologies developed by Passcape Software programmers, such as *Artificial Intelligence* or *Pass-phrase* attack.

Compared to similar products, Windows Password Recovery features a number of competitive advantages:

For home users - easy set up and use. Easily recovers or resets forgotten passwords to any Windows account.

<u>For system administrators</u> - password audit reveals security breaches, helping the administrators to ensure the reliability and security of the corporate network. Checks the security level of Windows operating systems.

<u>For forensics, industry and government security experts</u> - analyzes and audits system security policies, issues recommendations on improving the stability of the operating systems' password protection.

1.2 Features and benefits

- Contemporary, easily customizable graphical user interface.
- Support for 14 different file formats when importing hashes.
- Support for NTLM and LM hashes, Domain Cached Credentials, Windows PIN.
- Import directly from Windows registry or Active Directory; even if the files are locked by the system, the program still reads them.
- Import hashes from remote computers.
- Import hashes from system shadow copies, restore points, backup and repair folders.
- Backup\save local registry files and Active Directory database.
- Import password history hashes.
- Recover some account passwords on the fly (when importing locally).
- Support Active Directory (domain accounts).
- Support Domain Cached Credentials.
- Support for Microsoft and AzureAD accounts.
- Support importing from x86/x64 systems.
- Exports hashes to the PWDUMP file.
- The software uses 17 types of different attacks; 10 of them are unique, developed by our company, implemented upon patented technologies.
- The program supports for multithreading, fully leveraging the power of modern computers.
- Dictionary attack supports text dictionaries in the ASCII, UNICODE, UTF8, PCD, RAR and ZIP formats.
- Broad choice of online dictionaries for dictionary attacks (about 2 GB)
- Some of the program's functions e.g., word mutation are unique. For example, the total number of mutation rules exceeds a hundred and fifty. Not any other similar application features that!
- Supports unlimited number of inspected hashes.
- Intelligent analysis of found passwords.

- High search speed on modern computers over 100 million of passwords per second for 4-core CPUs and billions passwords/sec using GPU power.
- Includes auxiliary tools: hash generator, password strength check, rainbow table creation, etc.
- Extended toolset to work with wordlists: create, sort, convert, etc.
- Add-on modules for forensics and researchers: LSA secrets editor, domain cached credentials viewer, Active Directory and SAM explorers, DPAPI offline decoder.
- Clear-text passwords extraction from memory, Windows Cache, Windows Hello, hidden secrets, and so on.
- Advanced password reports

1.3 System requirements

Requirements

Windows XP or newer OS, about 100 Mb hdd space, 512 Mb RAM.

Some features of the program require administrator privileges.

To run a GPU-based attack, you will need a CUDA-based video card with compute capability 3.0, AMDbased GPU starting with 7xxx series, Intel HD Graphics 4xxx or higher

Compatibility

To use NVidia GPU, you will need Windows XP or higher OS. AMD Radeon cards require Windows Vista or higher OSes.

Known problems

The program although contains no harmful code, may be detected by some anti-virus/anti-spyware software as potentially dangerous or 'potentially unwanted program'. This is also known as 'False Alert', and it's quite a common problem for all password recovery software.

Program's interface

2 Program's interface

2.1 Overview

The program's interface is made in the form of the Single Document Interface, i.e. it allows working with only one project at a time. The program's operation can be conventionally divided into 4 stages:

- 1. Creating a project
- 2. Importing (loading) password hashes to the project. Editing the hashes: deleting, adding, selecting, etc.
- 3. Recovering the hashes. Includes selecting, configuring and launching the selected one or several attacks.
- 4. Analyzing the results.

The entire interface can be conventionally divided into several components:

- Menu Bar
- Information Bar for displaying brief information texts like tips, warnings, etc.
- Task Bar duplicates and compliments the menu bar, providing quick access to the most common operations. Consists of three parts:

- Project - includes the main operations over project - like opening, closing, creating a new project, and importing hashes.

- Hash Editor. Duplicates the most common editing operations.
- Tools includes a clock, calendar, and calculator.
- Main Window bears the main burden and consists of 5 parts. The first tab is the welcome window. The second tab contains the list of hashes to be analyzed and recovered. Then there goes a tab with the current attack state (progress) indicator and a tab with the statistics and reports. And finally a tab with the hardware monitor.
- Log Window displays information on the current state of the application, current operation, etc. The program's log can be copied to clipboard or saved to a file (right-clicking opens the corresponding menu).
- Status Bar is designed for informational purposes.



2.2 Project menu

2.2.1 Import

Windows Password Recovery offers a broad range of options for loading hashes depending on your capabilities. There are 5 major ways to import hashes to the program.

2.2.1.1 Import local hashes

Import Wizard		×
	Import hashes from local computer	
Local computer	Note! You can also dump password hashes from SAM/SECURITY and SYSTEM registry files of the current computer. Just switch to the 'Binary files' section and select files you need.	
Remote machine	Dump regular (CAM) bashes	
	Dump regular (SAM) hashes Dump Active Directory hashes Dump Demain Content of Analysis	
Binary files	Dump Windows PINs	
	Dump password history hashes	
	Scan system for plaintext passwords	
Textual or project files	Retrieve plaintext password of the currently logged on users	
- S		
System restor		
	ImportCancel	

Import hashes from the local computer - the most preferable method, as it implies the deepest overall analysis of the system and the passwords. Besides that, the hashes that are imported from the local computer can undergo the sophisticated *Intelligent attack*, which allows to relatively quickly recover the passwords to some accounts.

Importing local hashes runs well regardless to where the hashes are localized: in SAM, in SECURITY or in Active Directory. This item has two additional options: dumping password history hashes and searching for plain-text passwords that are stored in the system. The very process of searching for plain-text passwords is divided into 4 steps and consists of the actual searching for the passwords that are stored in the system using the reverse encryption, searching for the text passwords for the system accounts, searching for start-up passwords and an extra step, when the program analyzes some of the uncovered accounts, passwords to which can also be recovered from the system (for example, for the HomeGroupUser\$ account in Windows 7).

Disabling the last 2 options is not a good idea in most cases, as it allows to relatively painlessly and quickly recover the complete passwords to some of the system's accounts, the password history dump is completely opposite - disabling it is often very useful. For example, when the number of passwords to be imported exceeds hundreds of thousands or even millions. On the other hand, the program has a power of artificial intelligence, so if during an attack it finds one of the history passwords, it will take every effort to recover the remaining passwords by analyzing the user's preferences for the recovered password.

One of the latest version of the program can also dump user history hashes from DPAPI CREDHIST file. So setting the option is recommended now.

The local import functionality requires administrative privileges.

2.2.1.2 Import hashes from remote computer

Import Wizard		×
		t hashes from remote computer
Local computer	Note! Dumping re with many user a	mote hashes may take quite some time to complete (up to several hours for servers counts). Please, be patient.
Remote machine	Remote host:	stress-test-pc
	Share resource:	C\$
	User name:	John
Binary files	Password:	••••••
		 Dump password history hashes Scan system for plaintext passwords
Textual or project files		Load credentials Save credentials
8		
System restor		
		Import Cancel

Import hashes from a remote host. The program has means for dumping hashes from a remote host without employing third-party utilities. This does not compromise the remote system, as it still requires supplying the credentials for the remote host user.

Dumping from a remote host works as follows. First, you should enter the remote host name in the Remote Host field. You can use the [...] button to browse the network. Once you have selected the remote host, set up a shared resource (allowed for both reading and writing), through which the data will be transmitted. Usually, that is either C\$ or ADMIN\$. Here too, you can take advantage of the browse button to the right of the edit box. Next, in the two fields at the bottom type in the remote host account name and the password.

The 'Save Credentials' button saves current settings. Respectfully, the 'Load Credentials' button allows loading existing settings, so that you don't have to enter them manually every time you need them. The password is stored in the encrypted form!

This import option also requires administrative privileges on the target PC.

You may, however, experience some troubles connecting to remote PC, even if you have an Administrator account. When connection to the target PC with Windows Vista/7/8/10, you may get the following error:

📀 16:34:18 June 11 2015>	Application started
16:35:27 June 11 2015>	Importing from remote machine
16:35:27 June 11 2015>	COMP: JOHN-PC
16:35:27 June 11 2015>	SHARE: C\$
16:35:27 June 11 2015>	USER: John
🛞 16:35:30 June 11 2015>	system error 5
🛞 16:35:32 June 11 2015>	Failed to run remote service: can't connect remote machine.

The error 5 indicates that access is denied (even if the target account has Administrator privileges). The problem is that any remote connection in Windows Vista and higher OSes by default cannot perform administrative tasks. Microsoft documentation clearly states the following:

"When a user with an administrator account in a Windows Vista computer's local Security Accounts Manager (SAM) database remotely connects to a Windows Vista computer, the user has no elevation potential on the remote computer and cannot perform administrative tasks. If the user wants to administer the workstation with a SAM account, the user must interactively log on to the computer to be administered."

There's a however a flag in the Windows registry that allows to change the default behavior. Just launch the registry editor of the target PC and open the following key:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\system

Then create DWORD value LocalAccountTokenFilterPolicy and set it to one (1). So you will be able to connect to the admin share.

2.2.1.3 Import hashes from binary files

Import Wizard		×		
		ort from registry or Active Directory files		
Local computer	It is highly recommended that you supply the program with SECURITY and SOFTWARE registry files as well. This will allow to recover plaintext passwords to certain accounts.			
Bemote machine	SAM Active	e Directory DCC Windows PIN Cloud Cache		
	Import regular accounts (from SAM registry file)			
Binary files	SAM	C:\Windows\System32\config\SAM		
1	SYSTEM	C:\Windows\System32\config\SYSTEM		
Textual or	V Load history hashes			
project files	✓ Load hashes of machine accounts ✓ Scan system for plaintext passwords			
	Process deleted and hidden accounts, if any			
System restor				
		Import Cancel		

Import hashes from binary files. Windows Password Recovery can extract password hashes directly from binary files. Even those of them that are currently used by the system (i.e. locked).

Normally, password hashes are stored in SAM or SECURITY registry files, which reside in the '% WINDOWS%\System32\Config' folder. The same folder contains the SYSTEM registry, which is necessary for the recovery. If you have specified path to the registry in the current system, parsing it will take a bit longer (normally by a few seconds).

Password hashes for domain accounts are stored in the Active Directory database; or, to be more specific, in the very heart of it, in the ntds.dit file, which resides in the folder: '%Windows%\ntds'. The recovery of domain accounts also requires the SYSTEM registry file. Be careful! Dumping from the current system's AD database may take some time, especially when ntds.dit is of a considerable size.

The program works properly and supports all the SYSKEY encryption options: Registry SYSKEY, SYSKEY startup diskette, SYSKEY startup password.

If you are copying the files from other system, besides the SAM (ntds.dit) and SYSTEM files, it is also highly recommended to copy the SECURITY and SOFTWARE registry files. They should be located in the same folder with the SYSTEM file. That would allow you to recover the passwords to some user accounts quicker.

Importing Windows PINs requires full access to the Windows folder and read-only access to the users' profiles directory (typically, C:\Users).

To load Microsoft or Azure AD cached passwords, it's sufficient to show a path to the Windows directory.

Using additional options you can:

- Turn on/off loading history hashes. Turning off history loading will increase database parsing. From the other hand, when processing (attacking) hashes, guessing history passwords may give a clue to figure out the password for the primary account the hashes belong to.
- Discard loading machine accounts (ones end up with \$ character).
- Switch on/off instant check for plaintext passwords, if any.

2.2.1.4 Import from project/text files

Import Wizard		\times
	Import from project/text files	
Local computer	Select a proper format and a file you want to import hashes from	
Remote machine		
	File format: PWDUMP files (*.txt, *.pwdump)	-
	File name D:\Downloads\1.pwdump	
Binary files		
Textual or project files		
8		
System restor		
	Import Cancel	

Finally, you can load the hashes to your project by **importing them from other applications**. The software supports the following formats:

LM/NTLM hashes

- **PWDUMP** despite many disadvantages, this is a de facto standard format for storing password hashes. Important note: This format does not fully support national characters. Therefore, some user names or comments may not display correctly. Windows Password Recovery also supports textual PWDUMP files in UNICODE.
- LophtCrack (*.lcs) this file format is used by the LophtCrack software. The program supports all versions of LCS files, beginning with v4.
- **Project files *.hdt**, which are used by Proactive Password Auditor (used to be PWSEX) from ElcomSoft. Also supports all versions of the format, beginning with v3.
- *.hsh files, which are exported by Proactive System Password Recovery from the same notorious company.

- Hash lists *.lst, created by Cain & Abel. Windows Password Recovery supports lst files beginning with v.4.9.12. The earlier versions of LST files used the ';' delimiter instead of 'TAB'. Unfortunately, the LST file does not have a marker that specifies the version; therefore, if the LST file is unreadable, you may have to manually replace all the field delimiters with the 'TAB'.
- *.winpsw files, created by WinPassword, from good old LastBit. Supports all versions of WINPSW, beginning with v6.
- **SamInside project files (*.hashes)**. This format is similar to text PWDUMP, but it is more flexible and uses the 0 7f character instead of colon, which is more reasonable.
- **PasswordPro project files (*.hashes)**. This format is similar to text PWDUMP, except several changes. It is used by PasswordsPro product.
- Passcape Universal Configuration Files (*.puc). This container is used in <u>Reset Windows</u> <u>Password</u> software and can contain several different dumps.
- Plain hashes (*.*). Raw hashes in plain text format (32 or 16 characters on a line).

Domain cached credentials

- **Passcape export/import files (*.peif)**. This format holds domain cached credentials and is used by Passcape software. For example, in Network Password Recovery.
- Elcomsoft PSPR files (*.dcc). Textual files that contain domain cached credentials.
- CACHEDUMP files (*.txt or *.cachedump). A standard placeholder for domain cached credentials type 1 hashes. This format is obsolete.
- John The Ripper DCC2 files (*.txt). A DCC2 hash format used in John The Ripper.

Windows Hello PIN

- Hashcat/Elcomsoft/JtR pin hashes. Textual files containing Windows PIN hashes. Usually with *.pin or *.txt extension. Be careful, this format contains private keys that can be used to compromise the owner's security.
- All possible hashes
 - **Password backup files (*.zip)** created by Reset Windows Password. The password backups created by a free edition of the <u>RWP backup tool</u>. If the program detects that the backup is protected, you will be prompted to type in a password in order to be able to unlock it.

After importing hashes, the program automatically marks all the LM or NT hashes and launches the preliminary attack. This action is optional and can be disabled in the general settings. This option is enabled by default.

2.2.1.5 Importing hashes from system restore folders

Import Wizard	;	×
Remote machine	Select a hard disk from the drop down list. The program will scan the disk for available system restore folders. You should then select one to import data from. Note, this feature applicable for SAM hashes only.	
Textual or project files	Hard disk drive D:\- fixed, NTFS System backup files System repair folder System restore points System shadow copies	
	Import Cancel	

Yet another, not a less helpful option is **importing hashes from the system restore folders**. All you would need for that is to specify the path to one of the disks. The program will automatically find the recovery folders and, if it finds the necessary files, import the hashes.

The search is performed, first of all, in the system directory. Second, in '%Windows%\Repair' folder, which normally contains system registry backups. Third, in the 'System Volume Information' folder, which is used for undoing changes made to the system. This technology has been available since Windows XP and is also known as System Restore (XP) or Shadow Coping (Vista+).

Be careful though, the registry backups may contain obsolete data!

2.2.2 Export

All project hashes, along with the settings, are stored in the project file (*.wpr); however, for the sake of greater flexibility and compatibility with other software, the program can export hashes to a PWDUMP or POT files. If **'Export to POT'** is chosen, all found passwords along with corresponding password hashes (except LM ones) will be saved to file in the following format: *hash:password*

You can alter the default output format by holding down *SHIFT* key when clicking **'Export to POT'**. In that case, output format should look like this: *user (rid):password*

The passwords are UTF8 encoded.

2.2.3 New

Saves current project and creates a new one.

2.2.4 Open

Loads/opens a new project. The application's projects have the *.wpr extension and contain program settings and hashes. However, for speeding up the search speed, the program stores the current state of the attack in a separate file progress.ini.

2.2.5 Save

Saves current project. It is recommended to save critical projects from time to time.

2.2.6 Save as

Saves current project under a different name (renames it).

2.2.7 Close

Closes current project.

2.2.8 Hash Import Wizard

The hash import wizard helps to load Windows hashes without burdening you with unnecessary options and questions. For example, to import hashes from another operating system, all you need is to specify the source Windows directory. The rest stuff will be done automatically. WPR will determine the type of hashes (local or Active Directory), turn on plaintext password search mode on the target disk, load deleted/disables accounts, history hashes, biometric data, etc.

There are 4 general data sources to choose from: a local computer, a foreign operating system, a remote PC, and a hash file created in another program.

Hash import Wizard		×
Choose the source from which to retrieve the Windows hashes	Step 1/3	
Data source		
The local machine Extracting hashes (for all user accounts) that are stored locally. If this is a domain PC, the program counts including ones reside in hash history. You may need to disable your the program work as expected.	ram will load AV to make	
Windows directory on an external/attached drive Reading hashes from another Windows system. This OS should be located on a drive that was the local machine via USB/SCSI/SATA/IDE interface.	is attached to	
A remote PC Loading hashes from a remote machine. You will need an Administration account/password to remotely. The remote machine must have remote administration enabled. The program will guint this later.) connect de you through	
A file created by another program. Importing hashes from files that were created in other programs like PWDUMP, HashCat, PPA Cain and Abel, etc.	. L0phcrack,	
<u>N</u> ext >	Cancel	

2.2.9 Hardware Configuration Wizard

It's a good idea to use the Hardware Wizard to configure the hardware that will be used in password recovery. You can choose between auto and manual modes. The first, automatic mode determines the configuration that is appropriate for your computer. If you need to free up some CPU/GPU resources, consider trying the manual mode instead.

Once the automatic mode is enabled, all <u>CPU</u> and <u>GPU</u> settings (in General options) will not be available until you switch back to manual mode.

Hardware configuration Wizard		×
How do you want to set up your hardware?	Step 1/2	
Select the way you want to set up your hardware when running password recovery attacks. If you choose configuration, the program will use all available GPUs (when running a GPU-based attack) and all CPU of CPU-based attacks. Otherwise, you should configure the hardware utilization manually during the next We have been been been been been been been be	e Automatic ores for /izard step.	
Automatic configuration Optimized for best performance when running both CPU- and GPU-based attacks		
Manual configuration Set up your hardware manually		
Next >	Cancel	

2.2.10 Password Recovery Wizard

The recovery wizard uses the best and most up-to-date password search algorithms that have been invented over the past few years. And it's not just simple words. Here are just a few facts:

- The password recovery strength depends on the hardware used.
- To achieve the best result, the program launches different attacks that are optimally matched for searching for different types of passwords.
- The thorough search mode uses both the power of artificial intelligence and algorithms to generate passwords based on found patterns.
- The thorough search mode finds more passwords than any similar program.

Password recovery Wizard	×
Choose recovery type based on predefined settings	Step 1/2
Recovery type	
Quick password recovery	
This method searches for both simple and strong passwords but tries to perform it as fast as possib predefined attacks (depending on your hardware). Takes ~ 10-20 minutes to complete.	le using 6-9
Thorough password recovery	
Thorough search along with some advanced AI technics. Usually takes 1-3 hours. If no GPU were were not detected, some attacks will be either removed or cut to fit into the time frame.	e set or
Custom password recovery	
Custom search based on your own settings or upon a probable knowledge of the password.	
Next >	Cancel

2.3 Recovery menu

This menu item allows selecting and launching an attack. The '*Attack*' pane allows selecting the type of the attack and toggle between attacking LM or NT hashes. Take a note that before actually launching the attack you must have selected/marked the necessary hashes. You can do that through the **Edit-Select** menu. Launching the attack assumes that you have also made all the required settings (on the **Options-Attack Options** menu).

2.3.1 Run

Launches selected attack. When the attack is running, all other items on the menu are disabled. Please note that when the attack is over, the program runs a special mutation and password analysis routine

over the found passwords. This option is enabled by default, but it can be disabled in the general settings.

2.3.2 Continue

Resumes attack from the last stored point. Please remember that the last stored point is automatically erased when changes are made to the attack's options.

2.3.3 Stop

Pauses current attack.

2.4 Edit menu

The Edit menu is available only when the 'Hashes' tab is active; it includes four items: Edit, Copy, Select, and Search.

2.4.1 Edit

Editing a user accou	nt	×
Changing	properties for HomeGroupUser\$	
User name	HomeGroupUser\$	
RID	1003	
LM hash		
NT hash	B02FA1E266F9412CDDEB7B6EF7DEE421	
Description	Built-in account for homegroup access to the computer	•
	Change Cancel	

Selecting this item opens the dialog where you can manually edit the following fields for the selected account: user name, user RID, LM/NT or DCC hashes, plus the comment to the account.

2.4.2 Add

	Add new user account	x
It is obliga	atory to set a name, RID and LM/NT hash	
User name		
RID		
LM hash		
NT hash		
Description]
Or type in a string in F	WDUMP format here	_
test_THAMA:1:8EB9	843E0122DD11AAD3B435B51404EE:881487E14AC8C29D7	
	Add Cancel	

This item allows adding items manually. It allows entering PWDUMP-like strings.

2.4.3 Delete

Deletes entries from list: highlighted (i.e. the one being under the cursor), marked or all at once.

2.4.4 Reset passwords

Drops all found passwords and clears the list.

2.4.5 Copy

Copies current (highlighted) entry to Windows clipboard. Copies only the selected portion of the entry, not the entire entry. For example, user name or the found password.

2.4.6 Select

Selects hashes to be attacked (ones with checkbox option is on). If during the attack the password for the selected hash is found, the checkbox will be automatically cleared, and the record will be marked green. To select the NT hashes, you must first have deselected the LM hashes, and the other way around.

2.4.7 Search

	Find	User name	x	5
Find what:	k.tpraspberry			
Match <u>w</u> hole :	string only	Direction	Cancel	

When the number of entries exceeds a hundred of thousands, finding a specific entry often takes quite a bit of an effort. To make the job easier, the program offers the search of two types: searching a specific field - e.g., user name - and quick-searching of serial entries. In the latter case, the program scans the entire entry, character by character.

2.5 Reports Menu

You can create, print or save one of the program's reports here. The following reports are available:

- Password reports
- <u>Attack statistics</u>
- <u>Miscellaneous statistics</u>
- <u>Account statistics</u>
- Password-list analysis
- Group information



2.5.1 Password reports

The following reports are available here:

- · Password risk status displays empty, found, and not recovered passwords
- Password complexity reports the number of passwords and various character sets being audited
- Password length distribution shows overall length of the broken passwords
- Password uniqueness this report shows unique against reused passwords chart.
- Top reused passwords displays top 20 of the most popular passwords.
- LM vs NT reports the number of LM and NT hashes

- **Password types** loaded items ranged by hash type: regular (NTLM LM), Domain Cached Credentials (both type1 and type2), Windows Hello PIN, cloud cache (Microsoft and Azure Ad accounts)
- Regular vs history passwords reports the number of common and history passwords (only for hashes imported from SAM\NTDS.DIT files; eg. imported from a local computer)
- **Password recovery time** time took to crack a certain password(s). Most vulnerable passwords are marked in red palette.
- Recovered vs unbroken passwords displays the number of discovered and not-found passwords
- Passwords found shows a bit detailed report on found passwords



Password Length Distribution

2.5.2 Attack statistics

Attack statistics includes the following items:

- Preferred attack statistics on number and type of used attacks.
- Attack time analysis of time spent on each attack.
- Attack efficiency1 efficiency analysis: time spent vs. passwords found during attack ratio.
- Attack efficiency2 efficiency analysis: overall efficiency for each attack.



2.5.3 Miscellaneous statistics

Some additional stuff like:

- CPU speed password recovery speed comparison (for brute-force attack).
- **GPU speed** shows and compares password recovery speed for your GPU device. You can benchmark your CPU or GPU performance using the <u>Pass-o-meter tool</u>.
- **Cracked users** displays the number of cracked users. The full list of cracked user accounts can be saved to text file additionally.
- Cracked users and passwords displays the list of cracked accounts with passwords.



2.5.4 Account statistics

Account statistic are available for both local and domain accounts. To generate a report, first select the data source: local or external database, SAM or Active Directory. These are the reports available in this category:

- Regular vs. disabled accounts. This report shows the ratio of regular vs. disabled user accounts.
- Regular vs. locked accounts. Ratio of regular vs. blocked locked accounts.
- With/without password. Shows the number of accounts with blank and set passwords.
- User vs. machine accounts. Ratio of user vs. system accounts.
- Active vs. expired passwords. Report with statistics on accounts with active vs. expired passwords.
- Regular vs never expired passwords compares regular user accounts against those with 'Password never expires' flag or unlimited password live date set.
- Administrators vs. limited users. This report gives comparative statistics on accounts with administrative rights vs. restricted user accounts.
- Account types shows how much machine, user, administrator, etc. accounts.
- Account status displays active against disabled accounts. The same as the first report in the list but contains no additional pane on disabled accounts.
- **Top 10 active users.** Report on top 10 most active OS users. The statistics is gathered from the system's internal user logon counter.
- Bad password logons. Top 10 users with the highest rates in the failed logon counter.

Top 10 Active users



- Last 10 failed logons dispays the list of user accounts last tried to logged on unsuccessfully.
- Last 10 changed passwords shows the time of last 10 users who changed their passwords.
- Last 10 logons displays the time of last 10 users who logged on the system successfully.
- Last 10 logoffs the time at which the last 10 accounts logged off.
- Expired soon accounts user accounts that will expire soon.
- Logon activity groups users by time passed since last logon to system.
- Password age groups users by time passed since last password set/change.



Last 10 changed passwords

You can save some additional information to a text file by mouse-clicking a part of the report.

2.5.5 Password-list analysis

Password-list reports display various statistics and perform a deep analysis for input wordlists. As a source wordlists you can use, for example, the list of passwords recovered by the program. You can generate reports for all words of the input list as well as for passwords with a certain length only. The following reports are available here:

- Password length distribution displays the overall length of the password in a given wordlist.
- Password uniqueness this report shows unique against identical passwords chart.
- **Password popularity** displays the most popular passwords and their percentage of the total number of passwords.
- **Password format** statistics on the 20 most popular formats. The password format is defined by a character mask. For example, the DDUUUUDD mask corresponds to passwords consisting of two leading and two trailing digits, with four capital letters in the middle. You can save popular password masks into a file so that you can easily use them in a mask-based attack later.
- Character set exclusivity this report displays the number of passwords consisting of one unique character set and the percentage of these passwords to those consisting of several ones.
- Character set diversity -the percentage ratio of passwords consisting of one, two, or more character sets.
- Character sets lists all charsets the input passwords are made of.
- Character set ordering the most popular password templates corresponding to the character set order. For example, the *digit-string-special* template includes the following passwords: 123password!@#, 10ve*****, and 12monkey^, etc.

- Character frequency statistics on the frequency of characters in the input words. The 20 most frequent characters are displayed.
- Unique characters the 20 least frequent characters.
- Frequently used leading characters statistics on the most frequent combinations of 1 to 3 characters in the beginning of words.
- Frequently used trailing characters statistics on the most frequent combinations of 1 to 5 characters in the end of words.
- Frequent combinations the 20 most frequently used combinations of 4 to 8 characters.



Password popularity (all passwords)

2.5.6 Group information

This section is aimed mainly to analyze various information about Active Directory groups and aliases. Some reports however can be used to display statistics of a local PC by reading information from SAM registry file. The following reports are available here:

- Last 10 created groups. 10 recently created group accounts.
- Last 10 changed groups. 10 recently changed group accounts.
- Group types. This report shows different types group accounts belong to.
- Most populated groups displays top 10 groups with the largest number of users.
- **Sparsely populated groups** displays top 10 groups with the smallest number of users. Groups without users are not displayed here.
- Active vs inactive groups. The program assumes that active groups have at least one member while inactive groups have no users at all.
- Admin vs non-Admin groups shows statistics about Administrator privileges of the groups.
- Last 10 created aliases. 10 recently created alias accounts.

- Last 10 changed aliases. 10 recently modified alias accounts.
- Alias types. This report shows different types alias accounts belong to.
- Most populated aliases displays top 10 aliases with the largest number of users.
- **Sparsely populated aliases** displays top 10 aliases with the smallest number of members. Aliases without users are not displayed.
- Active vs inactive aliases. The program assumes that active aliases have at least one user while inactive aliases have no members at all.
- Admin vs non-Admin aliases shows how many aliases have Administrator privileges.
- **Domain object types** shows information about all found objects in a domain. For example: users, groups, computer accounts, domain trusts, etc.



Domain object types

2.6 Tools menu

The Tools consists of two parts: tools for controlling access to the application and tools for working with passwords.

2.6.1 Program access

Set the	e program access password	x
Restrict	access to the program	
Password	****	
Confirm password	****	
V Hide characters	as I type	
	Set	

If anyone besides you can access your computer or account, you can password-protect the application. In this case, when starting the program, user will be prompted for the password, and the application will fail to continue unless the valid password is supplied.

Access password is required in order to continue	x
Type in the program access password please	
Password I III III IIII IIII IIII IIIII IIIII IIII	
OK Cancel	

2.6.2 Pass-o-meter

Pass-o-meter	×	
Check the quality of your password		
The password quality depends hash) can be cracked within a compared to LM ones. Note! # weak.	on its length and complexity. The most of LM passwords (ones with non-empty LM couple of hours on a modern computer. NT passwords are much harder to decrypt the password is a common word or a phrase, it should be always considered as	
Source password		
Type in your password:	mysecret	
Charset length:	26	
Recovery configuration		
Password type:	LM	
Recovery speed:	609 mln. passwords per second	
Hardware:	GeForce GTX 750 Ti (GPU) Compute	
Password quality		
Password quality:		
Time to crack:	0m:13s	
Share your benchmarks	ОК	

A tool for measuring password strength. During its first start, the program asks you to test your computer's performance. To check the quality of a password:

- Enter the password in the corresponding field.
- Select the hash type: LM or NT. Please remember that beginning with Windows Vista operating systems store passwords as NT hashes by default.
- Select the computer type. 'This computer' indicates your computer's search speed.
- If you want to test the speed of your GPU device, select '*This computer (GPU)*' from '*Hardware*' combo box and click '*Compute*' button. Note, that you can do it from *Reports* menu as well.

The quality of your password, along with the time that would take your computer with the selected configuration to break it will be shown at the bottom. For example, breaking any LM hash of an alphanumeric password would take about 10 minutes on a modern CPU (at the search speed of over 100 mln. passwords per second). The search speed on a GPU can raise by another order of magnitude.

We would be grateful if you let us know the speed you've reached on your PC.

2.6.3 Password Checker

	Password Checker	х
Enter a passwo	ord to check it's hash	
Password:	123	
Status:	Matched !!!	
Current password		
LM hash:	CCF9155E3E7DB453AAD3B435B51404EE	
NT hash:	3DBDE697D71690A769204BEB12283678	
-Hashes to compare with		
LM hash to compare:	CCF9155E3E7DB453AAD3B435B51404EE	
NT hash to compare:	3DBDE697D71690A769204BEB12283678	=
	Remember Cancel	

This tool allows checking the password of a selected hash manually. The tool is often necessary for validating certain hashes. For example, when an LM hash, for one or the other reason, doesn't match the password's NT hash.

2.6.4 Hash Generator

	Hash Generator	x
Single hash gen	erator	
Current password		
Password:	123	
Password hash LM hash:	CCF9155E3E7DB453AAD3B435B51404EE	
NT hash:	3DBDE697D71690A769204BEB12283678	
PWDUMP string sample:	Test_123:1000:CCF9155E3E7DB453AAD3B435B51 📄	1
	Add	

The single-hash generator allows to quickly generate a test entry for a specified passwords and add it to the hash list.
Random passwords generator X		
Multiple hashes generator		
With this tool you can easily create a PWDUMP file with multiple randomly generated hashes.		
Initial charset	A-Z	
Passwords to generate	1000	
Minimal password length	1	
Maximal password length	7	
Generate		

If you want to create a PWDUMP file with a specific number of randomly generated passwords, use the multiple-hash generator. In the new hash dialog, select the minimum and maximum length, character range and the total number of the hashes to be generated.

Dictionary to hash gen	erator
Create a PV	NDUMP-like hash file out of a given word-list
Source word-list	
Dictionary name	c:\english.dic 🗃
Destination (PWDUMF Generate LM hash Generate NTLM hash Maximal number of	?) file options ash lines in the output PWDUMP file
	Generate Cancel

With the dictionary to hash generator you can easily create PWDUMP file out of a given word-list. This tool has a number of additional options here. For example, you can limit the number of output hash items or create PWDUMP file for NTLM hashes only.

2.6.5 Rainbow Tables Generator

Rainbow tables are special search tables used for reversing cryptographic One-Way Functions and cracking plaintext passwords derived from the hash functions. An example of such hashes would be a user password (LM or NTLM hashes) in the Windows OS.

Windows Password Recovery has <u>the password lookup implementation using rainbow tables</u>. The tables it requires can be downloaded off the Internet or created manually with the RT generation tool.

🧿 Rainbow tables generator	
Create your own rainbow tables	
┌ Table options	
Algorithm Min Length Max length Index	Chain length Chain count Table count
	10000 67108864 1
Charset name: alpha-space	•
Character set: ABCDEFGHIJKLMNOPQRSTUVWXYZ	
Table statistics Key space: 10862674479 Success rate: 99.90%	Disk space: 1024.00 Mb
Benchmarks Hash speed: 5.13 Mp/s	Step speed: 1.92 Mp/s
Table precomputation time: 4d 0h:54m:17s	Total precomputation time: 4d 0h:54m:17s
Max cryptanalysis time: 0m:25s	
Output folder: C:\0	
Thread to run: 4	
Benchmark	Start

Before you start generating your own tables, it is important to properly configure the respective related options and find their best combination. First, select one of the two algorithms (LM or NTLM) you need and setup a proper **character set** passwords will be limited to. The wider the character range is, the more passwords will be recovered in the rainbow table attack, but the more time it will take to precompute the tables and, perhaps, of greater size they will be.

Rainbow tables are used to recover passwords up to a certain length you should setup in the 'Min length' and 'Max length' fields. An LM hash in Windows consists of two 7-character halves; therefore, the maximum password length to be used when generating LM tables must not exceed 7.

'Chain Length' affects the following parameters of the table: password recovery rate, table generation time, and time it takes to recover a single password by the attack.

Chain count affects password recovery rate, table generation time, and its size.

Program's interface

Currently, the RT generation tool does not support tables greater than 2 GB in size; however, when creating large tables, you can increase the number of them (**'Table count**' option).

The implementation peculiarity of the rainbow table lookup algorithm is in the fact that the success of the recovery depends on several parameters, which you need to pick the best ratio for, depending on the size of the tables, the time it takes to generate them and the max time it takes to find a password in the rainbow attack.

The table generation tool supports multithreading, so before launching the precomputation you may want to set an appropriate number of simultaneous threads to be run for creating the tables.

2.6.6 Pascape Rainbow Tables Generator

Passcape Rainbow Tables are used for recovering passwords in Passcape table attack. This tool is intended for creating such tables.

Passcape rainbow tables generator	
Create Passcape rainbow tables	
⊂ Table options	
Algorithm Chain length Chain count nt 11000 67108864	Table count 2 Maximize lookup efficiency
Input wordlist: E:\Program Files\Passcape\WPR\dic	:\common.pcd 🚔 🔲 Make perfect rainbow table
Table statistics Key space: 168 013 091 236 Success rate: 99.04%	Disk space: 2048.00 Mb
Benchmarks	
Hash speed: 8.41 Mp/s	Step speed: 4.46 Mp/s
Table precomputation time: 1d 21h:59m:13s	Total precomputation time: 3d 19h:58m:27s
Max cryptanalysis time: 0m:27s	Fingerprints: 409 894
Output folder: G:\PRT\test	
Threads to run: 4	Benchmark
	<u>S</u> tart

Before you start generating tables, you should set a wordlist that will be used for creating a database of word-prints and specify the table parameters:

- **Chain Length**: affects the probability of finding passwords (e.g. success rate), table generation time and time needed to search for a single password during the attack.
- Chain Count: affects the success rate, table generation time and its size.

At the moment, the table generation tool does not support tables of over 2 GB in size. However, you can create several tables if you are working with very large arrays of data (see the '**Table count'** parameter).

Success in recovering a password using the tables depends on several factors, and it's important that you find their best values depending on the size of the tables you work with, their generation time and cryptanalysis time – that is, the time needed for recovering a password during the attack.

Two additional options are used to manipulate table generation efficiency:

- Maximize password lookup efficiency: allows you to generate more wordprints from the source wordlist by adding numbers, keyboard and frequently used combinations. This option works well with small wordlists.
- Make a perfect rainbow table: as you may know, password chains in rainbow tables can merge. It means that there is a waste of information, time and disk space. This option allows you to create the so-called 'perfect tables' with no merged chains. Perfect tables occupy considerably less disk space and make password recovery a bit faster. However, the payoff for these advantages is a lower success rate in password recovery. To compensate for this lower success rate, you should at least double the number of password chains and increase the number of generated tables.

The table generation tool supports multi-threading, so make sure to set the necessary number of concurrent threads to be run by the program prior to starting the process.

2.6.7 Wordlist tools

Rather a scant number of acceptable tools for working with specialized password dictionaries has inspired the developers of this software to create their own toolkit. With this toolkit, you can easily create new and edit existing wordlists, as well as use them with any password recovery applications.

2.6.7.1 Create new wordlist by indexing files

This tool is designed for creating a new wordlist by selecting (indexing) words from local files on your computer. For example, those could be *.html, *.xml, *.txt, *.doc files, as well as *.mdb, *.pdf, *.exe files, etc.

The indexing is based on the **IFilter** technology, which you can read about in <u>Wikipedia</u>. The idea of the technology, developed by Microsoft, comes down to the possibility of indexing the text of any file, which an appropriate IFilter plugin is installed for. This way, you could access the text contained, for example, inside *.exe or *.dll files, e-mail client's database, etc.

Despite the fact that numerous IFIIter plugins, both commercial and free, can be found on the Internet, Windows Password Recovery has internal support for the following types of files:

- Archives: *.zip, *,cab, *.rar, *.7z
- Programs: *.exe, *.dll, *,cpl, *.ocx, *.sys, *.scr, *.drv
- Text: *.txt, *.dic, *.udic, *.utf
- Internet: *.html, *.htm

In other words, files with these extensions can be parsed by the program even without a single IFilter installed on the computer.

Windows 7 has an internal Windows Desktop Search tool, which has a wide range of filters for supporting the majority of popular documents. Under other operating systems, Windows Desktop Search can be installed manually; the setup file can be downloaded from the official website of Microsoft.

Wordlist Tools	×
Set up file indexation opt	tions Step 2/2
The word indexation is based on IFilter search extension. Without an appropriate IFilter, conte some 3d-party IFilters may work incorrectly!	engine. You can index any file if an IFilter was installed for the file ents of a file cannot be parsed and indexed by the program. Be careful,
C Select a folder the files (to be indexed) are lo	cated at
Parse files in the given folder only (in all su	ubfolders otherwise)
D:\3\proza.ru	
O Index all files	
 Index files with the following extension(s) 	only
html,htm,php	
Index all files except the following extension	on(s)
Multithreaded document parser	Enumerate file name only, skip file content
Accept alpha-numeric passwords only	✓ Include phrases
 Limit maximal word size to (characters): 	64
Skip file if its size is greater than (Mb):	100
Use custom word delimiters	
Output wordlist format: UTF8 text file	Ţ
	Next > Cancel

The configuration options for this tool consist of two groups. In the first group, you should specify the path to initial folder, where you need to index the files, and select a file parsing method, namely:

- Parse files in the specified folder only. If this option is not set, the program recursively analyzes all the subfolders and files inside them.
- Index all files
- Index files with certain extensions only
- Index all files except certain extensions

File extensions are to be typed without the dot and to be separated by a comma. Example: txt,dic,xml,chm,htm

The additional options group allows to customize file parsing methods, namely:

- Multithreaded document parser. This option, if set, speeds up the indexation process drastically by utilizing as many CPU cores as your system has.
- Enumerate file name only, skip file content. This option, if set, creates a wordlist out of the found file names. The content of the files will be ignored.
- Accept alpha-numeric passwords only. If set, this option will skip all special characters. Only alpha-numeric passwords will be processed.
- Include phrases. This option also allows putting phrases into destination wordlist. A phrase is considered as a string of characters (of up to 256 symbols) with at least one space character in it.

- Limit maximum word size. It is recommended to always set this option. The best maximum word length in a wordlist is 16-64 characters. Cutting the maximum length sometimes radically speeds up the file parsing process. It wouldn't be worthless to remind that the maximum allowed password length in Windows is 128 characters.
- Skip files with size greater than specified. Some IFilters take very long to parse large files; that can cause the program to "hang".
- Use custom word delimiters. You can set your own word delimiters for parsing files. For example, you could use characters like: !"#\$%&'()*+,-./:;<=>?@{}[]_ and, of course, space.

Clicking the **Next>** button launches the actual indexing, which may take considerable time. For the sake of speeding up the process, the list of words found during the indexing is created and maintained in the computer memory; that requires significant resources. So, if you get a runtime error of lacking the memory, try decreasing the maximum word length or limiting the number of files being parsed and then try running it over again. Once the operation is completed, and the found words are saved to disk, sort them out to get a truly valuable wordlist. Found words are guaranteed to be unique, i.e. they do not contain duplicates.

Be careful, some third-party filters could fail to run properly and cause the application to "hang", fail or abnormally terminate. For example, some filters for parsing PDF in Windows XP are known to generate errors.

2.6.7.2 Merge wordlists

A wordlist merging tool is used when you need to combine two or more wordlists in one.

Wordlist Tools		×	
G Rerge	several wordlists into one	Step 2/2	
Destination dictionary is a wordlist and remove duplic	Destination dictionary is a text unicode-based file, lines delimited with CRLF. Set sort checkbox on to sort the output wordlist and remove duplicate strings. Make sure you have enough space on your destination drive.		
☐ Input wordlists			
Dictionary name	Full path	^	
Somali.udic	E:\SYS\PWC11\Wikipedia (part 3)		
Spanish.udic	E:\SYS\PWC11\Wikipedia (part 3)		
Sundanese.udic	E:\SYS\PWC11\Wikipedia (part 3)	Add folder	
Swahili.udic	E:\SYS\PWC11\Wikipedia (part 3)	Add files	
Swati.udic	E:\SYS\PWC11\Wikipedia (part 3)		
Swedish.udic	E:\SYS\PWC11\Wikipedia (part 3)	Delete selected	
Tagalog.udic	E:\SYS\PWC11\Wikipedia (part 3)		
Tajik.udic	E:\SYS\PWC11\Wikipedia (part 3)	- Clear list	
Output wordlist			
Sort output wordlist	Limit RAM usage (Mb)		
Output wordlist format:	ASCII text file		
	<u>4</u>	lext > Cancel	

If the 'Sort output word-list' option is not set, merging comes down to simply adding new words, without sorting or checking for duplicates. In practice, however, more common is merging with sorting; it ensures that all the words in the output wordlist are alphabetically sorted and duplicate-free.

Sorting may take a considerable amount of memory; therefore, it is appropriate to set a limit for the amount of memory that can be used by the process (at the expense of a little downgrade of the operation speed).

2.6.7.3 Wordlist statistics

Wordlist analyzer gathers and shows the following statistics:

Word-list Tools	×
G Word-list statistics	Step 2/2
Select a word-list you want to view the statistics for and click 'Next' button to start word analysis	
Word-list to view the statistics for	-2
E:\Program Files\Microsoft Visual Studio\passcape\WPR\debug\dic\Australian.pcd	
Analyse also character frequency	
Statistics *****Common information***** File name: Australian.pcd File size: 54 771 Last modified: April 15 2011 - 12:24:05 Word-list type: Passcape Compressed Dictionary Sorted ascending: Yes	× E
*****Word statistics:***** Total words: 79 409 Non-English words: 0 Multi-word phrases: 0 Bytes per word: 0.69 Bits per character 0.55 1 character words: 16 2 character words: 62 3 character words: 348	
<u>N</u> ext >	Cancel

Common information

- Dictionary name
- Size in bytes
- File type
- · Last modified date and time
- Whether or not alphabetically sorted (the check takes place only if the file is sorted ascending)

Word statistics

- Total words
- Non-English words
- Multi-word phrases, i.e. words separated with space
- Bytes per word, less word delimiter. Shows average wordlist compression ratio.
- Bits per character. Shows real wordlist compression ratio. For example, in UNICODE the bits per character value tends to 16 (not counting word delimiter), in regular ASCII wordlists to 8. In certain compressed PCD wordlists one letter can be coded by less than 1 bit (see the screenshot).
- Word statistics how many words consist of 1, 2, 3, etc. characters.

Character frequency analysis (if the respective option is set)

· Indicates how frequently a certain character appears in a wordlist

2.6.7.4 Sort wordlist

The toolkit offers 6 modes of wordlist sorting; 4 of them are common, and 2 are extended. The common sorting modes include sorting wordlists in the alphabetical order (both ascending and descending) and by word length. When sorting alphabetically or by word length, the program automatically removes word duplicates.

Wordlist Tools	×
G Sorting file	Step 2/2
6 types of sort available here: sort by ascending alphabetic string order, sort by descending order, so descending by string length, sort by string length and save results to multiple files associated with the 1.txt, 2.txt, 3.txt etc.) Additionally you can sort input wordlists in alphabetic order and save results to r associated with the first letter of the words. For example, a.txt, b.txt, c.txt z.txt.	rt ascending or string length (eg. nultiple files
Input wordlists	
C:\1\1\wl.txt	6
Sort options	
Sort ascending (and remove duplicates)	
Sort descending (and remove duplicates)	
Sort ascending by string length	
Sort descending by string length	
Sort by string length and save output to multiple files	
Sort in alphabetic order and save output to multiple files (eg. a.bt, b.bt, c.bt)	
□ Ignore case (disregard the case of letters) 32 ↓	
Output wardlist formate LITES text file	
Output wordist format.	
Next >	Cancel

Additionally, you can sort a wordlist by length and save the results in multiple files, associated with word length. For example, file 1.txt would contain 1-character words, 2.txt - two-character, etc.

The sixth sorting mode works similarly. At the same time, the program sorts the source wordlist in the alphabetical order and creates several target wordlists that correspond with the first letter of the word. For example, all words beginning with letter A would be written to file A.txt, words beginning with B - to B.txt, etc. You should keep in mind that certain words may begin with characters that cannot be used in a file name. In this case, the program automatically suggests a replacement by issuing an appropriate warning in the messages window.

If the 'Ignore case' option is set, the sorting is carried out regardless of letter case; i.e., the words *bad*, *Bad* or *BAD* are considered identical, with all the ensuing consequences.

Target wordlist name may be the same as the source; however, that is not recommended.

Sorting large files (supports files larger than 4 GB) involves intensive use of RAM; the amount of it can be limited by the respective option. For large files, it is not recommended to set the memory limit less than 16 MB, as that can affect the speed of sorting.

While sorting, the program may create auxiliary files in the application's temporary folder. Make sure that the disk with the temporary folder has enough room for the swap files.

2.6.7.5 Convert/compress wordlist

Numerous wordlists that can be found on the Internet are usually represented by three major formats: **ASCII**, **UTF16** (Unicode) and **UTF8**. With this tool, you can convert a wordlist from one format to another and optionally compress wordlists to ZIP files. Besides the three above mentioned formats, the program supports its own format **PCD** (Passcape Compressed Dictionary), which, in the majority of cases, gives a greater gain in size even compared to a compressed ZIP archive.

Creating large PCD files may take considerable time!

Word-list Tools	A A	
G Nor	d-list format converter	Step 2/2
Select an input word-list you can do it implicitly.	you want to convert. By default, the program	n detects input file format automatically. However
Input word-lists		
E:\Downloads\fbna	nes\facebook-f.last-withcount.txt\facebook	t-firstlast.pcd 💕
Input file format	Auto-detect	•
Output word-list file f	ormat	
Output file format	PCD (Passcape Compressed Dictionary)	*
Compress output	ile into ZIP archive	
		Next > Cancel

This tool's user interface is pretty easy. In the upper group, select the source wordlist and its format. By default, the program detects the format of the file automatically, but you can also specify it by hand.

While the format of a PCD can be clearly recognized, with text files it's not that easy. As a rule, text files/wordlists in UTF16 or UTF8 begin with a two- or three-byte marker that describes the type of the file. However, there are Unicode wordlists that do not have any identifying markers. For such "hard" cases, you need to set the type of the source file manually. Otherwise, the program, being unable to see an appropriate identifier, improperly recognizes the file as ASCII.

Target wordlist, similarly, is defined by one of the four above mentioned formats. With the compression option set, the program additionally compresses the file to a ZIP archive.

Target wordlist name may be the same as the source; however, that is not recommended.

2.6.7.6 Compare wordlists

Sometimes, it is necessary to determine whether two wordlists are identical. That is what the wordlist comparison tool for.

Word-list Tools
Compare 2 word-lists Step 2/2
Select 2 word-list you want to compare. Keep in mind that binary mode uses byte-by-byte comparison, while string comparison mode is a bit different. For example, you can compare word-lists of different types here (eg. PCD and UTF files).
Word-lists to compare
Word-list 1
C:\Downloads\imdb\unique.txt
Word-list 2
C:\Downloads\imdb\english.pcd
Comparison mode
Ignore case (disregard the case of letters)
Next > Cancel

This tool offers two operating modes:

1. Binary comparison, for comparing files by-byte

 String comparison, which compares words rather than bytes. This mode is noteworthy for its ability to compare wordlists of different formats. For example, PCD and UNICODE, or UNICODE and ASCII.

If the ignore case option is set (string comparison mode only), then, for example, the words *bad* and *Bad* will be considered identical.

2.6.7.7 Additional operations

The additional tools are designed primarily for editing and tuning up existing wordlists.

Wordlist Tools	x	
G Additional operations for existing w	ordlists Step 2/2	
Select, what do you want to do with your wordlist. Even though it is not recommended, you can overwrite your input wordlist by setting the same output wordlist name as the input one.		
Input wordlist: E:\E\SYS\PWC07\Real Password	ls (part 4)/rp4.utf	
Action		
Convert all words to lower case	Convert all words to upper case	
Capitalize words (eg. bad -> bad) Remove pop-English words	Remove word duplicates	
Remove Findlish words	Cut leading and trailing spaces	
Split wordligt to churke by setting max, size	Split to churke by setting max, number of strings	
Set minimal word length	Set maximal word length	
3	64	
Change line delimiter	Wipe out HTML tags and entities	
CR LF		
Set custom filter based on Hybrid Dictionary attack ru	les	
Output wordlist format: UTF8 text file		
	Next > Cancel	

The tools include the following operations:

- Convert all words in wordlist to lower case. For example, BAD -> bad.
- Convert all words to upper case. For example, Bad -> BAD.
- Capitalize words upper-case first letter, lower-case all others. For example, bad -> Bad.
- Remove word duplicates.
- Remove non-English words.
- Remove words that entirely consist of numbers and/or special characters. For example, 12345, !@#\$%, 08-19-10, etc.

- Remove English words.
- Cut/remove leading and trailing spaces.
- Cut/remove leading and trailing spaces and numbers.
- Cut/remove leading and trailing spaces, numbers and special characters.
- Split wordlist to chunks by maximum size.
- Split wordlist to chunks by maximum word count.
- Remove words of length smaller than specified.
- Remove words of length greater than specified.
- Change line delimiter.
- Wipe out HTML tags and trash. This menu also converts HTML entities to human-readable form.
 For example, <u>&</u> -> <u>&</u>, <u>@</u> -> <u>@</u>
- Set your own filter based on <u>Hybrid Dictionary rules</u>

For source wordlist, the program takes ASCII, UTF16, UTF8 and PCD files. Target wordlist can be a text of ASCII, UTF16 or UTF8.

Source and target wordlist name may be identical (not recommended). In this case, the source wordlist will be overwritten.

2.6.7.8 Index HDD sensitive areas

Creating a wordlist by indexing the hard disk (followed by an attack using this wordlist) is a pretty useful and sophisticated tool for decrypting passwords to local Windows accounts.

Often users, instinctively, set same passwords to their Windows accounts, Web, ICQ, etc. The idea of this tool is to create a wordlist of all found formerly used passwords, user's messages, words from recently opened files, etc. and then use the accumulated wordlist for looking up passwords to the local accounts. This technique is engaged in the Artificial Intelligence attack.

The configuration of the tool conventionally consists of four parts:

Word-list Tools	×
Index words and passwords in HDD sensitive areas	Step 2/5
Select what kind of system passwords do you need to index	
 Enumerate all known system passwords Active Directory plaintext passwords Autologon passwords 	
Image: Cached logon credentials Image: SQL logon passwords Image: SQL logon passwords	
Well known NT plaintext passwords Windows Media passwords MSGINA stored plaintext passwords (Win2K only) RAS_Dialup_VPN_DSL connection passwords	
CRD credentials (WCM backups) Windows Credential Manager passwords Explorer FTP passwords	
 Wireless connection passwords Old IM passwords Far passwords 	
Other Windows passwords Other Windows Other Other	
	Next > Cancel

First, select the system modules to be used when generating the wordlist. These modules find and index the following types of passwords on your computer's hard disk: Active Directory plaintext passwords, startup passwords and cached startup passwords, SQL, IIS, Windows Media, Win2K text passwords, RAS, Dialup, VPN, DSL, WEP, WPA, FTP connection passwords, Windows Credential Manager passwords, Instant Messengers, etc. passwords.

Word-list Tools	×
Index words and passwords in HDD sensitive areas	Step 3/5
Select what kind of browser/email passwords do you need to index	
 Passwords for popular browsers and email clients Safari browser passwords Chrome browser passwords Opera browser passwords Mozilla (Firefox, Flock, K-Meleon, Thunderbird, etc.) passwords Internet Evolorer browser passwords 	
Interfet Explorer browser passwords TheBat! email client passwords Eudora email client passwords Outlook Express email client passwords Outlook email client passwords Outlook email client passwords	
Windows Mail and Windows Live Mail passwords	
<u>1</u>	lext > Cancel

In the second part of the configuration, select the browsers and e-mail clients, passwords from which are also to be found and added to the wordlist being created. The program supports the following major web browsers: Safari, Chrome, Opera, Mozilla-based browsers (Firefox, K-Meleon, Flock, etc.), Internet Explorer. E-mail clients are represented by: TheBat!, Eudora, IncrediMail, Outlook Express, Outlook, Windows Mail, and Windows Live Mail.

Word-list Tools	×
Index words and passwords in HDD sensitive areas	Step 4/5
Indexing all words from email databases (messages, attachments, etc.)	
 TheBat! (mailboxes, messages, attachments) Eudora (mailboxes, messages, attachments) IncrediMail (mailboxes, messages, attachments) Windows Mail (mailboxes, messages, attachments) Mozilla, Firefox, Thunderbird, K-Meleon, etc. (message store) 	
 Opera (messages, mailboxes, dictionaries) Outlook Express (mailboxes, messages, attachments) Outlook (mailboxes, messages, attachments) 	
<u>N</u> ext	t> Cancel

Besides merely gathering passwords, the program can index user's e-mail communication, scanning all found mailboxes, messages, attachments, etc. The hard disk search is performed for all accounts in a system, so the process may take considerable time, especially when the system hosts many users or when e-mail clients' databases are large. One way or the other, you can enable/disable each module individually.

Word-list Tools				
G Index words and passwords in HDD sensitive areas Step 5/5				
Indexing words from recently used files. Click 'Next' to start indexation. The operation may take minutes or even hours to complete, be patient please.				
Index all words from recently used files				
Additional options				
Imit maximal word size to (characters): Skip file if its size is greater than (Mb): 32 10				
Use custom word delimiters				
Don't index (skip) files with the following extention(s)				
rar,7z				
Next > Cancel				

Finally, in the last dialog, you can set the options for indexing words from all files, recently opened by current user. Available options include:

- Set the maximum length of words that can be added to the wordlist. All words with length greater than the specified limit will be skipped.
- Skip files with size greater than specified. The size is specified in MB.
- Use custom word delimiters. By default, word delimiters are all non-alphabetic characters.
- Do not index files with specified extensions. Use this option to skip files that you consider unnecessary.

Clicking the **Next>** button starts the indexing process.

Keep in mind that it can take considerable time!

2.6.7.9 Extract HTML links

This tool is designed for extracting HTML hyperlinks from HTML files.

Wordlist Tools
G I HTML link extractor Step 2/2
Using the link extractor you can easily extract all links out of html files located on your local drive. Just specify the initial directory where to start scanning from. Consider sorting output list to remove duplicates.
Select a folder the files (to be indexed) are located at
Parse files in the given folder only (in all subfolders otherwise)
E:\Sites\domains\hts-cache
O Index all files
Index files with the following extension(s) only Index files with the following extension(s) only
Index all files except the following extension(s)
15,055
Additional options
Type of link
HREFS
Output wordlist format: ASCII text file
Next > Cancel

The configuration options for this tool consist of two groups. In the first group, you should set a path to the initial folder, where the HTML files are located, and select a file parsing method, namely:

- Parse files in the specified folder only. If this option is not set, the program recursively analyzes all the sub-folders and files inside them.
- Index all files
- Index files with certain extensions only
- Index all files except certain extensions

By default, the tool checks *.htm and *.html files only.

The additional options group allows to set the type of links, as well as where to look for them:

- Look in HTML header
- Look in HTML body
- Look links in HREF tag, SRC tag or in both tags.

Clicking the Next> button launches the search, which may take considerable time. Once the operation is completed, and the found links are saved to disk, consider sort them out to get ride of duplicates.

2.7 Utils menu

Utilities menu consists of additional addons aimed mainly for advanced users.

2.7.1 Backup system files

Registry backup tool allows easily create a backup copy of your Windows registry. Even if the registry file is locked by operation system. You can set additional option to save extra space and compress the backup files to ZIP archive.

Backup system file	s X
	Backup registry files
Backup registry Backup Active	You can backup (save) a binary registry file here currently locked by the system. Set compression option to compress the output file into ZIP archive (may take some time to complete).
Directory	
	SAM
	Compress output file (into ZIP archive)
	Backup Cancel

Backing up Active Directory database is much similar to the registry backup, except that the path to Active Directory the program determines automatically.

Backup system file	s X
	Backup Active Directory files
Backup registry	You can backup (save) your Active Directory database here currently locked by the system. Set compression option to compress the output file(s) into ZIP archive (may take some time to complete).
	Compress output file (into ZIP archive)
	Backup

Administrator or Backup Operator privileges are required to run this plug-in.

Creating and saving Active Directory database may take quite some time: minutes or even hours for huge databases.

2.7.2 Asterisk Password Revealer

V Asterisk Password Revealer			
WPR asterisk password revealer			
Q	Run a program (or switch to a window) containing asterisks passwords. Drag the magnifying glass over the **** passwords. If the program is able to reveal the passwords, they will be shown here. Please note, some programs are not supported by WPR (eg. Opera, Firefox, some applications written in Java, etc.)		
l	Check to set this window topmost		
Status			
Mouse position:			
Password available:			
Window title:			
Revealed password			
	OK		

This tool allows to recover passwords hidden behind asterisks. It is often helpful when you need to quickly recall a **** password and don't have the necessary recovery tools handy. In order to get the *** password visible, you should have to drag the magic magnifier from the WPR window to the field with asterisks.

This method works both for Windows controls and Internet Explorer windows. It has a number of restrictions though:

- Some applications have their own GUI, and therefore Asterisks Revealer may be unable to interact with such applications. Those include Opera, Mozilla, Firefox, etc.
- Some websites have a built-in protection, which hides either the garbage or the actual asterisks behind the asterisk characters * (asterisks hidden behind asterisks!).
- In some Windows system dialogs asterisks also hide the * character and not the real password.

To ensure the proper operation of this tool, you are to have the administrator privileges.

2.7.3 Offline Password Remover

A helpful plug-in for removing and modifying passwords directly in the SAM/SECURITY registry file or in NTDS.DIT. For example, to regain access to a locked system, you do not necessarily have to recover the Windows logon password. Instead, you can just copy the SAM and SYSTEM registry files from the unbootable system, use this plug-in to remove the password for the account (or clear the lockup flag) and copy the files back. The password remover plug-in is made out as a wizard and consists of 4 steps:

1. On the first step, select the password source. That could be either SAM file - for the regular accounts, DCC - for domain cached credentials or NTDS.DIT - for removing passwords in a domain.

Offline Password Remover	×
Select the type of password you want to reset	Step 1/4
This powerful utility allows you to reset or change a password for any external (even non-bootable) operating system. Select, what kind of p SAM (regular), domain cached or Active Directory (domain) user acco	Windows account of any assword do you want to reset: unt.
Select password source	
SAM - regular user account	
Active Directory - domain user account	
OCC - domain cached accounts	
	Next > Cancel

2. On the second step of the wizard, specify the path to the SAM, SECURITY or NTDS.DIT file along with the SYSTEM registry. By default, NTDS.DIT is located in c:\windows\ntds. Registry files reside in c: \windows\system32\config.

Offline Password Ren	nover	×
G 🕵 sh	ow path to SAM and SYSTEM registry files	Step 2/4
To reset a regular use example, you can cop place them back.	r password, you should select two registry files: SAM a y the files from another system here, reset the passwor	nd SYSTEM. For rd you need, and then
Select the files copie	ed from another system	
SAM	C:\2\SAM	6
SYSTEM	C:\2\SYSTEM	2
	<u>N</u> ex	d > Cancel

3. On this step, we need to select the account we need to modify the password for. Select the user name and move on to the final step.

ffline Password Remover			×
G Select the user you w	ant to reset th	ne password for	Step 3/4
The top of the page contains the list of us Press 'Next' button to proceed to the final account selected.	er accounts fou Wizard step an	und. Click one of th d reset/change the	em to see its properties. e password for the
User name	User ID	Administrator	Password set
Admin	500	Yes	Yes
Guest	501	No	No
HelpAssistant	1000	No	Yes
SUPPORT_388945a0	1002	No	Yes
Nikita	1003	Yes	Yes
ASPNET	1004	No	Yes
Account properties			
Account locked: No	Accou	unt disabled: No	
Password expired: Never	Passv	vord required: Yes	3
Account description:			
		<u>N</u> e	xt > Cancel

4. The 'New password' field is made for the new password (leave it blank to reset the password). If this field is disabled, it means that the password for that account is already empty. The same applies to the advanced option for unlocking locked or disabled user accounts.

Offline Password Remover	×
Co Keset use	r password and properties Step 4/4
Enter a new password for the Pay special attention to the a locked or disabled.	selected account or set the input field empty to reset the password. dditional option. Windows will decline your password if the account is
User account	
SAM/AD path	C:\2\SAM
User name	Nikita
User ID	1003
Account description:	
Reset user password and i	properties
New password	
Unlock/enable accourt	t (if disabled, locked or expired)
	<< RESET / CHANGE >>
	Finish Cancel

Don't forget to save your SAM, SECURITY or NTDS.DIT files before making the final changes to them!

2.7.4 Forensic tools

2.7.4.1 LSA Secrets Dumper

LSA secrets is a special protected storage for important data used by the Local Security Authority (LSA) in Windows. LSA is designed for managing a system's local security policy, auditing, authenticating, logging users on to the system, storing private data. Users' and system's sensitive data is stored in secrets. Access to all secret data is available to system only. However, as shown below, some programs, in particular Windows Password Recovery, allow to override this restriction.

Windows Password Recovery plugin for handling LSA secrets is a small tool for viewing, analyzing and editing LSA secrets. The plugin's wizard-driven user interface is quite simple and contains of just three steps:

1. First, select the type of secrets you are going to deal with. These can be secrets of the local system, where the application is running, or secrets of an external PC.

LSA Secrets Dumper	×
Select Windows LSA secrets location	Step 1/3
LSA Secrets Dumper is a small tool to display and edit LSA secrets stored in the registry of the currer LSA secrets may store Windows logon passwords in plaintext. Select, what kind of secrets you wan secrets of your local computer or external PC.	nt or external PC. t to view: LSA
Select secrets type	
CLSA secrets of your local computer	
LSA secrets of an external PC	
<u>N</u> ext >	Cancel

2. When selecting secrets of an external PC, you need to specify path to two registry files: SYSTEM and SECURITY. The SECURITY file contains encrypted secrets, and SYSTEM is necessary for decrypting those. You can find out more on encrypting secrets in <u>our article</u>. Please note that encrypting secrets involves SYSKEY. By default, SYSKEY is configured the way that it can be extracted from the registry (that is what SYSTEM is for).

Account Database Key				
Password Startup Requires a password to be entered during system				
start. Pass <u>w</u> ord:				
Confirm:				
 System Generated Password Store Startup Key on Floppy Disk Requires a floppy disk to be inserted during system start. Store Startup Key Locally Stores a key as part of the operating system, and no interaction is required during system start. 				
OK Cancel				

In some cases, it can be configured otherwise: to be either stored on a boot disk or to be derived from user password when the OS starts. One way or the other, the plugin supports all types of SYSKEY encryption.

Data stored in secrets is crucial for the operation of the entire system. Therefore, LSA secrets are stored in two copies: current (active) and previous (former). Modifying a secret places its current copy to the former one and replaces it with the new, modified secret. The plugin has an option for showing both active and former secrets.

LSA Secrets Dumper	Carlot Constant Constant	x
G 🐹 Se	lect secrets of an external PC	Step 2/3
LSA secrets reside in continue.	Windows registry. Please provide SECURITY and SYSTEM bi	inary registry files here in order to
Select secrets sour	ce	
SECURITY	C:\0\Security	<u> </u>
SYSTEM	C:\0\SYSTEM	
	Show expired and out-of-date secrets	
		Next > Cancel

3. The last step of the Wizard decrypts secrets and shows them as a list. To show the value of a secret, just click on its name. Enter the edit mode by double-clicking on one of the characters in the Hex or Ascii field (this marks it in yellow), and enter the new value. In the edit mode, use the cursor keys to move to the next character. Modified values are marked in red. To save changes, right-click on the Hex/Ascii field and then select the save item on the menu that appears.

SA Secrets Dumper				
G View and edit V	Vindows LSA se	crets		Step 3/3
The list below contains found and d HEX/ASCII value to edit it and then modifying or removing some secrets	lecrypted LSA se right-click to sav may cause your	crets. Click one of them to vi re changes. Even though the system to fail or even not reb	ew its value. Dou e secrets are edit poot!	uble-click the able, be careful:
Secret name	Data size	Modified	Expired ^]
0083343a-f925-4ed7-b1d6-d95d	1 0	09-09-2010 14:18:02	Yes	
0083343a-f925-4ed7-b1d6-d95d	1 28	09-09-2010 14:18:03	No	
0083343a-f925-4ed7-b1d6-d95d	1 0	09-09-2010 14:18:02	Yes	Add secret
20ed87e2-3b82-4114-81f9-5e21	9 0	09-09-2010 14:18:03	No	
20ed87e2-3b82-4114-81f9-5e21	9 0	09-09-2010 14:18:03	Yes	Remove secret
DefaultPassword	20	30-09-2010 14:01:36	No	Chara
DefaultPassword	n	30-09-2010 14:01:36	Yes	Share names
Addr	Hex		Asc	:ii
0000 62 00 75 00 67 00	61 00 67 0	0 61 00 6B 00 5F 00	b.u.g.a.	g.a.k
0010 38 00 38 00			8. <mark>1</mark> .	
Undo	All Changes			
Сору	/ Data as Hex			
Сору	Data as Unico	de String		
Save	Changes Perma	anently		
			Finish	Cancel

Keep in mind that certain secrets contain critical data, and modifying them may cause system instability or even impossibility of booting!

The plugin also allows adding and deleting secrets (secrets of current operating system only). Deleting a secret, whether old or new, automatically deletes both its copies.

You can share your secrets with developers (Share Names button). This e-mails only the secret names, without the actual data. Analyzing the secret names will help us make the program more efficient.

2.7.4.2 Domain Cached Credentials Explorer

Windows uses cached domain records to be able to connect to the server even if the logon server is unavailable for whatsoever reason.

The Domain Cached Credentials Explorer allows decrypting DCC hashes in 3 simple steps:

First select the credentials source: cached credentials of the current operating system or ones from another computer.

Domain Cached Credintials Explorer	×
Select domain cached credentials location	Step 1/3
Domain Cached Credentials Explorer allows you to decrypt, view and edit domain cached entries, a domain cached passwords to text file. Select, what kind of credentials you want to display: credenti computer or external PC.	s well as export als of your local
Domain cached credentials location	
Cached credentials of the local computer	
Cached credentials of an external PC	
<u>N</u> ext >	Cancel

Domain cached credentials are stored in the SECURITY registry file. Thus, when selecting the option to read items from an external PC, on the next step of the Wizard, you should specify path to SECURITY and SYSTEM registry files used for decrypting the records. Once the local computer option is chosen, the program should locate those files automatically. The registry files are located at the following folder C:\%WINDIR%\system32\config\, where %WINDIR% is the Windows directory.

Domain Cached Cre	dintials Explorer	- X -
🕒 🌄 c	ached credentials of the local computer	Step 2/3
Domain cahed crede SECURITY/ Cache	entials are stored encrypted in Windows registry, under the 'HKEY_LOCAL_M key. You should provide SECURITY and SYSTEM registry files in order to co	ACHINE/ ontinue.
Domain cached cre	edentials location	
SECURITY	E:\Windows\system32\config\SECURITY	
SYSTEM	E:\Windows\system32\config\SYSTEM	
	Next >	Cancel

If the reading was successful, in the final dialog you will see the decrypted domain credentials. Each record has several attributes. For example, user name, last logon time, group membership, cached user password (actually, hash).

Right-clicking on the list of records opens the context menu, which allows to:

- Save records with all attributes to a text file.
- Export password hashes to a PWDUMP, *.DCC or *.PEIF file. Please note that the PWDUMP format stores records not quite properly; therefore, it is more preferable to store password hashes as *.DCC or *.PEIF files.
- Check or edit the password for a cached domain record.
- Delete record.

Do	Oomain Cached Credintials Explorer Image: Comparing the second state of the secon						
T	he list ecove	t below contains found or the cached password	and decrypted domain cached entri ls, export the list to file and then feed	es. Right-clio I it to Netwo	ck the list to display co k Password Recover	ontext menu. To y Wizard.	
	#	User name	Last logon	User ID	Effective name	Full name	
	0	NetShowServices	April 22 2009 - 09:15:24	1001	NetShowServices	Эта учетная з	
	1	Администратор	April 22 2009 - 09:15:50	500	Длминистратор		
	2 IWAM_W2K September 10 2007 - 09:35 3 u1 September 04 2008 - 17:50 4 user1 September 04 2008 - 17:53			Save list t	o text file shes to CACHEDUM	пис. P file	
				Export to Export ha	PSPR *.dcc file shes to Passcape *.p	peif file	
				Check part Reset/cha	ssword nge password		
	•			Remove e	ntry	Þ	
					Finish	Cancel	

2.7.4.3 Active Directory Explorer

Active Directory Explorer is a small utility for viewing, analyzing and editing properties (attributes) of domain accounts, both public and private.

In the beginning, select the type of the AD database you are going to work with: local or external.

Active Directory Explorer	×
Selecting Active Directory source	Step 1/4
Active Directory Explorer can help you in viewing/investigating both public and private properties accounts	of your domain
Read more information about Active Directory Explorer	
Active Directory location	
Active Directory of the local computer	
Active Directory of an external PC	
<u>N</u> ext >	Cancel

When selecting the external database, specify the path to the **NTDS.DIT** file and to the **SYSTEM** registry. The latter is required for decrypting private data. If the automatic decryption is enabled, all the encrypted attributes of an account will be decrypted on the fly. In any case, the editor allows editing both decrypted and raw data. For safety reasons, the editor mode is available for external databases only!

You should also specify what object you want to display. There are 10 types of domain objects. See the table below.

Domain object	Description
User object	An object of class user. A user object is a security principal object; the principal is a person or service entity running on the computer. The shared secret allows the person or service entity to authenticate itself.
Global domain object	Represents a typical domain object that do not conform to other types.
Computer accounts	Represents a computer object that is associated with individual client or server machines in an Active Directory domain.
Domain trusts	Represents a user object that is used for domain trusts. A trusted domain is a domain that is trusted to make authentication decisions for security principals in that domain.
Alias objects	A security or distribution group that can contain universal groups, global groups, other domain local groups from its own domain, and accounts from any domain in the forest.

Domain object Description

	Aliases can be granted rights and permissions on resources that reside only in the same domain where the domain local group is located.
Aliases which	Represents an alias object that is not used for authorization context generation.
are not used	
for	
authorization	
Group objects	A database object that represents a collection of user and group objects and has a
	security identifier (SID) value.
Groups which	Represents a group object that is not used for authorization context generation.
are not used	
for	
authorization	
Application-	An application-defined group.
defined groups	
Query groups	An application-defined group whose members are determined by the results of a query.

Active Directory Expl	orer		x			
G Sh	ow path to AD databa	ase and SYSTEM registry files Step 2/4	4			
If the auto decryption undecrypted data will	If the auto decryption option is set, the program will try to decode all private properties automatically. Otherwise, raw undecrypted data will be shown. In any case, the editor allows modifying both decrypted and raw data.					
Active Directory loc	ation		_			
NTDS.DIT	C:\Passcape\1\AD\2\r	ntds.dit 🛛 🚰				
SYSTEM	C:\Passcape\1\AD\2\S	SYSTEM 📴				
	Decrypt private data automatically					
⊂ What to show ——			_			
User objects		⊙ Global domain objects				
Computer acco	unts	O Domain trusts				
 Alias objects 		Aliases which are not used for authorization				
Group objects		Groups which are not used for authorization				
Application-defi	ned groups	Groups whose members are determined by a query				
		Next > Can	cel			

Once the data source is selected, move on to selecting accounts. Some Active Directory databases contain tens or even hundreds of thousands of domain records. Reading such large databases and

completing the list of users may take some time. Selecting just one record shows brief information on it at the bottom: status, whether a password is set and whether it is expired, account description, etc. Clicking the '*Next* >' button launches the process of gathering and decrypting all available attributes for the selected object.

Act	Active Directory Explorer							
)	Select the user account whose p	roperties you	want to explor	re 📕	Step 3/4		
T Y	The top of the page contains the list of user/group/alias items found. Click one of them to see its properties. Press 'Next' button to proceed to the final Wizard step and view or edit the selected item attributes.							
	User list -	User name	User ID	Administrator	Password set			
	2	<all></all>	<all> 🔎</all>	<all></all>	<all></all>			
	1	Administrator	500	Yes	Yes			
	2	Guest	501	No	No	=		
	3	TsInternetUser	1000	No	Yes			
	4	IUSR_HOME2K-1R2LZLKY	1001	No	Yes			
	5	IWAM_HOME2K-1R2LZLKY	1002	No	Yes			
	6	krbtgt	502	No	Yes			
	7	testuser	1109	Yes	Yes	-		
	Account properties Account locked: No Account disabled: No Password expired: Never Password required: Yes Account description: Built-in account for administering the computer/domain							
					<u>N</u> ext >	Cancel		

Each attribute consists of a name and a value. For example, '**Common-Name'** contains the account name, and '**Unicode-Pwd**' attribute stores its password hash. For a more detailed description of an attribute, select it on the list and then click on the link that appears on the description field. Double-clicking on the data field opens the selected attribute for editing. When done editing, right-click on the text to open the context menu and then save the changes to the ntds.dit file or discard them.

Here is the description of some account attributes. The complete description is available on the website of Microsoft.

Common-Name

The name of the account.

DBCS-Pwd

Contains LAN Manager password of the account.

Unicode-Pwd

The password of the user in Windows NT one-way format (OWF). Note that you cannot derive the clear password back from the OWF form of the password.

Lm-Pwd-History

Contains the password history of the user in LAN Manager one-way function format. The attribute is used for compatibility with LAN Manager 2.x clients, Windows 95, and Windows 98.

Nt-Pwd-History

The password history of the user in Windows NT OWF format.

Primary-Group-ID

Relative identifier (RID) for the primary group of the user. This is Domain Users group, by default.

Bad-Pwd-Count

Contains the number of times the user tried to log on to the account using an incorrect password.

Admin-Count

Indicates that the account is a member of one of the Administrative groups (directly or transitively).

Logon-Hours

The hours that the user is allowed to logon to the domain.

Last-Logon

The last time the user logged on to the account.

Bad-Password-Time

The last time the user attempted to log on to the account with an invalid password. This value is stored as a large 8-byte integer that represents the number of 100 nanosecond intervals since January 1, 1601 (UTC).

Last-Logon-Timestamp

This is the time that the user last logged into the domain.

Pwd-Last-Set

The date when the password for this account was last changed.

Account-Expires

Supplemental-Credentials

Stores the encrypted version of the user's password. Used in authentication.

User-Account-Control

Flags that control the behavior of the user account. This value can be a combination of one or more of the following values.

0x00000001 Logon script is executed for the account. 0x00000002 The account is disabled. 0x00000008 Home directory is required. 0x00000010 The account is currently locked out. 0x00000020 No password is required. 0x00000040 The user cannot change the password. 0x00000080 The cleartext password is to be persisted 0x00000100 This is an account for users whose primary account is in another domain.
0x00000200 This is a default account type that represents a typical user.

0x00000800 Trust account for a system domain that trusts other domains.

0x00001000 This is a computer account for a computer that is a member of this domain.

0x00002000 This is a computer account for a system backup domain controller that is a member of this domain.

0x00010000 The password for this account will never expire.

0x00020000 This is an MNS logon account.

0x00040000 The user must log on using a smart card.

0x00080000 The account, under which a service runs, is trusted for Kerberos delegation.

0x00100000 The security context of the user will not be delegated to a service even if the service account is set as trusted for Kerberos delegation.

0x00200000 Restrict this principal to use only Data Encryption Standard (DES) encryption types for keys.

0x00400000 This account does not require Kerberos pre-authentication for logon.

0x00800000 The user password has expired.

0x01000000 The account is enabled for delegation. Enables a service running under the account to assume a client identity and authenticate as that user to other remote servers on the network. 0x04000000 The object is a read-only domain controller (RODC)

2.7.4.4 SAM Explorer

SAM Explorer allows you to view, analyze and edit the properties and statistics of Windows user accounts. SAM, which is short for **Security Account Manager**, is an RPC server, which manages Windows accounts database and stores passwords and private user data, groups logical structure of accounts, configures security policy (e.g., password or account lockout policy), gathers statistics (last logon time, logon count, failed logon attempt count, etc.) and controls access to the database. The SAM database is stored in the registry (in the key **HKEY_LOCAL_MACHINE\SAM\SAM**), which is inaccessible to anyone, except the system (even to administrators). On the physical level, the SAM database is a binary registry file with the respective name, located in %WINDIR%\System32\Config, where %WINDIR% is the Windows installation folder.

In the beginning, the Wizard prompts you to select the type of the SAM database: local or external. Please note: if you select a local database, for safety reasons, the editor will not be available, and the database will open in the read-only mode.

SAM Explorer	
Selecting SAM registry source Step 1/4	כ
SAM Explorer can help you investigating both public and private properties of any regular user account, as well as some attributes and internal structure of your Security Account Manager database.	
Read more information about SAM Explorer	
Active Directory location	
SAM database of the local computer	
SAM database of an external PC	
Next > Cancel	-

If you select the SAM database on an external computer, on the second step of the Wizard, specify the path to the SAM and SYSTEM registries. By default, both the files are located in C: **Windows\System32\Config**. Keep in mind that Windows can providently store copies of the registry files in the backup folders, such as C:\Windows\Repair or C:\Windows\ Config\RegBack.

SAM Explorer		×
G 🐫	Show path to SAM and SYSTEM registry files	Step 2/4
Show the path to y C:\Windows\Syste C:\Windows\Repa	rour SAM and SYSTEM registry files here. Normally, the files reside in em32\Config directory. However Windows can prudently store backup copie: air or C:\Windows\System32\Config\RegBack folders.	s of the files in
Registry files loc	ation	
SAM	C:\1\Windows7\Sam	
SYSTEM	C:\1\Windows7\System	<u></u>
	Decrypt private data automatically	
	<u>N</u> ext	> Cancel

On the third step, move on to selecting the account you need to get the attributes for. Select the user and then click Next.

M Explorer				X
Select the user a	account whose prop	erties you want	to explore	Step 3/4
he top of the page contains the lis utton to proceed to the final Wizan	t of user accounts four d step and view/edit th	nd. Click one of the the account at	nem to see its prop trubutes.	perties. Press 'Next'
User list	1	1	1	1
User name	User ID	Administrator	Password set	
Administrator	500	Yes	No	
Guest	501	No	No	
John	1001	Yes	Yes	
HomeGroupUser\$	1003	No	Yes	
Account properties		Account disable	ad Yes	
Password everyond: Never		Pageword re-	radi Vez	
rassword expired: Never		r assword requi	ed. res	
Account description: Built-in acco	ount for administering	the computer/do	main	
			<u>N</u> e:	xt > Cancel

That gives you the list of attributes for the selected account. Selecting a certain attribute on the list shows the data common to that attribute at the bottom of the editor. To open it for editing, double-click on the data field; upon completion, select the save changes item on the context menu.

SAM Explore	er				×	
0	Viet	w/edit attri	butes for selected account	Step 4	/4	
The list belo data it cont	ow shows fo ains. Right	ound and de -click the da	ecrypted attributes for selected user account ata field to view context menu. Double-click t	. Click one of the attribute to view the data field to enter edit mode.	v the	
Attribute n	name	Data size	Attribute description		-	
LMHash NTHash	LMHash 0 LM password hash associated with the user account NTHash 16 NTLM password hash associated with the user account					
IMHistory	ory⊓as /Hashes	0	IM password history bashed of the user a	count		
UserHint	ridarica	10	User hint (displayed during unsuccessful lo	aon)		
•					τ •	
Addr			Hex	Ascii		
0000	<mark>53</mark> 00 74	00	Copy Data as Hex Cope Data as Ascii String	<mark>8</mark> .t.d.1.2.		
			Lopy Data as Date Copy Data as Unicode String			
		L	Jndo All Changes Gave Changes Permanently	Finish Car	ncel	

Description of SAM account attributes.

DataRevision

32-bit unsigned interger that stores version of the data structure. It is divided into 2 WORDs: version major and version minor.

LastLogon

A 64-bit value, equivalent to a FILETIME, indicating the time at which the account last logged on.

LastLogoff

A 64-bit value, equivalent to a FILETIME, indicating the time at which the account last logged off.

PasswordLastSet

A 64-bit value, equivalent to a FILETIME, indicating the time at which a password was last updated.

AccountExpires

A 64-bit value, equivalent to a FILETIME, indicating the time at which an account is no longer permitted to log on.

LastBadPasswordTime

A 64-bit value, equivalent to a FILETIME, indicating the time at which an account last tried to logged on unsuccessfully.

UserID

A 32-bit unsigned integer representing the RID of the account.

PrimaryGroupId

A 32-bit unsigned integer indicating the primary group ID of the acount.

UserAccountControl

A 32-bit flag specifying characteristics of the account.

CountryCode

A 16-bit unsigned integer indicating a country preference specific to this user. The space of values is the international country calling code. For example, the country code of the United Kingdom, in decimal notation, is 44.

CodePage

A 16-bit unsigned integer indicating a code page preference specific to this user object. The space of values is the Microsoft code page designation.

BadPasswordCount

A 16-bit unsigned integer indicating the number of bad password attempts.

LogonCount

A 16-bit unsigned integer indicating the number of times that the user account has been authenticated.

AdminCount

A 16-bit unsigned integer indicating that the account is a member of one of the administrative groups (directly or transitively).

OperatorCount

A 16-bit unsigned integer indicating that the account is a member of the Operators group.

UserName

Unicode string that specifies the name of the user account.

FullName

Unicode string that contains the full name of the user.

AdminComment

Administrator comment associated with the user account.

UserComment

Second user comment associated with the user account.

Parameters

Extended user parameters. Microsoft products use this member to store user configuration information.

HomeDirectory

Unicode string specifying the path of the home directory for the user account.

HomeDirectoryDrive

Specifies the drive letter to assign to the user's home directory for logon purposes.

ScriptPath

Unicode string specifying the path for the user's logon script file. The script file can be a .CMD file, an .EXE file, or a .BAT file.

ProfilePath

Unicode string that specifies a path to the user's profile.

WorkStations

Unicode string that contains the names (separated by commas) of workstations from which the user can log on. Up to eight workstations can be specified. The account flag UF_ACCOUNTDISABLE allows to disable logons from all workstations to this account.

LogonHours

21-byte bit string that specifies the times during which the user can log on. Each bit represents a unique hour in the week, in Greenwich Mean Time. The first bit is Sunday, 0:00 to 0:59; the second bit is Sunday, 1:00 to 1:59; and so on. Note that bit 0 in word 0 represents Sunday from 0:00 to 0:59 only if you are in the GMT time zone. In all other cases you must adjust the bits according to your time zone offset (for example, GMT minus 8 hours for Pacific Standard Time).

Groups

List of groups to which the user account belongs or does not belong.

LMHash

LM password hash associated with the user account.

NTHash

NTLM password hash associated with the user account.

LMHistoryHashes

LM password history hashed of the user account.

NTHistoryHashes

NTLM password history hashed of the user account.

UserHint

User hint (displayed during unsuccessful logon).

UserPicture

Logon picture associated with the account.

2.7.4.5 DPAPI tools

Starting with Windows 2000, Microsoft began equipping their operating systems with a special data protection interface, **Data Protection Application Programming Interface** (DPAPI). Currently DPAPI is very widely spread and used in many Windows applications and subsystems. For example, in the file encryption system, for storing wireless network passwords, in Microsoft Vault and Credential Manager, Internet Explorer, Outlook, Skype, Google Chrome, etc. This system has become popular among programmers first of all for its simplicity of use, as it consists of just a couple of functions for encrypting and decrypting data: CryptProtectData and CryptUnprotectData. However, despite its apparent simplicity, the technical implementation of DPAPI is quite complicated.

Passcape Software first in the world offers a set of 6 tools for comprehensive analysis and decrypting data encrypted with DPAPI. These utilities allow you to:

- Decrypt DPAPI blobs for any account
- Search DPAPI blobs on disk
- Decrypt DPAPI blobs encrypted under the SYSTEM account (e.g., WiFi passwords)
- Analyze and decrypt user's Master Keys
- Check user's password without dumping hashes from SAM or NTDS.DIT
- Decrypt history hashes of all passwords entered earlier (without using SAM or NTDS.DIT)

2.7.4.5.1 Decrypt DPAPI blob

The decryption of DPAPI blobs consists of four steps of the wizard.

Select DPAPI-encrypted blob file

DPAPI offline decoder		×
E Sel	lect DPAPI blob location	Step 1/4
DPAPI blob is an opa System, Wireless con as Internet Explorer, V passwords and sensit	que data structure that holds encrypted data. Many system components such as nection wizard, Windows Credentials Manager, CardSpace, etc. and popular ap Windows Mail, Outlook, Skype, Google Talk uses DPAPI to securely store their s ive data in DPAPI blobs. You can extract DPAPI blobs using 'blob search' utility.	s Encrypting File plications such secrets,
Read more informatio	on about DPAPI offline decrypter	
Select DPAPI blob	file	
DPAPI blob file	E:\Users\test\AppData\Local\Microsoft\Credentials\DD76822F13ADC704B7	65BC8FA 📄
Windows directory	E:\Windows	٩
	<u>N</u> ext >	Cancel

On the first step, specify the path to the DPAPI blob and Windows directory. It must be said that actual DPAPI objects may be stored in different locations of the operating system; for example, in individual xml files, in the registry, in Active Directory; and in different formats: binary, ASCII, UNICODE. There is a <u>special tool</u> for locating, extracting and saving DPAPI blobs to files. With that utility, for example, you can save all DPAPI blobs from a user's registry to individual files and use them in the program.

Here are storage locations for some DPAPI objects.

- Internet Explorer and Outlook passwords, WiFi passwords (XP only): user's registry, %APPDATA% \ntuser.dat
- Google Chrome: %LOCALAPPDATA%\Google\Chrome
- WiFi passwords (Windows Vista and higher): %PROGRAMDATA% Microsoft Wlansvc
- Network connection passwords (Windows Credential Manager): %LOCALAPPDATA% \Microsoft\Credentials or %APPDATA%\Microsoft\Credentials

Use the finder utility to extract DPAPI data from there.

Select Master Key

DPAPI offline decode	r	×
Set	elect Master Key location	Step 2/4
Master Key is used in For example, EFS ce located at the %APF directory, and %SID folder: %WINDIR%\	n DPAPI as a pimary cypher key to decrypt user protected data and passwords ertificates, WiFi, MSN, Outlook, Internet Explorer, Skype credentials, etc. User N 2DATA%\Microsoft\Protect\%SID%'folder. Where %APPDATA% is the user app % is the textual SID of the user. However SYSTEM Master Key may be located System32\Microsoft\Protect.	(i.e. DPAPI blobs). laster Key file is plication data at the following
Master Key file	F·\ lsers\test\AppData\Boaming\Microsoft\Protect\S-1-5-21-1052517986-2	417873 🧀
Musici Noy no		
CREDHIST file	E:\Users\test\AppData\Roaming\Microsoft\Protect\CREDHIST	<i>i</i>
How do you want	the MasterKey to be decrypted?	
 Decrypt using I 	Domain backup key (no user password is required)	
Decrypt with us	ser credentials	
	<u>N</u> ext >	Cancel

Master Key is a set of 64 random bytes, used as the primary key when decrypting DPAPI blobs. Master Key is encrypted with user's password (or system's password if that is a system Master Key). User's Master Key is always located in **%APPDATA%Microsoft\Protect**. It must be noted that there can be several Master Keys, and only one of them is suitable for decrypting a certain object, the one with the name stored inside the DPAPI blob. When searching for a Master Key, the program may filter out unnecessary names. The folder **%APPDATA%Microsoft\Protect** also contains the **CREDHIST** file, which is optional parameter, and in the majority of cases is not required for the decryption.

Decrypt Master Key

DPAPI offline decoder		×
User/	system credentials needed for successful blob decryption	Step 3/4
You should specify user some DPAPI encrypted thave to provide a path to was was created using e simple binary file with the Additional parameters	SID and user logon password here in order to decrypt the DPAPI encrypted slobs, eg. encrypted using SYSTEM account, require machine credentials. to the SYSTEM and SECURITY registry files. Optional entropy file is required entropy (refer to CryptProtectData API for more information). You should man entropy data and show the program path to the file.	d data. However In this case, you'll d when the blob hually create a
User SID	S-1-5-21-481208688-241787349-1052517986-1001	
 User password 		•
⊖ User PIN		•
O Password-less (extr	act from biometrics, ARSO, cached logons, etc.)	
Entropy file (optional)		
	<u>N</u> ext >	Cancel

At least two parameters must be set in order to decrypt a user Master Key: the user's logon password and his security identifier (SID), which is normally specified in the path to the Master Key or flashed in CREDHIST. One way or the other, Windows Password Recovery calculates user's SID automatically. To decrypt a system Master Key, as it has been said already, setting a password doesn't make sense, as the program retrieves all data necessary for the recovery from SYSTEM and SECURITY registry files. If additional entropy was used when creating the DPAPI blob, you must manually create the binary entropy file and specify the path to it. For example, when encrypting Internet Explorer passwords, the UNICODE-formatted website name is used as entropy.

It is curious that Windows 2000 has a critical vulnerability, which allows decrypting any(!) DPAPI blob on a standalone PC without necessarily specifying user's logon password! I.e. all the data protected with DPAPI are actually vulnerable. This is a major fault in the implementation of DPAPI, which is known to Microsoft; however, other operating systems do not have this drawback. If the **CRYPTPROTECT_LOCAL_MACHINE** flag was set in the CryptProtectData function when protecting data, the decryption of that data is also possible without the user's logon password (for example, wireless network passwords). However, this is a peculiarity of an interface implementation and is not a bug.

Windows Password Recovery starting with version 9.7 utilizes some <u>new vulnerabilities in DPAPI Master</u> <u>Key protection</u> which were detected by our company. Thus to decrypt a Master Key of a domain user, the owner logon password is not necessary any longer.

WPR v11.7 supports for Trusted Boot Auto-Logon feature of Windows 10. If the program detects the Trusted Boot Auto-Logon is set for the user, <u>no logon password is required</u> to decrypt the data.

WPR v15 got a couple of new ways to decrypt a DPAPI blob. You can do it using the owner PIN or even with no PIN or password at all. Once the program detects that the user account is using Windows Hello biometric logon, <u>ARSO</u>, cached passwords, etc. that it uses the data to unlock the protection without need of PIN or password!

Decrypt data

DPAPI off	ine o	leco	der														×
•			Decr	ypte	ed D	PAP	I blo	b									Step 4/4
The list b	The list below contains decoded data of the DPAPI blob file. Right-click the list to display the context menu.																
DPAPI bl Master K	ob file ey file	Ð				E:\w E:\U	in7\8 sers\	D24 John	7934 1 (App	iC92 Dat	2766 ta (Ro	EOF8 Damir	2824 ng (M	443B licros	3A3/ soft≬	AE81 Prote	.004 ect\S-1-5-21-2897855234-39502
Addr								He	2X								Ascii
0000 0010 0020 0030	30 00 02 5A	00 00 00	00 00 00	00 00 00	D8 33 00 40	00 25 00	00 8B 00 65	00 1A 00 00	00 63 00 67	00 D2 00	00 CB 00 61	00 01 00	01 08 00 63	00 00 00	00 00 00 79	00 00 00	0Ø
0050 0060 0070 0080 0090	0040 47 00 65 00 6 Copy Data as Hex 0050 74 00 61 00 7 Copy Data as Hex 0060 45 00 52 00 4 Copy Data as Ascii String 0070 69 00 72 00 7 Copy Data as Unicode String 0080 2D 00 37 00 3 Save to File									t.a.r.g.e.t.=.T. E.R.M.S.R.V./.V. i.r.t.u.a.l.X.P. 7.6.1.4.8							
00A0	69	00	72	00	74	00	75	00	61	00	60	00	58	00	50	00	i.r.t.u.a.l.X.P.

Having all that is necessary, Windows Password Recovery performs the final decryption of the DPAPI blob data, which you can then copy to clipboard or save to file. If the final step of the decryption ends up with an error, it is most likely because you have not set properly or not set at all the additional entropy. For example, Internet Explorer and Vista Ftp Manager uses the source page where the password was entered as entropy. Windows Credential Manager, similarly, uses certain string constants, and so on.

2.7.4.5.2 Analyse DPAPI blob

A DPAPI blob is an opaque binary structure, which contains application's private data encrypted using DPAPI. Many Windows applications and subsystems store passwords, secrets and private data in DPAPI blobs. To create files with DPAPI blobs (for further analysis), use our <u>DPAPI blob look-up utility</u>.

Specify path to DPAPI blob

DPAPI blob analysis		×
) 💽 Se	lect DPAPI blob location	Step 1/2
DPAPI blob is an opa System, Wireless con as Internet Explorer, V passwords and sensit	que data structure that holds encrypted data. Many system components suc nection wizard, Windows Credentials Manager, CardSpace, etc. and popula Windows Mail, Outlook, Skype, Google Talk uses DPAPI to securely store th ive data in DPAPI blobs. You can extract DPAPI blobs using 'blob search' u	ch as Encrypting File ar applications such neir secrets, tility.
Read more information	on about DPAPI blob viewer and analyser	
Select DPAPI blob f	île	
DPAPI blob file	E:\win7\8D247934C92766E0F8282443B3A3AE81.004	<u></u>
	<u>N</u> ext >	Cancel

This is the file that was created by the blob search tool.

And proceed to analyzing data

DPAPI blob analysis	
Co Decoded str	ructure of the DPAPI blob Step 2/2
The list below contains decode	d data of the DPAPI blob file. Right-click the list to display the context menu.
Attribute name	Data
dwVersion	1
guidDefaultProvider	df9d8cd0-1501-11d1-8c7a-00c04fc297eb
guidMasterKey	23fa9ba2-95e7-4c71-ab70-7188ed6a5533
dwFlags	536870912
szDataDescription	Local Credential Data
algCrypt	26128
dwCryptAlgLen	256
pHMACKey	B723B127284D43CA638D333EB4702A2EFA8FC6C799D3613EC93B4F0164D73
algHash	32782
dwHashAlgLen	512
pHMACKey2	9A5998AAC7F283BB4B423381D04B1835405A1FF5FE1A5AD9C90B6E9A26120
oData	1BF16F93EF276C950F3B013EEF3B71ED971D8CC27FB4466CE5E8C87E74D31[
•	4
	Finish Cancel

DPAPI blob is a binary data structure, which consists of the following consecutive attributes:

- dwVersion data structure version. Current data version 1.
- guidDefaultProvider data encryption provider, used in encryption function calls, ensures compatibility of versions and organizes simple cryptological primitives. For example, you can set Blowfish or RC5 as a block cipher. Currently, Windows has the following default crypto provider: df9d8cd0-1501-11d1-8c7a-00c04fc297eb, which corresponds with the following registry key: HKLM\SOFTWARE\Microsoft\Cryptography\Protect\Providers\df9d8cd0-1501-11d1-8c7a-00c04fc297eb

00c04fc297eb.

- guidMasterKey Master Key GUID, which data is encrypted with. To decrypt data within a DPAPI blob, first of all you must decrypt the Master Key with the name set in the binary structure guidMasterKey. Only one Master Key can be attached to a DPAPI blob.
- dwFlags various flags. For example, when the bit 3 is set, it indicates that the decryption of the data is to be carried out under the SYSTEM account. The bit (dwFlags&0x20000000) is set at all times.
- **szDataDescription** data descriptor, which is set by the optional parameter <u>LPCWSTR</u> <u>szDataDescr</u> in the function CryptProtectData.
- **algCrypt** data encryption algorithm. By default, Windows 7 uses AES 256 (which corresponds to 0 6610 in the hexadecimal or 26128 in the decimal notation), Windows XP 3DES, Windows 2000 RC4.
- dwCryptAlgLen key length in the encryption algorithm.
- pHMACKey HMAC key 1.
- **pSalt** salt (optional).

- algHash hashing algorithm. By default, Windows 7 uses SHA 512, Windows XP and Windows 2000 SHA1.
- dwHashAlgLen hash length in the hashing function.
- pHMACKey2 HMAC key 2.
- **pData** actual encrypted data.
- **pSignHash** digital signature for verifying data integrity.

2.7.4.5.3 Search DPAPI blobs

The DPAPI blob search dialog is rather trivial. All you need to specify is the source folder, which the program would search for DPAPI blobs, and the target folder, where found blobs are to be stored. The program searches for both binary and text blobs.

Search DPAPI blo	bs	X
Fill in the	search options to	start scanning for DPAPI blobs
Input directory to sea	arch blobs at	
E:\Users\John		
Search for binary Search for text A Search for text U Output folder to save E:\Win7	r blobs SCII blobs INICODE blobs e found blobs to —	
Verwrite existing	g files (prompt othe	rwise)
Statistics Current file: Data scanned: Blobs found:	541 Mb 7	
		Start

Example of a path, where you can find files, containing <u>binary</u> DPAPI blobs: :\Users\John\AppData\Roaming\Microsoft\Credentials

Example of a path, where you can find files, containing <u>textual</u> DPAPI blobs: C:\ProgramData\Microsoft\Wlansvc

Keep in mind that if you want to search for blobs in current user's registry or in Active Directory database, you should first <u>back up</u> the files to a separate directory.

2.7.4.5.4 Master Key analysis

Master Key is 64 bytes of data, which are used as the primary key when decrypting a DPAPI blob. A user's Master Key is encrypted with the user's logon password.

<u>Set path to Master Key file</u> and specify user SID, which the program normally calculates automatically from the specified path.

DPAPI user Master Key	analysis	X						
Selec	t Master Key location	Step 1/2						
Master Key is used in DI For example, EFS certific located at the "%APPDA directory, and %SID% is folder: %WINDIR%\Syst Read more information	Master Key is used in DPAPI as a pimary cypher key to decrypt user protected data and passwords (i.e. DPAPI blobs). For example, EFS certificates, WiFi, MSN, Outlook, Internet Explorer, Skype credentials, etc. User Master Key file is located at the "%APPDATA%\Microsoft\Protect\%SID%' folder. Where %APPDATA% is the user application data directory, and %SID% is the textual SID of the user. However SYSTEM Master Key may be located at the following folder: %WINDIR%\System32\Microsoft\Protect.							
Select Master Key file								
Master Key file	C:\Passcape\1\Win10\users\test\appdata\roaming\microsoft\F	Protect\S-1-5-21						
User SID	S-1-5-21-3827922096-3704335162-3420645426-1005							
Windows dir	E:\WINDOWS	٦						
		Next > Cancel						

All of that user's Master Keys are located in **%APPDATA%/Microsoft/Protect/%SID%**. For example, C:\Users\John\AppData\Roaming\Microsoft\Protect\S-1-5-21-2897849034-3956381361-16091305341-1001\23ab9bc1-9397-4cb1-ab74-7166ed6a8713

The system's Master Keys are stored in **%SYSTEMDIR%**/Microsoft\Protect folder.

Analyzing Master Key

DPAPI user Master Key analy:	sis	x
Co Decoded st	ructure of the Master Key Step 2	/2
The list below contains decode use a simple wordlist to brutefo	ed entries of the MasterKey file. Right-click the list to display the context menu. Ye rce the initial logon password the Master Key is protected by.	ou can
Attribute name	Data	
dwVersion	2	
szGuid	23fa9ba2-95e7-4c71-ab70-7188ed6a5533	
dwPolicy	5	
dwUserKeySize	176	=
dwVersion	2	
pSalt	9ADDBC11B755D5CB1965E3DE4185CA59	
dwPBKDF2IterationCount	5600	
HMACAlgId	32782	
CryptAlgId	26128	
рКеу	ADA40F37099640B12D4E72123E838B6A22E6F0663E7BC64E34CADEF1	
dwLocalEncKeySize	144	
dwVersion	2	
oSalt	A67DA74D1BD768E4EA2F08203D50FF15	T
	Finish	ncel

Master Key file is a binary structure, which consists of a service header and four slots, namely: the actual user's Master Key, local encryption key (for unprotecting local backup key), local backup key (in Windows 2000) or CREDHIST GUID (in Windows XP and higher) and domain backup key.

The Master Key structure list consists of attribute names (i.e. binary fields) and values that corresponds with them. Each section is uniquely colored:

- field with header attributes
- slot with user's Master Key attributes
- slot with Local Encryption Key attributes
- slot with Local Backup Key or CREDHIST file's GUID attribute
- slot with Domain Backup Key attributes

Now, a little more detail.

Header attributes

- dwVersion Master Key file version.
- szGuid Master Key textual GUID. It normally matches the file name.
- **dwPolicy** various flags. For example, if bit 3 is set, the program uses the SHA1 password hash when decrypting user's password; otherwise, it uses MD4. Thus, in Windows 2000 this bit is always cleared. A set bit 2 tells us that backup is require for the Master Key.

User's Master Key attributes

• dwUserKeySize - current slot length.

- dwVersion data structure version. Version 1 implements only attribute with salt.
- **pSalt** pSalt salt, i.e. 16 random bytes of data, involved in the decryption of the Master Key and preventing data attacks using rainbow tables.
- **dwPBKDF2IterationCount** iterations in the PBKDF2 encryption key generation function.
- **HMACAIgId** hashing algorithm identifier.
- **CryptAlgId** encryption algorithm used.
- **pKey** user's encrypted Master Key.

Local Encryption Key attributes

- dwLocalEncKeySize current slot length.
- **dwVersion** data structure version. Win2K uses only one attribute with salt.
- pSalt salt.
- dwPBKDF2IterationCount iterations in the PBKDF2 encryption key generation function.
- HMACAIgId hashing algorithm identifier.
- CryptAlgId encryption algorithm used.
- pKey encrypted Local Encryption Key, used for decrypting Local Backup Key in Windows 2000.

Local Backup Key attributes (Windows 2000)

- **dwLocalKeySize** current slot length.
- dwVersion data structure version.
- **pSalt** salt.
- **pKey** encrypted Local Backup Key.

CREDHIST file's GUID attributes (Windows XP and higher)

- **dwLocalKeySize** current slot length.
- dwVersion data structure version.
- guidCredHist CREDHIST file binary identifier.

Domain Backup Key attributes

- **dwDomainKeySize** current slot length.
- **dwVersion** data structure version.
- **pSalt** 16 random bytes of data, involved in the decryption of the Master Key and preventing data hacks using rainbow tables.
- dwPBKDF2IterationCount iterations in the PBKDF2 encryption key generation function.
- **HMACAIgId** hashing algorithm identifier.
- **CryptAlgId** encryption algorithm used.
- **pKey** encrypted Domain Backup Key. Its decryption requires the domain controller RSA private key, stored in Active Directory database.

To decrypt user's Master Key, you must know the user logon password or PIN. From the context menu, you can check the password/PIN for that Master Key and even try to guess one using a dictionary. However, don't flatter yourself too much. While in Windows 2000 the search speed is ranged in tens and even hundreds of thousand passwords per second, in Windows 7 the count goes down to single items. See the table below (the speed is measured for a single-core Intel Q8400 2.66GHz CPU).

Operating System	Encryption algorithm	Hash function	PKCS#5 PBKDF2 rounds	Password check speed (p/s)
Windows 2000	RC4	SHA1	1	95000
Windows XP	3DES	SHA1	4000	76
Windows Vista	3DES	SHA1	24000	12
Windows 7	AES256	SHA512	5600	10

Operating System	Encryption algorithm	Hash function	PKCS#5 PBKDF2 rounds	Password check speed (p/s)
Windows 10-11	AES256	SHA512	8000	7

2.7.4.5.5 Dump user credentials history hashes

Due to peculiarities of DPAPI implementation, in order to guarantee the successful decryption of all DPAPI blobs, Windows must store all user's previous passwords in the system. User's password history is located in the following file:

%APPDATA%\Microsoft\Protect\credhist

All user's older passwords (along with certain service data) are stored as pairs of hashes: **SHA1** and **NTLM**. Moreover, in order to decrypt the last pair, you must know the hash of user's current password, to decrypt the previous hashes, you need the last decrypted pair, and so on, along the line.

Windows Password Recovery is the world's first utility, which allows to decrypt password history hashes from CREDHIST files.

To do so, on the first step of the application's wizard, specify the path to your CREDHIST file and Windows directory.

Credentials history d	umper	×			
Se	lect credentials history (CREDHIST) file location	Step 1/2			
CREDHIST is a keyr password, the old pas decrypt the hashes, y %APPDATA%\Micro Windows XP: 'C:\Do Windows 7: 'C:\Use	CREDHIST is a key-ring file that keeps all previous user password hashes. Every time a user changes his or her password, the old password hash is added at the end of this file and then encrypted by the new password. Thus to decrypt the hashes, you'll have to know the current password of the user. CREDHIST file is located at the "%APPDATA%\Microsoft\Protect' folder. For example: Windows XP: 'C:\Documents and Settings\Adminstrator\Application Data\Microsoft\Protect' Windows 7: 'C:\Users\John\App Data\Boaming\Microsoft\Protect'				
Read more information	on about credentials history				
CREDHIST file	C:\Passcape\1\Win10\users\dn\appdata\roaming\microsoft\Protect\CREDHIS	π			
Windows dir	C:\Passcape\1\Win10				
	Next >	Cancel			

Then you can decrypt and save hashes from CREDHIST to a textual PWDUMP-like file, if saving as **NTLM** is selected, or to a plain-text file, if the **SHA1** hash format is selected.

Credentials history dumper		×
Oump DP/	VPI credentials history hashes	Step 2/2
To decrypt all user's previous last password he/she set). Ho allowed. Decrypted hashes ca	nashes stored in CREDHIST, you should know the current p wever partitial dump (for example if you know one of the prev n be saved either in SHA1 or NTLM (PWDUMP compatible)	password of the user (i.e. the vious passwords) is also) format.
Statistics		
CREDHIST key chains:	2	
CREDHIST GUID:	93c85e9c-130e-4ede-9063-576492e41a1d	
Dump options Dump SHA1 hashes Dump NTLM hashes, P	VDUMP compatible output	
Decryption password:		
Use password of the cur	rently logged on user	
	DUMP	
		Finish Cancel

It is important to know that in order to decrypt CREDHIST hashes you must know user's current password. If you are decrypting CREDHIST of a currently logged on user, make sure to set the respective option. In this case, you will not have to enter the decryption password, it will be retrieved from the system cache.

The program supports partial dump of history hashes. That means that if user's current password is unknown, but at least one of the older passwords is available, the program can decrypt the passwords the user used earlier, i.e. before that old password was entered.

Be aware, in Windows 8 and higher OSes the dumped hashes for LiveID, Microsoft or Azure AD accounts are not correspond to those ones derived from the logon password.

2.7.4.5.6 Analyse credential history

CREDHIST is a password history file, made out as a chain, where each link represents user's older password hashes. Each time user changes the password, the old password hash is appended to the file

and encrypted with a new password. Therefore, to decrypt all the hashes in a chain, you must know user's current password.

Along with hashes, the chains store other service data, which is also analyzed by this utility.

Select CREDHIST file

DPAPI credentials his	story analysis	×
Se	lect credentials history (CREDHIST) file location	Step 1/2
CREDHIST is a key- password, the old pa decrypt the hashes, y "%APPDATA%\Micro Windows XP: "C:\Do Windows 7: "C:\Use	ing file that keeps all previous user password hashes. Every time a user chang ssword hash is added at the end of this file and then encrypted by the new par you'll have to know the current password of the user. CREDHIST file is located soft\Protect' folder. For example: yourments and Settings\Adminstrator\Application Data\Microsoft\Protect' rs\John\App Data\Roaming\Microsoft\Protect'	ges his or her ssword. Thus to d at the
Read more information	on about credentials history ile	
CREDHIST file	C:\Passcape\1\Win10\users\test\appdata\roaming\microsoft\Protect\CRE	DHIST 🧀
Windows dir	C:\Passcape\1\Win10	<u>a</u>
	<u>N</u> ext >	Cancel

And proceed to analyzing its content

DPAPI	credentials h	istory analysis				X
0		redentials history a	nalysis			Step 2/2
The l use a	The list below contains parsed entries of the CREDHIST file. Right-click the list to display the context menu. You can use a simple wordlist to bruteforce CREDHIST hashes.					
CRE CRE	DHIST key cha DHIST GUID:	ains: 2 93c85e9c-	130e-4ede-9063-5	76492e41a1d		
#	dwVersion	guidCredHist	dwNextCredSize	dwCredChainType	algHash	dwPbkdf2lterationCoun
1	1	7d3ce967-4271	144	1	32782	5600
2	1	dbbad621-e033	0	1	32782	5600
•						•
					Fir	nish Cancel

On the screenshot, you can see that the CREDHIST identifier is 93c85e9c-130e-4ede-9063-576492e41a1d. This is the identifier (GUID) all user's Master Keys in the context of the data owner are attached to. The number of links in the hash chain is 2.

The list below contains all attributes and their values for each link of our CREDHIST.

Attribute description

- dwVersion data structure version
- guidLink current link unique identifier
- dwNextLinkSize next link size
- dwLinkType link type
- algHash hashing algorithm used when decrypting the link
- dwPbkdf2lterationCount iterations in the PKCS#5 PBKDF2 key generation routine
- dwSidSize owner security descriptor (SID) size
- algCrypt encryption algorithm
- dwShaHashSize SHA1 hash size
- dwNtHashSize NTLM hash size
- pSalt salt used in the encryption
- sidUser data owner SID
- pShaHash SHA1 hash
- pNtHash NTLM hash

To guess the original CREDHIST password or PIN, right-click on the attributes and then select '*Use wordlist to check password...*' on the context menu that appears. You can validate password for both currently selected and all the records. The validation time increases proportionally to the number of the records (i.e. links).

See the original CREDHIST password search speed comparative table. The speed is measured for a single-core Intel Q8400 2.66GHz CPU for default OS configurations.

Operating System	Encryption algorithm	Hash function	PBKDF2 counter	Password check speed (p/s)
Windows XP	3DES	SHA1	4000	76
Windows Vista	3DES	SHA1	24000	12
Windows 7	AES256	SHA512	5600	10
Windows 10-11	AES256	SHA512	8000	7

2.7.4.6 Windows Vault Explorer

What is Windows Vault

Windows Vault is a protected storage for user or system secrets, passwords, network keys, web password and other personal information. Data stored in Windows Vault is structured and represents a set of records belonging to a certain Vault schema (see pic. below).



On the physical level, Vault is a disk-based folder with a set of the following files:

Policy.vpol - set of encryption keys for Vault records (credentials). These keys can be protected using two basic methods: either using DPAPI or using a specific user password. The latter protection method is not used in Windows 8 and currently is not supported by the software.

<GUID>.vsch - Vault schema that contains data description, flags and other system information.

<GUID>.vcrd - Vault credential that stores the original encrypted data associated with a certain schema. The data may consist of and normally consists of several fields. Description of the fields is stored in <GUID>.vsch.

Windows Vault Explorer

Windows Vault Explorer is a utility for offline analyzing and decrypting Vault credentials. The decryption Wizard splits the entire process into the following steps:

- 1. Looking for Vault folder
- 2. Looking for user's or system's Master Key
- 3. Setting registry files and other information necessary for decrypting the Master Key
- 4. Selecting Vault Schema
- 5. Looking for Vault records belonging to selected schema
- 6. Decrypting selected Vault credential

Looking for Vault folder

Windows Vault Explorer	×
Select Vault location	Step 1/6
Vault explorer is a tool to view, decrypt and analyse private data stored in Windows Vault. There are 2 to Windows Vault: user and system. Typical location for the User Vault is C:\Users\ <user>\AppData\Local\Microsoft\Vault\<guid>. The System vault stores its data in C:\Windows\System32\Config\systemprofile\AppData\Local\Microsoft\Vault\<guid> <u>Read more information about Windows Vault</u></guid></guid></user>	ypes of
Select Vault location (a folder with policy.vpol file inside)	-B380-
Next >	Cancel

There are currently two types of Vault storage: system and user. The user Vault storage can be located in the following folders:

<USER_APP_DATA>\Microsoft\Vault\<GUID>

<USER_LOCAL_APP_DATA>\Microsoft\Vault\<GUID>

For example,

:\Users\John\AppData\Local\Microsoft\Vault\18289F5D-9783-43EC-A50D-52DA022B046E :\Users\Helen\AppData\Roaming\Microsoft\Vault\4BF4C442-9B8A-41A0-B380-DD4A704DDB28

The default location of the system Vault storage is: <SYSTEM_APP_DATA>\Microsoft\Vault\<GUID>

<SYSTEM_LOCAL_APP_DATA>\Microsoft\Vault\<GUID> <PROGRAM_DATA>\Microsoft\Vault\<GUID>

For example,

:\Windows\System32\config\systemprofile\AppData\Local\Microsoft\Vault\4BF4C442-9B8A-41A0-B380-DD4A704DDB28

:\Windows\System32\config\systemprofile\AppData\Roaming\Microsoft\Vault\4BF4C442-9B8A-41A0-B380-DD4A704DDB28

C:\ProgramData\Microsoft\Vault\AC658CB4-9126-49BD-B877-31EEDAB3F204

Note that some of the specified folders have the system attribute set on, which makes these folders hidden.

Windows has VaultCmd.exe utility for creating and managing your own Vault storages.

Selecting Master Key

Windows Vault Explo	rer	×			
G 💽 Sel	ect Master Key location	Step 2/6			
Master Key is used in Vault's credentials. Us %APPDATA% is the u should be located in t	Master Key is used in Vault to decrypt its pimary encryption key. While the primary encryption key is used to decrypt Vault's credentials. User Master Key file is located at "%APPDATA%\Microsoft\Protect\%SID%' folder. Where %APPDATA% is the user application data directory, and %SID% is the textual SID of the user. SYSTEM Master Key should be located in the following folder: %WINDIR%\System32\Microsoft\Protect.				
Select Master Key f	ile				
Master Key file	E: \Windows\System32\Microsoft\Protect\S-1-5-18\User\59ccd485-35a	l-4828-bc2·			
CREDHIST file					
	<u>N</u> ext >	Cancel			

Once a certain Vault folder is selected, you need to specify path to the Master Key used in the protection of the Vault encryption keys. The user's Master Key always resides in the folder **%APPDATA %**Microsoft\Protect\%SID%, and the system account's Master Keys are stored in **%SYSTEMDIR% \Microsoft\Protect**. It must be noted that there could be a number of Master Keys, while a specific

object could be decrypted using only one key, the name of which is stored in the Policy.vpol file. When searching for the Master Key, the program can filter out unnecessary names.

Decrypting Master Key

Windows Vault Explorer		×
Ce Decod	ed structure of the Master Key St	ep 3/6
You should specify SID, Io However system encrypte SYSTEM and SECURITY	ogon password and SAM registry file here in order to proceed the user Vault decry d Vaults require machine credentials. In this case, you'll have to provide a path to ′ registry files. re required in order to proceed the data decryption	ption. the
User SID	S-1-5-21-481208688-1052517986-241787349-1001	
 User password 		
⊖ User PIN		•
⊖ Password-less (extra	ct from biometrics, ARSO, cached logons, etc.)	
Windows directory	c:\windows	٩
Entropy file (optional)		
	<u>N</u> ext >	Cancel

To decrypt a user's Master Key, you need to provide at least two parameters: the user's logon password and his security identifier (SID), which is normally included in the path to the Master Key. The program finds user's SID automatically. If that hasn't been done for whatsoever reason, set it up manually. To decrypt the system's Master Key, we don't need to specify the password; the program will extract all the necessary information from the two registry files: **SYSTEM** and **SECURITY**.

Windows Password Recovery starting with version 9.7 utilizes some vulnerabilities in DPAPI Master Key encryption. Thus to decrypt any Vault entry of a domain user, the owner logon password is not needed any longer.

WPR v11.7 supports for <u>Trusted Boot Auto-Logon</u> feature of Windows 10. If the program detects the Trusted Boot Auto-Logon is set for the user, no logon password is required to decrypt the data.

WPR v15 supports for additional decryption methods using Windows Hello PIN or biometrics (password-less recovery).

Selecting Vault Schema

Vindows Vault Explo	prer		×		
🕒 🖳 Se	lect Vault schema		Step 4/6		
Every Vault consists descriptor). You shou	Every Vault consists of credentials (encrypted user data). Each credential belongs to a certain schema (credential descriptor). You should select one of available schemas to process further credentials enumeration.				
Vault information Vault location: Selected schema:	E:\Windows\System32\config\systemprofile\AppDa WinBio Credential Manager Credential Schema	ata\Local\Microsoft\Vault	\4BF4C442-		
Select one of the s	chemas Manager Credential Schema		•		
C-h		C.1			
Schema version:	Winblo Credential Manager Credential Schema	Schema flags:	4		
Schema GUID:	fec87291-14f6-40b6-bd98-7ff245986b26	Credentials:	1		
		<u>N</u> ext >	Cancel		

On the fourth step, if the previous ones passed successfully, the program prompts you to select one of the schemas belonging to our Vault from the dropdown list. Just below the list, we can see the general characteristics of the selected schema: its name, version, GUID, flags, number of attributes and credentials.

Selecting Vault credential

Windows Vault Expl	orer		x					
😋 🗾 si	elect Vault credential	S	Step 5/6					
Select a credential y belongs to a certain back and change th	Select a credential you need to decode. Every Vault consists of credentials (encrypted user data). Each credential belongs to a certain schema (credential descriptor). If you want to enumerate credentials for another schema, step back and change the schema to one you need.							
Vault information								
Vault location:	E:\Windows\System32\config\systempro	ofile\AppData\Local\Microsoft\Vault\	4BF4C442-					
Selected schema:	WinBio Credential Manager Credential Sc	chema						
Selected credenti	al: WinBio CredProv Credential							
Select a credentia	l to decrypt							
WinBio CredProv	Credential							
Cred name:	WinBio CredProv Credential	Cred flags:	4					
Last modified:	September 27 2012 - 15:21:41	Data size:	208					
		<u>N</u> ext >	Cancel					

In a similar manner, select one of the credentials of interest that belongs to the schema we have selected during the previous step.

Decrypting Vault credential

Wi	Windows Vault Explorer																		
	C View decrypted Vault credential											Step 6/6							
T	The hexidecimal list below contains decoded data of the selected Vault credential. Right-click the list to display the context menu.																		
	of it cost																		
s	electe	ed sch	em	na:		1	WinB	io Cr	eder	ntial	Man	ager	Cre	dent	ial So	them	a		
C	reder	ntial file	e:			E	E:\W	indo	ws\S	yste	m32	\con	fig\s	yste	mpro	ofile\	Appi	Data	Local\Microsoft\Vault\4BF4C442
F																			
	Add	r								He	5X								Ascii
	002	0 0:	2 (C3	В9	AC	81	61	73	EB	5D	E9	89	62	E9	03	00	00	.ù¬asë]ébé ▲
	003	0 0:	L (00	00	00	32	00	00	00	57	00	69	00	6E	00	42	00	2W.i.n.B.
	004(0 6	Э (00	6F	00	20	00	43	00	72	00	65	00	64	00	50	00	i.oC.r.e.d.P.
	005	0 73	2 (00	6F	00	76	00	20	00	52	00	65	00	73	00	6F	00	r.o.vR.e.s.o.
	006	0 7!	5 (00	72	00	63	00	65	00	00	00	03	00	00	00	26	00	u.r.c.e&.
	007	0																	
	008	0																	
	009	0																- 0	
	00A(0 00	C (00	00	00	05	00	00	00	08	00	00	00	4A	00	6F	00	J.o.
	00B	0 6	в (00	6E	00	00	00	4A	00	6F	00	68	00	6E	00	2D	00	h.nJ.o.h.n
	0000	0 5) (00	43	00	00	00											P.C
																			
-																			
	Finish Cancel																		

And at last the final step, where you can view the decrypted record, copy it to clipboard or save to file for further analysis. The figure shows decrypted plain-text password (it is clobbered) of the administrator account configured to logon using biometric information (fingerprint).

2.7.4.7 Windows Hello Explorer

Windows Hello is a brand-new biometrics technology that enables users to authenticate to their Windows 10 devices with just a fingerprint, iris scan, facial or voice recognition. Windows Hello is supposed to be more user friendly and secure than using a password.

Windows Password Recovery has a set of utilities to analyze Windows Hello security system. This kit includes three features to <u>extract plaintext passwords saved by Windows Hello system</u>, to decrypt digital identities (for example, users' fingerprints) stored in biometric databases and <u>PIN recovery tool</u>.

2.7.4.7.1 Windows Hello credentials

This feature is aimed to locate and decrypt plaintext credentials stored by Windows Hello. Windows hello uses Next Generation Cryptography (NGC or also called CNG) to protect and store user private data and encryption keys. Even though NGC is very elaborated and sophisticated system (worth mentioning that it uses even some undocumented trick to protect encryption keys and data), Passcape

software was first, just like in case of <u>DPAPI</u>, who managed to create almost identical set of APIs but with support for offline mode. Yes, the tool can be used both for a living system and for any external OS. That makes it easy to work with the program even for a newbie, leaving all the numerous and routine operations to the program.

1 Setting Windows folder

Windows Hello credentials	×
Select Windows directory	Step 1/3
This feature is aimed to locate and decrypt users' plaintext credentials stored by Windows Hello sub both in online (for a living system) and offline modes. Please, specify Windows directory of the target sure the directory has read and access change permissions, otherwise the program will not be able extract some NGC-protected passwords.	system. It works t system. Make to get access and
Read more about Windows Hello recovery	
Select Windows directory	
C:\Windows	
Next >	Cancel

The working mode is detected automatically once a Windows directory is set up. The Windows folder holds files and encryption keys protected from being accessed to even by Administrators. To extract the keys, this folder should permit access change or file write operations. Otherwise, the program will not be able to decrypt passwords encrypted with NGC.

2 Selecting data to decrypt



During this step the program outputs all found but not yet decrypted credentials, keys, PINs. Definitely, Windows Hello was developed by several development teams, because multiple encryption subsystems are used and all personal data are scattered through the whole system. Currently, the program supports for the following data types:

- · Picture passwords
- · Logon plaintext passwords protected with picture passwords
- Biometric credentials
- · Logon plaintext passwords protected with biometric credentials
- Biometric encryption keys
- Plaintext PINs
- PIN history (if this option is set in Windows)
- Logon plaintext passwords, stored in Windows Vault and protected by NGC
- · Logon' plaintext passwords, stored in Windows registry and protected by NGC

To finalize the decryption, just double-click an item in bold.

<u>3 Decrypted credentials</u>

Windows Hello system protects personal data differently. Depending on selected credentials' type you should get a decrypted plaintext password, a pin, a picture password or a private key. Use the buttons at the right to copy or save the decoded information for further analysis.

Windows Hello credentia	lls	×
🚱 💆 Decrypt	ed credentials	Step 3/3
Windows Hello subsystem p either decrypted plaintext p the decoded information for <u>Read more about Window</u>	protects personal data differently. Depending on selected credentia assword, pin, picture password or private key. Use the buttons at th r further analysis. <u>s Hello recovery</u>	ils' type you should get ne right to copy or save
Windows Hello type: Decrypted data type:	Picture passwords User logon password, enrollment data	
System name: User name: User password: Picture Figure 1 Figure 2 Figure 3	Test-PC John C:\ProgramData\Microsoft\Windows\SystemData\S Line (x1=230, y1=810, x2=500, y2=290) Circle (x=470, y=320, r=100) Point (x=970, y=580)	Copy to Clipboard Save to file
	Fin	ish Cancel

Decrypted logon password (hidden here) and picture password.

Windows Hello credentials	5	×			
🚱 🗾 Decrypter	d credentials	Step 3/3			
Windows Hello subsystem protects personal data differently. Depending on selected credentials' type you should get either decrypted plaintext password, pin, picture password or private key. Use the buttons at the right to copy or save the decoded information for further analysis.					
Read more about Windows	Hello recovery				
Windows Hello type:	Plaintext PINs				
Decrypted data type:	User logon password, plaintext PIN code				
System name:	Test-PC				
Ser name:	Mary				
••• User password:	SuperPassword123				
W PIN:	7788				
		Copy to Clipboard			
		Save to file			
	Finish	n Cancel			

Decrypted logon password and PIN code.

Windows Hello credentials		×				
🚱 🗾 Decrypted c	redentials	Step 3/3				
Windows Hello subsystem protects personal data differently. Depending on selected credentials' type you should get either decrypted plaintext password, pin, picture password or private key. Use the buttons at the right to copy or save the decoded information for further analysis.						
Read more about Windows He	<u>llo recovery</u>					
Windows Hello type:	NGC-encrypted credentials (vault)					
Decrypted data type:	User logon password					
System name:	Test-PC					
User password:	SuperPassword123					
		Copy to Clipboard				
		Save to file				
	Finish	Cancel				

Decrypted logon password for user 'Mary'.

Note that plaintext passwords protected with NGC can be decrypted using either a biometric key or a PIN. The program first tries to locate and use biometric keys and if it fails to (for example, the biometric logon has no been set), WPR prompts for PIN in order to be able to decrypt the data.

2.7.4.7.2 Biometric databases

Biometric databases contain digital identities used to authenticate a certain user in Windows Hello system. Those identities are fingerprints, 3d-faces, voice or iris.

Biometric databases	×
Select Windows directory	Step 1/3
Biometric databases contain user identity information used in Windows Hello biometric authentication facial, voice, iris, etc. Please, specify Windows directory of the target system. This could be your cut folder or Windows folder of any offline (external) system.	on like fingerprint, urrent Windows
Read more about Windows Hello recovery	
Select Windows directory	
C:\WINDOWS	
<u>N</u> ext >	Cancel

You should set Windows directory of the target system first. This could be the Windows directory of your current or of any external operating system.

Biometric databases		×
Select biometric database	Step 2/3	
The tree contains a list of found biometric databases used in Windows Hello. To proceed the decry database and click 'Next >'.	rption, select a	
Read more about Windows Hello recovery		
Found credentials		
□ □ Test-PC □ □ Fingerprint □ □ □ <td></td> <td></td>		
Next >	Cance	

To decrypt a database, just double-click it in the list.

Biometric databases		×
📀 🌏 Decrypt auth	nentication information	Step 3/3
The list contains all found and du user in Windows Hello system. Y	ecrypted digital identities (fingerprints, 3d-faces, etc.) used to auther 'ou can save specific digital identity for further analysis.	nticate a certain
Read more about Windows Hel	lo recovery	
Windows Hello type:	Fingerprint	
Biometric database:	4ABE16E8-DDCC-42CA-AB09-D96DC7E3BA39.DAT	
User name	Digital ID	
John	031145A00F0F8FC4201217A306A1151483A5C10	
John	19DE35C00D0A6849400F08E07540140C767DC00	
John	1B5A29E0041DDBA980110857D6C01513D60DA10	
		Cause to file
		Save to file
	Finish	Cancel

The decrypted database contains found and decrypted digital identities, like fingerpints, 3d-faces, etc. For example, if a user has set 3 fingerprints to logon into system using Windows Hello, the fingerprints should be decrypted and outputted right of the user's name. Just like in the picture above.

You can save the digital IDs for further analysis.

Despite the Microsoft assertion of extremely security, the digital IDs are badly protected against substitution (unless they're used with TPM devices) and can be easily migrated or copied from one PC to another. For example, you can create your own fingerprint, copy it to another PC into another user account. Then you can simply logon into the alien account using your own fingerprint. Because of the serious nature of this vulnerability that compromises the whole system security, the digital IDs migrating function was disabled in this version of the program.

2.7.4.7.3 PIN brute-forcer

Windows Hello enables different types of biometric sign-in: fingerprint, iris, facial or voice recognition. When you set up Windows Hello, you're asked to create a PIN first. The PIN is very well protected and is not stored anywhere in Windows 10. However it can easily be decrypted in Windows 8. To guess a lost PIN, you should specify Windows directory of the target system first. This could be your current Windows folder or Windows folder of any external system.
To provide PIN synchronization for all devices, Microsoft keeps the copy of your PIN on its servers (for Microsoft accounts only).

Windows Hello credentials	×
Select Windows directory	Step 1/3
Windows Hello enables different types of biometric sign-in: fingerprint, iris, or facial recognition. Whe Windows Hello, you're asked to create a PIN first. The PIN is very well protected and is not stored i Windows 10. To guess a lost PIN, specify Windows directory of the target system first. This could b Windows folder or Windows folder of any external system. <u>Read more about Windows Hello recovery</u> Select Windows directory	en you set up anywhere in ie your current
C:\WINDOWS	
Next >	Cancel

You should show Windows directory of the target system first. In order to be able to extract a PIN, the Windows directory should allow access change or file write operations. In case you have set your current Windows folder, it is enough to run the program with Administrator privileges.

Windows Hello credentials	×
Select user account	Step 2/3
The tree contains a list of found user accounts that were ever configured to use a PIN. Select a user t final PIN recovery.	to proceed to
Read more about Windows Hello recovery	
User list	
Test-PC Mary John	
Next >	Cancel

On this step, the program shows all found user accounts that have been set to use PIN to logon into the system. Just select a user and proceed to the PIN recovery dialog.

PIN bruteforcer		×
G 🐫	Decrypt selected PIN	Step 3/3
Everything is read 'Start' button to ru	ly to start searching for PIN now. Select either dictionary or brute-force lookup me n PIN recovery. Do know that numerical PINs can easily be recovered using Bru	thod and click the teforce method.
Read more about PIN user: PIN hint: Status:	<u>t Windows Hello recovery</u> John The PIN consists of 6 digits	
	Brute-force attack Dictionary attack Mask attack	
Ru Clic sta	Character set 0123456789 Min length 6 Max length	 6 *
		Cancel

The program supports three recovery methods: brute-force, dictionary, and mask attack. Settings for each of them are rather trivial. In the case of brute force, you will have to set up a character set, minimal and maximal PIN length. For a dictionary attack, set up a dictionary and select the word mutation level you need. To be able to run a mask attack, set up a recovery mask (assumes that you know something about the password).

To run or stop the attack, use the round button to the left of the settings.

In certain cases, the program can detect character set used to create a PIN. If this is your case, the hint appears in the corresponding field.

PIN codes protected with TPM are not supported!

2.7.4.8 Windows Credentials Explorer

The **Windows Credential Manager** is a built-in feature in Windows 7 and later operating systems designed to provide a secure storage facility for personal data and user credentials. It acts as a central repository, allowing users to manage a variety of credentials like usernames, passwords, certificates, and generic credentials with ease and convenience. The Data Protection Application Programming Interface (**DPAPI**) ensures that these credentials are safely protected.

The Credential Manager comes with an interface that enables authorized users and applications to manipulate and access stored credentials. In the latest versions of Windows, the Credential Manager supports syncing credentials across devices via a Microsoft account. This makes it more convenient for users to gain access to resources across different systems.

Using the Credential Manager APIs (**CredUl** or **CredProv**), applications can retrieve or delete previously saved credentials, create new ones, or use stored credentials to authenticate without having to prompt the user repeatedly. It's worth noting that these APIs work solely for the logged-on user account on the current system.

The **Windows Credentials Explorer** tool allows you to decrypt credentials stored in the Windows Credential Manager by directly accessing raw data, without relying on any known Credential Manager APIs. This means that you can retrieve credentials from any user account on the system, including the System account.

1 Select Source of Credentials

Windows Credential Manager		×
Select credentia	als location	Step 1/5
The Credential Manager is a tool us or in credentials backup files. First,	ed to decrypt and analyze Windows credentials stored locally, you should select the type of credentials you wish to decrypt.	on external drives,
Read more information about Wind	lows credentials	
Credentials location		
Credentials of the local comp	uter	
Credentials of an external PC		
Credentials backup		
CRD backup file:	E:\123.crd	
Password:	*****	۲
	Next >	Cancel

The first dialog window prompts you to select the source of the credentials you wish to decrypt. You have three options to choose from:

1. Local PC: Select this option to decrypt credentials stored on your current machine.

- 2. External PC: Choose this option to decrypt credentials stored on an external hard drive.
- 3. Backup File: If you have a backup file containing the credentials you wish to decrypt, select this option. Credentials backup files usually have a .crd extension and securely store encrypted copies of all credentials in one place. If you know the password the .crd file was protected with, you can unprotect and view stored credentials using this option of the Windows Password Recovery.

2 Specify Windows Directory and User Profiles Folder

Windows Credent	ial Manager	×
😋 😡 s	Set all required directories	Step 2/5
Provide the Window	ws directory along with the profiles folder (usually located at C:\Users).	
 Location of the sy 	stem folders	
Windows director	ry: D:\win	
Profiles directory:	d:\Users	
	Next >	Cancel

In the second dialog, you must provide the location of the Windows directory and the user's profiles folder. This information is crucial for the program to locate the appropriate files and folders related to the Credential Manager.

3 Scan and Select User Account

Windows	Windows Credential Manager X			
G	Choose the user a	account		Step 3/5
To procee	d with decryption, select the	e user acco	ount you need, and click t	the 'Next' button
	User name	Found	Last logon	Profile directory
1	SYSTEM	1	_	D:\win\System32\Config\systemprofile
2	admin	25	2023-03-12 11:45:15	d:\Users\Admin
Windows	Credentials Guard is OFF	:		
				Next > Cancel

The third dialog of the Wizard scans for user accounts based on the specified folders and displays a list of user names found. From this list, choose the user account you want to search and decrypt credentials for.

4 Enter User Password or PIN

Windows Credential M	anager	×
🚱 🔣 User o	or system credentials are required to decrypt the data	Step 4/5
Provide the user's login p encryption data was crea entropy file is required as provide the file's program	assword or PIN (Microsoft accounts only) here in order to decrypt the Mi ted using additional entropy (refer to the CryptProtectData API for more i well. In that case, create a simple binary file with the entropy string or da path here.	aster Key first. If the information), the ita, save it, and
Additional parameters a	are required in order to decrypt the data	
User SID	S-1-12-1-2214964667-1090076210-1622446738-457609414	
User password		
O User PIN		
Password-less (extr	act from biometrics, ARSO, cached logons, etc.)	
Entropy file (optional)		
	<u>N</u> ext >	Cancel

Once you select the user account, provide either the password or PIN (Microsoft account only) associated with that account. This information is necessary for decrypting credentials that are protected with DPAPI. Please ensure that you have the authentication details correct before proceeding. The password-less decryption is the state-of-the-art feature worth trying out if the user password is forgotten or unknown for some reason. If the program locates any sensitive information like biometrics, ARSO, cached logon, etc., you can decrypt credentials without providing the user password or PIN.

5 Locate and Decrypt Stored Credentials

Ca Decrypted credentials	
	Step 5/5
Now select the credential file to decrypt and view the properties it stores. Right-click the table to	o open the context
admin	
E5FE1A7B6923945F4BF74AA1CEF67F97	Ţ
Name Value	
Credential type Domain password Persist Enterprise Modified 2023-02-08 15:45:29 Attributes 0 Target name Domain.target= Comment SspiPfc User name Password	

In the final dialog box, the program searches for any stored credentials associated with the selected user account and decrypts them using the provided authentication details. Once the decryption process is complete, you can view properties of each decrypted credential, such as network or user name, password, and other relevant information.

2.8 Settings menu

2.8.1 General settings

The general settings are divided into five parts.

2.8.1.1 General options



Use sliding transitions in wizards and dialog - enables a graphical transition effect in dialogs.

<u>Check for updates at startup</u> - check if an update is available every time the program starts. The option works only if PC is connected to internet.

<u>Duplicate log messages to wpr.log file</u> - this option when set, writes all messages the log window holds to WPR.LOG file. Setting this option may cause performance degradation on big list of hashes because the wpr.log flushes its content to disk every time new message is arrived. It can however be helpful when the program stalls or works unstable. WPR.LOG is located in the program's installation directory.

<u>Overwrite log file</u> - overwrite the log file every time the program starts. Otherwise new messages will be appended to the end of the log file.

Password Cache

All passwords found by the program are cached by default. A very helpful thing that is engaged in many subsystems. For example, in the intellectual or preliminary attack. Deleting password cache is recommended in cases of the extreme need only. For example, when their number exceeded ten thousand. In this case, the search speed for some attacks can drop significantly.

Additionally, you can duplicate found passwords to text file. So even if the program fail unexpectedly or in case of sudden power failure, the found passwords guaranteed to be written to file.

2.8.1.2 Attack options



Attack progress

The first group of settings allows setting the save and update intervals for the current state of an attack. By default, an attack saves its state every 5 minutes (further on, you can resume the attack from the last saved point) and updates the screen every 3 seconds.

Advanced

- 'Save project every time ...' setting this option will force the program automatically save project every time new attack is starting or stopping.
- *'Finalize mutation on found passwords when attack is stopped or finished'* activate password analysis and mutation module for found passwords after attack. This option can be extremely useful; for example, for recovering similar passwords.
- '*Run simplified fingerprint analysis upon attack completion*' activate second analysis module. It launches on attack completion, creates new fingerprint dictionary out of found passwords, trying to retrieve more passwords. Useful on big list of hashes, history hashes, etc.
- 'Automatically run preliminary attack when importing hashes' automatically launch preliminary attack upon import. This attack recovers extremely weak passwords within seconds.
- 'Automatically select hashes upon import' automatically select all hashes after importing.
- *'Keep LM hash even if it does not correspond to the found password'.* Windows Server 2003 and higher OSes store noise (randomly created) data in LM slots if the LM authentication is disabled on the server. If this program's setting is not set, the WPR checks the appropriate LM hash (if one

exists) every time a new password is found. Once the program detects that the LM hash does not correspond to the found password, it just zeros out the LM hash. If this setting is set on, the program does nothing with incorrect LM hashes.

2.8.1.3 CPU settings

General options		\times
Ś	CPU settings	
General options	CPU options	
ß	Processor utilization: 8	
55	Instruction set: AVX2 Detect	
Attack Options	Attack thread(s) priority:	
	CPU information	
	Name: Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz Speed: 3398	
CDU settings	Cores: 8 Type: family 6, model 12, stepping 3	
GPU settings	Features flag: BFEBFBFF	
	MMX: Yes	
a contraction of the second se	SSE2: Yes	
GPU health monitor	AVAZ. IES	
	Automatic hardware utilization	
	OK Cancel	

Since the majority of the attacks supports multithreading, you can set the number of search threads to be run simultaneously. In the majority of cases, it should match the number of cores in your CPU. However, if the CPU supports the Hyper-Threading technology, you can even double the number of search threads that run simultaneously.

The DES, MD4 and SHA-1 hash algorithms in Windows Password Recovery are optimized for the following CPU architectures: X86, MMX, SSE2 and AVX2. Naturally, on CPU that support newer architecture, the search would run faster.

It is not recommended to set the attack priority above normal; otherwise, you may observe a considerable reduction of performance of the entire system.

The processor settings are unavailable if automatic hardware utilization is set on.

2.8.1.4 GPU settings

General options			×
General options	GPU settings Platform C NVidia CUDA T Image: Setting of the se	Devices ☑ GeForce RTX 2060	
CPU settings	Device description Device name Compute capability Peak performance (billion ops/sec Multiprocessor count	GeForce RTX 2060 7.5 3226 30	
GPU settings	Cores Clock rate Total memory Memory clock rate Memory bus width	1.68 Ghz 4096 Mb 7001 Mhz 192-bit	
monitor	Adjust GPU Timeout Detection and F	Recovery	Save device info to file Cancel

Before running an attack on a GPU, select it in the application's general settings by simply ticking the check box by the GPU name. All the main characteristics of the device are displayed in the property table.

The software supports NVidia (built on the CUDA platform) and AMD (built on OpenCL platform) GPUs.

The GPU settings are unavailable if automatic hardware utilization is set on.

2.8.1.5 GPU health monitor

General options		\times
Ś	Limit GPU activity	
General options	⊂ GPU device	_
2 P	GeForce GTX 1060 3GB 1152 SP, 1.71 GHz, 3072 MB	
Attack Options	Radeon(TM) RX 460 Graphics 14 SP, 1212 MHz, 2048 MB	
CPU settings		
GPU settings	GPU limits GPU limits Set retain temperature (°C) Set abort temperature (°C)	
	Temperature maintaining method	
	Increase fan speed	ור
GPU health monitor	Custom fan speed (%) 50 ‡ Start test	
	Custom power (%) 100 ‡ Reset to default	S
	OK Cancel	

You can use the GPU health monitor to retain a GPU temperature at a certain value, as well as to abort password search in case the temperature reaches a certain critical value.

To set the retaining temperature, select your GPU from the list of available devices, check the appropriate option on and set up your value at degrees Celsius. Then select one of retaining methods:

- Simultaneously increase fan speed and decrease GPU power. Use it if you want to reduce the fan noise, but a slight decrease in the performance is not a problem for you.
- Increase fan speed. When this method is selected, the program increases fan speed when the temperature exceeds the limit and decreases when it is cooling down. Use it when you need maximal performance. It is also recommended to set up the abort temperature to avoid GPU overheating.
- **Decrease GPU power**. It is quite effective to cool down a GPU temperature. The GPU processor frequency automatically adjusts to the retaining temperature while the fan speed will be determined by your system settings. The disadvantage includes a significant performance drop down in certain cases.
- Custom settings include user-defined fan speed and/or GPU power.

Use the abort temperature to stop searching for passwords once the GPU temperature reaches the limit.

Warning! Once an AMD device is selected, the program supports for abort temperature option only. AMD drivers do not allow overriding some values manually for most cards and have numerous bugs that may cause system instability or even BSOD. Consider using the Radeon built-it tool to set up your own performance profile instead.

2.8.1.6 Event notifications

General options		×
General optio	Event notifications	
Attack Options	Be careful, even though the notifications have a min events executed with errors may cause the program Event notifications	or impact on the program's performance, some improperly set events or to work incorrectly.
	Triggering event	Notification type
	A password was found	Play sound
CPU settings	A password was found	Send e-mail message
	Task is complete	Post HTTP notification
	New password items were imported/added	Save to textual file
	Task is complete	Exit the program
GPU settings		
GPU health monitor		
Event	Add	Remove Remove all Load Save
		OK Cancel

The event notifications can be used to automate the password recovery process. There are several types of events:

- The program is starting. This event occurs when the program has just been started and is ready to go.
- The program is closing. Nearly to exit the program.
- New project was loaded/created. Creating an empty project conforms to this event as well.
- **Project was saved**. Either a user or the program saved the current project to disk. This event may occur multiple times during an attack.
- Project was closed. A user closed the current project.
- Task was started. A new task has been started. A task includes one or more attacks and at least one password item.
- Task is complete. This event occurs when a user stopped the task, the last attack is over, or upon some errors.
- New password items were imported/added. A user loaded or added one or more password items to the current project.
- Attack was started. A new attack was launched. This event may be fired multiple times within a task scope. For example, when running a Batch attack.
- Attack is complete. This event occurs upon successful or unsuccessful completion of the current attack.
- A password was found. The program found a password.

Once a new event is selected, you are free to set a notification that should be executed when this event is triggered. Here's the notification types:

- Play sound file
- Print out
- Send e-mail message
- Post HTTP notification
- Run a program
- Save to textual file
- Save to *. INI file
- Save to SQLITE database
- Save to *.HTML file
- Exit the program
- Shutdown the system
- Restart the system
- Hibernate the system
- 2.8.2 Attack Settings
- 2.8.2.1 Preliminary attack

Preliminary attack (developed in Passcape) is quite effective against short, simple, dictionary, repetitive, keyboard, etc. passwords and consists of several mini-attacks. Each mini-attack can be enabled/disabled individually.

Preliminary attack options
Preliminary attack is quite effective against simple and well-known passwords
Preliminary attack is a time-limited simple set of several mini sub-attacks. It is often run when there's no need to launch a full attack. Preliminary attack is used also when importing hashes into the program. You can activate/deactivate it in 'General Options' dialog. Read more about preliminary attack
Use the following preliminary sub-attacks
 Common brute-force attack for short and simple passwords Dictionary attack against well-known passwords Extended dictionary attack (simple mutation is on) Attack repeatable sequence of a characters, eg. '11111', 'aaaaaa' Attack simple patterns, eg. '123456', 'qwerty' Attack complex patterns Keyboard combinations attack. Based on all possible keyboard combinations like 'qwerty', 'qazwsx', etc. National keyboard combinations attack. The same as previous but uses national keyboard layout. Complex keyboard attack Smart check complex keyboard combinations Passcape Password Prediction attack. Effective against hashes imported from local computer. User name mutations attack. For passwords based on user name. Check hex passwords (ones based on hexadecimal values). Numbers as words Search for non-standard symbols
OK Cancel

Preliminary attack run about 10-20 minutes or even faster. It consists of at least the following subattacks:

- Common brute-force attack. Performs several simple brute-force attacks based on predefined character sets.
- Simple dictionary attack. Fast check the password by verifying all words from a given dictionary.
- Extended dictionary attack. It's almost the same as above but with some smart mutation options set on.
- Attack on repeatables. Checking passwords as a repeatable sequence of a character. Eg. '1111111' or 'xxxxxxx'.
- Attack on simple patterns, like '123456' or 'qwerty'.
- Attack on complex patterns. The same as above, for compound patterns.
- Keyboard attack checks for keyboard passwords and all possible combinations. Eg. 'qwer', 'qazwsx', 'asdzxc', etc.
- National keyboard attack. The same as above, but checks passwords typed in national keyboard layout.
- Complex keyboard attack is the same as previous 2 attacks, for compound keyboard patterns.
- Passcape Password Prediction attack is the most complicated and state-of-art password prediction tool.
- Attack on name-based passwords.
- Attack on hex passwords (eg. 7A49F3).
- Attack passwords based on numbers (as words).
- Search for non-standard symbols and short passwords that were created using non-standard UNICODE symbols.

2.8.2.2 Artificial intelligence attack

Artificial Intelligence Attack is a new type of attack developed in our company. It is based upon a social engineering method and has never been implemented in password recovery applications yet.

This one is mostly used when the hashes are imported from the local computer. Intellectual attack scans the local computer, indexes and creates the list of found words and passwords, analyzes them, upon the results of the analysis produces user's preferences, performs the mutation of the found words and, based on all that, attempts to recover the passwords.

This attack allows, without resort to time-consuming and costly computations, to almost instantly recover certain passwords encrypted with hash functions. The basic idea behind the Artificial Intelligence attack is that an average user very often chooses similar words and word combinations or follows the same password generation rule when creating one's passwords. With that in mind, we could attempt to figure that rule out and pick the original password.

tificial Intelligence attack options					
Use the power of Artificial Intelligence to guess passwords quickly					
The Artificial Intelligence attack has provided that the search is performe are: Password mutation - normal, In before launching the attack.	The Artificial Intelligence attack has proven itself to be most effective when searching for Windows passwords; provided that the search is performed on the original system. The best (by the speed-quality ratio) attack settings are: Password mutation - normal, Indexation level - normal. It is highly recommended to close all other applications before launching the attack.				
Read more about Artificial Intelligen Index files Search passwords by indexing	g files, mailboxes, browser configurations, mru items, etc.	ון			
Password mutation level:	Light (fast)				
Word indexation level:	Word indexation level: Normal (slow)				
Index sectors on a drive O Search passwords by scanning	ng physical sectors on a drive				
Drive:	Disk 0: C: 1862.92 Gb				
Word indexation level:	Light (fast)				
Accept alpha-numeric passwords only Limit maximal word size to (characters): Use custom word delimiters					
	OK Cancel				

Although this sounds somewhat abstractive, in the reality the attack clearly splits into four successive steps.

- 1. Initiating the collection of private data. Here comes into action the password retrieval and indexation module, which looks for all available and hidden in the system passwords entered by user at any moment of time. Those include network access passwords, ICQ, email, FTP, Windows account passwords, server passwords, LSA Secrets, etc.
- 2. Launches the data collection and indexation module. During the execution of this step, we analyze the activity of the user (or all users, if the indexation module selected is different than

Light) in the system. Next, basing upon that, we generate the list of words - potential passwords selected from the text files, archives, internet browsers' history, email correspondence, etc.

- 3. Includes the semantic analysis module for the database of found passwords and the list of potential passwords.
- 4. On the final stage, the data analysis module will perform the mutation of the words and attempt to pick the passwords.

In the beginning of the attack, the program will search the system for all passwords it knows of. For that purpose, there are currently 32 mini modules for decrypting system, mail, browser, messenger, archive and other passwords. Then there goes the file and data indexation, along the course of which the program generates a potential attack dictionary. The third module breaks the passwords and words into pieces, out of which in the last module it will assemble new combinations for picking and guessing the original password.

In average, with the least indexation and mutation levels, the attack time may vary between 1 minute and 10-15 minutes, depending on the network activity of the user. On a home computer, the entire route normally takes not more than 2-3 minutes. Naturally, the more complex is the mutation and indexation level, the more efficient will be the search. However, reaching the topmost indexation and analysis level may take hours and even days, depending on the speed of the password validation algorithm and the number of users in the system.

The Artificial Intelligence attack has proven itself to be most effective when the search is performed on the original system. Only two options are available here: password mutation depth and word indexing level. The most preferred options for running a speedy attack are *Light:Light*. For a deeper (and at the same time slower) search, set these options to *Normal* or even *Deep*. The duration of an intellectual attack also depends on the configuration of your system, your network load, and other factors.

It is highly recommended to shut down all other programs before launching the attack. If your Artificial Intelligence attack runs very slow, you may need to remove your program's cached password (eg. total amount of the cached passwords exceeds 10000).

Windows Password Recovery version 9.5 now comes with a brand new feature which allows password searching by indexing raw sectors on selected drives. This feature works for both LM and NTLM hashes, looking for both ASCII and UNICODE passwords. You can change some advanced search options here. For example, '*Word Indexation level*' sets additional mutation on all found passwords. Be careful, walking through all sectors of the target drive with this option set to 'Hard' may take quite a time. Note that the sector-based scanning algorithm is not effective against drives which have a full-disk encryption set on. Like Bitlocker or TrueCrypt, for example.

2.8.2.3 Fingerprint attack

Fingerprint attack is a relatively new tool for recovering complex passwords, which could not be decrypted by other attacks. The idea of the attack is that here, to recover a password, we take neither individual words from the source dictionary, like in the dictionary attack, nor even word combinations, like in the combined attack, but so-called "fingerprints". Now, every source word from the dictionary is used for generating several fingerprints. If some password is found during the attack, it participates in generating new fingerprints, and the attack goes another round.

Before launching the attack, specify the source dictionary to be used for creating the fingerprint bank. The software comes with a dictionary, common.pcd, optimized for this attack, but you can use yours or download one off the Internet ('Online dictionaries' tab). There are no certain requirements to the dictionary, except one: the source dictionary must not be too large; otherwise, the attack will take

significant time. You can use dictionaries with national passwords, if you suspect that the sought password contains characters in a national encoding.

Fingerprint attack options	X
d General options Dictionary generator Online dictionaries	Þ
Fingerprint general options	
Fingerprint attack uses input dictionary to generate all possible variations for complicated passwords. Once a password is found, it then uses in further identification of more complicated passwords. The attack is very effective if all additional mutation options are set on.	I
Read more about Fingerprint attack	
Initial dictionary	
E:\Program Files\Passcape\WPR\dic\common.pcd	
Additional mutation options	
Use "Passcape Password Prediction" engine to generate additional source passwords	
 ✓ Use Reyboard and frequently used sequences ✓ Use dates 	
Use numbers and common sequences	
Use extra word mutations (time-expensive)	
✓ Maximize enciency when generating ingerplints ✓ Loop until no more passwords are found	
OK Cancel	

Here is the way to generate fingerprints: first, break each word from the source dictionary into onecharacter passwords, then - into 2-character, etc. For instance, break the source word **crazy** into onecharacter fingerprints. We get:

c r a z y Now, two-character: cr ra az zy Next, three-character: cra raz azy And, finally, four-character: craz razy

We have got 5+4+3+2=14 fingerprints, not counting the source word.

Repeat this for each word of the source dictionary. After this, all the fingerprints are dumped into a single database, naturally, discarding duplicates. We have got a database of fingerprints that would be used for checking passwords by gluing all the fingerprints with each other and finding the match.

The real fingerprint generation algorithm is much more sophisticated. Moreover, there is an option in the attack settings, **Maximize efficiency when generating fingerprints**, which uses a more sophisticated algorithm, which maximizes the efficiency (at the expense of speed) by generating additional fingerprints.

Let's take a look at the remaining options.

Use PPP engine to generate additional passwords - use passwords found in other attacks when generating fingerprints.

Use keyboard and frequently use sequences - add keyboard combinations and common sequences to fingerprint bank.

Use dates - add dates to fingerprints.

Use numbers and common sequences - use digits and simple combinations of letters.

The most careful attention should be paid to the option **Loop until no more passwords are found**. That is where fingerprint attack can really show itself off. Here is how it works: if at least one password is found during an attack, when the attack is over, the password participates in generating new fingerprints, and the attack runs again. This option works great on large lists of hashes and on password history hashes. However once the option is set, you will not be able to proceed the attack from the last saved position.

The second tab with the settings allows to create and record a custom dictionary using current options of fingerprint attack. Be careful; that dictionary may take up a lot of space on your computer's hard disk.

Fingerprint attack options	x
General options Dictionary generator Online dictionaries	Þ
Dictionary generator - create fingerprint wordlist	
Generated by this attack passwords can easily be saved to file. So you can create your own dictionary and use it in another program. Be careful, dictionary creation may take quite some time depending on the source wordlists given and creation rules set.	
Read more about Fingerprint attack	
Dictionary generator	
Initial dictionary E:\Program Files\Passcape\WPR\dic\common.pcd Generate	
OK Cancel	

On the third tab, you can download source dictionaries for fingerprint attack from the Internet.

General options	Dictionary establish a o vnloading. Se	y generator	Online dictio e Passcape s you need an	naries erver and retri d click 'Downl	eves the list of oad' button to retrieve
Dictionary name	D/L size	Real size	Strings	Ratio	Group
Roget.pcd	47 Kb	186 Kb	17474	25%	InsidePro Softv
Hungarian.pcd	47 Kb	187 Kb	18013	25%	01 - Languages
15.pcd	61 Kb	191 Kb	11525	31%	03 - Sorted
Pockt.pcd	46 Kb	193 Kb	21110	23%	InsidePro Softv
dictionary swahili.pcd	43 Kb	209 Kb	19839	20%	InsidePro Softv
Swahili.pcd	43 Kb	209 Kb	19839	20%	01 - Languages
Common.pcd	57 Kb	213 Kb	24657	26%	02 - Common
Actor.pcd	56 Kb	215 Kb	25338	26%	InsidePro Softv
BSD.pcd	54 Kb	220 Kb	24482	24%	InsidePro Softv
Turkish.pcd	46 Kb	235 Kb	25865	19%	01 - Languages
	AC IZE	DOF ML	20000	100/	
•					•
lease select a dictionary to	download			Total files: 25	0 , total size: 1 384 1
Update list	Download	Use local v	version		

2.8.2.4 Brute-force attack (exhaustive search)

In cryptanalysis, a brute force attack is a method of defeating a cryptographic scheme by trying a large number of possibilities; for example, exhaustively working through all possible keys in order to decrypt a message. This definition was taken from <u>Wikipedia site</u>.

Well, to put it in simple words, brute-force attack guess a password by trying all probable variants by given character set. Eg. checking all combination in lower Latin character set, that is *'abcdefghijkImnopqrstuvwayz'*. Brute-force attack is very slow. For example, once you set lower Latin charset for your brute-force attack, you'll have to look through 217 180 147 158 variants for 1-8 symbol password. It must be used only if other attacks have failed to recover your password.

	Brute-force attack options	23
d Brute-force char	acter set Password length and position	Þ
A C Brute-force c	haracter set	
Brute-force attack assum can select predefined ch your own ones. Custom c	es trying all possible variations from the specified character range (set). You aracter sets (e.g., Latin characters, numbers or special characters) or define haracter sets can be saved to disk.	
Read more about Brute-fo	orce attack	
Predefined charsets		
AZ		
Custom character set	Load Save	
	•	
Statistics		- I
Total passwords	8 353 082 582	
Charset string	ABCDEFGHIJKLMNOPQRSTUVWXYZ	
	OK Cancel	

The brute-force attack options consist of two tabs.

The first tab is for setting the range of characters to be searched. You can use the predefined sets or create your own ones. To define your own character set, select the option '*Custom charset*'. This will enable two fields for defining a custom character set: the first one - for entering ASCII characters, second one - for entering non-printable characters. You can save your custom character set on disk. The program comes with several examples of user-defined character sets.

On the second tab, set the minimum and maximum length of the passwords to be searched. Please note that for attacking LM hashes the maximum password length should not exceed 7 characters. You can also set a starting password, which would start the search.

Below is a table that shows password strength depending on the password length and complexity. Assuming that the recovery speed is 100M passwords per second.

Character set	Password length	Password example	Time to crack (exhaustive brute-force search)
A Z	5	CRUEL	Instantly
A Z	6	SECRET	3s

Program's interface

A Z	7	MONSTER	1m 23s
A Z	8	COOLGIRL	36m 11s
A Z, 0 9	5	COOL3	Instantly
A Z, 0 9	6	BANG13	22s
A Z, 0 9	7	POKER00	13m 26s
A Z, 0 9	8	LETMEBE4	8h 3m 37s
AZ, az, 09	5	P0k3r	9s
AZ, az, 09	6	S3cr31	9m 37s
AZ, az, 09	7	Didlt13	9h 56m 33s
AZ, az, 09	8	GoAway99	25d 16h 26m 34s

2.8.2.5 Dictionary attack

In contrast with a brute-force attack, where all possibilities are searched through exhaustively, a dictionary attack only tries possibilities which are most likely to succeed, typically derived from a wordlist or a dictionary. Generally, dictionary attacks succeed because many people have a tendency to choose passwords which are short, single words in a dictionary, or are simple variations that are easy to predict.

Dictionary attack option	s			x
Dictionaries Filters Mutations Dictionary generator Online dictionaries			Þ	
Dictionaries	s list			
First you need to set up at least one good dictionary for successful recovery. In contrast with a brute-force attack, where all possibilities are searched through exhaustively, a dictionary attack only tries possibilities which are most likely to succeed, typically derived from a wordlist or a dictionary. The program comes with a short English wordlist. For complete list of dictionaries, check out our 'Wordlist Collection' please. Read about Passcape Wordlist Collection				
Dictionary name	Dictionary size	Strings	Full path	
WPR.pcd	758 858	416 797	e:\Program Files\Microsoft Visual Stu	
				_
Add Add folder Remove Move up Move down				
OK Cancel				

On the 'Dictionaries' tab, set up the list of dictionaries to be used in the attack. Supported are plain-text dictionaries in the formats ASCII, UNICODE and UTF8, as well as encrypted/compressed dictionaries in the native PCD format, developed in Passcape Software. ZIP and RAR packed wordlist are supported as well with some restrictions. To deactivate a dictionary, simply clear the checkbox by its name. In this case, the dictionary, although it remains on the list, will be skipped during an attack. The software comes with a 360000-word dictionary. For complete list of dictionaries, check out our wordlist collection please. Or you can use our online dictionaries as an alternative.

The '*Filters*' tab filters the words from a dictionary by the include/exclude principle. If the first, inclusive, filter is enabled, the attack will accept only the words that contain at least one of the characters entered in the filter. If the second, exclusive, filter is set, the program will skip the words that contain at least one of the entered characters.

The 'Mutation' tab allows setting all kinds of possible combinations of the words to be searched. For example, if you set a strong mutation, the program will create several hundreds of analogs for each word from the dictionary. For example, secret - Secret - s3cr3t - secret123, and so on. You can set up to three mutation rules: *Weak* - less number of mutations and, in its turn, greater verification speed; *Strong* - for greater number of mutations, to the prejudice of the speed, and the happy medium, default option (*Normal*).

You can use Dictionary Generator to create your own wordlists based on options of the first three tabs.

Online dictionaries. The program has a great feature that allows downloading and using existing dictionaries available on the Passcape website. We have accumulated quite a large dictionary collection - over 250 items. That should get you rid from the extra hassle on finding the required content on the Net.

Customizing mutations

Starting with version 4.0, the program has ability to customize the smart mutation of the Dictionary attack. All mutation rules are clustered into 16 primary groups. You can set one of three mutation levels or disable mutation separately for each group.

Dictionary attack options						
Smart mutatio	Smart mutation					
Weak - less number of smart mutations and greater verification speed; Strong - for greater number of mutations, to the prejudice of the speed, and the happy medium, Default option. The smart mutation is based upon 150 social rules and combinations. Be careful, mutating even a single word may take quite significant time ! <u>Read more about Dictionary attack</u>						
Mutation level		Weak (fast)				
Normal (slow)		Strong (very slow)				
Custom		0				
Character case	Weak (fast)	Character swap	Normal (slow)			
Digits append/prepend	Weak (fast)	Character duplicate	Normal (slow)			
Head and tail	Normal (slow)	Insert delimiters	Weak (fast)			
L33t	Normal (slow)	Add dates	Weak (fast)			
Abbreviations	Normal (slow)	Oem convertion	Disabled 🔹			
Dups and revers	Weak (fast)	Word shift	Disabled			
Vowels and consonants	Normal (slow)	Character substitution	Normal (slow)			
Character skip	Normal (slow)	Length truncate	Weak (fast)			
OK Cancel						

For example, you can turn off OEM mutation (and thus double your Dictionary attack speed) if you sure the password you're looking for contains Latin characters only. Simple description of what all these mutation groups mean is given below:

Group name	Description	Examples (for word ' password')	Comments
Character case	Checks case combinations of the input word.	Password, PassworD, PaSsWoRd	Maximal (Strong) level of the mutation group DOES NOT generate all possible case combinations of input words. To check all possible case variants, consider using <u>Hybrid</u> <u>dictionary attack</u> (aN rule)
Digits append/prepen d	Adds digits to the beginning or to the end of the word.	password99, 2Password, PASSWORD3	Maximal level adds 2 digits.
Head and tail	Almost the same as previous one, but appends or prepends words,	#Password#, password12345,	

Program's interface

Group name	Description	Examples (for word	Comments
l33t	abbreviations, characters, keyboard combinations, etc. Creates different combinations using <u>leet language</u> .	4everPASSWORD, Passwordqwerty p@ssword, P@\$\$w0rd, P@\$\$W0RD	
Abbreviation	Converts several character combinations (if the initial word contains any) into abbreviations.	ihateyou -> ih8you, lh8u	
Dups and revers	Revers, duplicates the word, etc.	drowssap, passwordpassword, PasswordDrowssap	
Vowels and consonants Character skip	Mutates vowels and consonants (English characters only). Skips a single character of the original word.	Psswrd, PaSSWoRD, pAsswOrd assword, Passwrd, Pasword	
Character swap	characters.	apssword, Passowid	
Character duplicate	Duplicates characters.	ppasword, ppaasswwoorrdd, Passworddddd	
Delimiters	Separates characters with	p.a.s.s.w.o.r.d, P-a-s-	Maximal level uses 10
Dates	Adds dates to the end of the word.	Password2010, password1980	Even though the mutation engine can generate more complicated variations (for example, password03171998 or Password19710830), this feature if turned off here even in maximal mutation level.
Oem convertion	Converts English word into another language and vice-versa using alternative keyboard layout (second language of the OS). Stupidly shifts all characters of the	If your OS has 2 languages installed (let it be English and Russian), the program will convert initial word password into Russian and Russian will be converted into gfhjkm . asswordb, dbasswor	The program works correctly for 2 or even more languages. So if you have 5 languages installed locally (including English one), there will be 4 different combinations of the input word.
Word shift	Stupidly shifts all characters of the word to the right or to the left.	asswordp, dpasswor	
Character substitution	Replaces a character of the initial word.	oassword, passqord	This is quite helpful rule assuming the fact that the characters for substitution are taken from a special table. For example, the character 's' will be replaced with the following ones: 'a', 'w', 'e', 'd', 'x', 'z'. You can notice that all of these

Group name	Description	Examples (for word ' password')	Comments
			characters are located near 's' on any qwerty keyboard.
Length truncate	Truncates word length to probe all possible length combinations.	passwor, passwo, Pass	

2.8.2.6 Mask attack

Mask attack is an irreplaceable tool when you know a fragment of the password or have any specific details about it. For example, when you know that the password consists of 12 characters and ends with the *qwerty*, it is obvious that searching the entire 12-character range of passwords is unreasonable. All what would be required in this case is to pick the first 6 characters of the sought password. That is what mask attack is for.

In our case, we could define the following mask: **%c%c%c%c%c%cqwerty**. That means that the program would serially check the following combinations: aaaaaaqwerty .. zzzzzqwerty. If the original password is *'secretqwerty'*, it perfectly hits our range.

Mask attack options	×
Mask options Dictionary generator Mask tips	Þ
Brute-force attack by a given mask	
The mask attack is used if there's some information about the password. For example, you know that the password begins with 'loveme' and followed by a name. So you can set the mask 'loveme%c%c%c%c%c%c check all possible variants starting with 'lovemeaaaaaa' and up to 'lovemezzzzz'. You can set multiple mask within one attack. <u>Read more about Mask syntax</u>	to is
Password masks	
%c%c%c%c%cqwerty	
Custom character sets	5
Statistics	
Password range aaaaaaqwerty zzzzzzqwerty	
Total passwords 308 915 776	
OK Cancel	

The group of the Password Masks options is aimed to set a mask (or several ones), that will be used to generated passwords by. In most cases, if you know a part of the password, it is enough to specify a single mask. When a mask is selected, the statistics group shows the range of output passwords and the number of passwords generated by this mask. You can save your masks to disk to use it in another project, for example. The program also allows you to generate dictionaries by given masks.

Suppose that you have a list of decrypted passwords and you would like to generate mask templates

out of these passwords. Nothing easier. Run the mask generator and show the path to your password list there. You can sort the resulting masks alphabetically, popularity, or by search range (the fastest go first).

Generating multiple password masks X	(
Generating multiple password masks	
To generate all possible masks, please supply a wordlist with real passwords. If the wordlist contains passwords in different languages, worth splitting it into several ones using the wordlist tools for example.	
Input wordlist with real passwords	
D:\Passcape\1\1.txt	
 Sort output masks Sort by alphabet Sort by popularity (most frequently-used first) Sort by password range (fastest masks first) 	
Imit the maximal number of generated masks Max number of masks 100	
Generate	

The mask syntax is quite trivial and consists of static (unmodifiable) and dynamic (modifiable) characters or sets. Dynamic characters/sets always have a leading %. For example, if you set the mask secret%d(1-100), the program will generate 100 passwords (secret1, secret2: secret100).

Windows Password Recovery supports the following dynamic mask sets:

- %c lower-case Latin characters (a..z), 26 symbols
- %C upper-case Latin characters (A..Z), 26 symbols
- %# full set of special characters (!..~ space), total 33 symbols
- %@ small set of special characters (!@#\$%^&*()-_+= space), 15 symbols
- %? all printable characters with ASCII codes of 32..127
- %* all ASCII characters (codes 1 through 255)
- %d one digit (0..9)
- %d(x-y) numbers between x and y inclusive
- %r(x-y) user-defined characters with serial UNICODE codes between x and y
- %r(x1-y1,x2-y2...xn-yn) set of several non-overlapping sequences of UNICODE characters.
- %[1..9] a character from user defined charset 1..9

- %[1..9](min-max) user-defined range of variable length (from min to max). You can set up to 9 your own custom character sets.
- %% standalone static character %

 generates password range test0test9, 10 passwords total
- test1980 test2007, 28 passwords
- 256 passwords with Arabic character at the end
test ~test~, 1089 passwords
- admina adminzzzzz, where %1 is user defined charset 1 (az)
- aaapin000 zzzpin999, %1 is user character set az and %2 is second user-
ontains characters 09

By switching to **Dictionary generator tab**, you can generate your own dictionary by a given mask, and save it to disk. This feature available in Advanced edition of the program only.

Third tab of the mask options contains a short description of the mask syntax and a couple of simple examples.

If you feel lost and sick of all these stuff, then use the Mask Builder (in Password Recovery Wizard) that has a much more user-friendly graphical interface.

Passv	assword recovery Wizard X							
0		Mask builder					Step	o 3/4
A ma either progr	sk consists o r from predefi am will use th	f known and unk ned or from custo ne mask to brute-f	nown parts (ele m charset. Buil orce all possibl	ements). Every u d your password e character con	nknown eler I mask by sin Ibinations yo	nent can be repre nply adding one o u specified.	sented by a chara r more mask elem	acter ients. The
	ssword mask						quatu	
	u	02	U	u z	02	02	quoity	
	Add know	vn 😣 Rei	nove	🛟 Edit		Move left	A Move	right
	Add unkno	wn 🗈 Dup	licate					
Raw	mask:		%c%c%c%	c%c%cqwerty				
Total	passwords:		308 915 77	6				

aaaaaaqwerty ... zzzzzqwerty

wvzygbgwerty

2.8.2.7 Base-word attack

Password range:

Password example:

Base-word attack (developed by Passcape) is in many ways similar to mask attack. However, here you don't need to set up the syntax; simply enter the keyword, which supposedly was the base word for the password. It is an irreplaceable recovery tool when you know a portion of the password or its basic component. Normally, such cases dispose to using mask attack; however, it does not always allow coping with the task set forth. Suppose our password was 'S10wDr1v3r'. Trying to recover such a complicated password using brute-force attack would be an ungrateful job, even if you are quite sure that it is based upon the 'slowdriver' word. These are the cases when the base-word attack will rescue you.

With this tool, the program will attempt to recover the original password, trying all possible combinations founded upon 15 groups of rules (total over 150 rules). If you enter '*slowdriver*' in the field, you will see that the program has generated several thousands of different combinations upon this phrase, and one of those combinations could match our password.

Cancel

 $\underline{N}ext >$

Base-word attack options	X				
Base-word attack is eff	Base-word attack is effective if you know a part of the original password				
Type a word or a phrase the password of the phrase 'slowdriver'. Type this phrase for the real password base upon this wo will be able to find 'freaked' passwords I Read more about Pase. Word attack	may consist of. For example, you know that your password was formed on e and start the attack. The program will try all possible combinations to scan rd. In our example the program will generate thousands mutated words and ike SlowDriver, Slowdriver123, SLoWDRiVer, S10wDr1v3r, etc.				
Options					
Single base-word	Multiple input base-words				
Base-word of the password	secretlife				
Input wordlist					
Mutation level	Strong (very slow)				
Dictionary generator					
Dictionary size	Generate				
	OK Cancel				

If the length of the input phrase exceeds 8-10 characters, the mutation may take significant time. If you remember the original password precisely and simply have forgotten the sequence of the upper-case and lower-case characters in it, you can select the option '*Mutate character case only*'. With this option selected, the program will generate passwords with all possible combinations of upper-case and lower-case characters, total 2ⁿ passwords, where n - is password length. For example, for the password '*slowdriver*' the program will generate 2ⁿ10=1024 different combinations for each keyboard layout installed on your computer. You can also generate a dictionary on those mutations and save it on a disk (available not in all editions).

Note, if your password length exceeds 15-16 characters, it may take quite some time to prepare (mutate) the password for the attack.

In Windows Password Recovery version 9.5 the Base-word recovery was split into 2 modes: single input word and many input words. The multiple input words mode acts like the Dictionary attack with maximal mutations set on, but generates much much more passwords (even if the mutation level of the Base-word attack is set to 'Weak'), which maybe useful in a certain situation.

2.8.2.8 Combined dictionary attack

Combined dictionary attack (developed by Passcape Software) is great at recovering passwords that consist of 2,3 and even 4 words. This type of attack on difficult and compound passwords is very similar to the simple dictionary attack, except that instead of using a single word for password verification here we use a combination of words or a phrase created by combining words from specified dictionaries. To successfully utilize this attack, set at least two dictionaries and the rules for generating passwords. You can set the regular dictionaries used in the simple dictionary attack, but it is recommended to use rather

small dictionaries with the most common words. Perfect dictionaries for the combined pass phrase attack are those that have different forms of words in them; e.g. jump, jumper, jumped, jumping.

Combined attack sets a certain limit to the number of dictionaries that can be used; that's not more than 4. Thus, the general limitation of this attack is that only password phrases of not more than 4 words can be recovered using this attack.

Another essential drawback is the wide range of phrases generated. And, as the consequence, the proportional increase of the time spent on the validation of a password. Keep in mind that when generating passwords that consist of 3 or 4 words, the generation process takes considerable time

If finding the right dictionary is difficult, don't worry. The software comes with a special dictionary for the combined attack. You can also take advantage of the <u>Online Dictionaries</u> tab or the corresponding button to download such dictionaries from the Passcape website.

Dictionaries

	Combined dictionary attack options	x				
Dictionaries	Creation rules Pass-phrase generator Online dictionaries	⊳				
Dictionaries	Dictionaries list to combine the words from					
This type of attack is ve for password verification specified dictionaries. Ye Bead more about combi	ry similar to the simple dictionary attack, except that instead of using a single n here we use a combination of words or a phrase created by combining word ou should specify at least 2 dictionaries to start.	e word Is from				
Primary1	C:\Program Files\Passcape\WPR\combdic.pcd	9				
📝 Secondary2	C:\Program Files\Passcape\WPR\combdic.pcd					
Secondary3						
Secondary4						
Statistics Output passwords Rules/combinations Size (strings)	233 401 014 6 Prim1=6237, sec2=6237					
	OK Cancel					

The way the combined attack works is really simple. For example, if you have set two dictionaries, the program will generate the passwords as follows: it will take the first word from the first dictionary and glue it with the first word from the second dictionary, then with the second word, and so on until the end. Then it checks the second word from the first dictionary and goes the same route, and so on.

9 phrases total.

To understand how the combined attack works, let's take a look at a couple of password generation examples that involve, in the first case, the same dictionary and in the second case - two different ones.

1. Suppose we've got a single dictionary with three words: action, bad, and computer. We will set this dictionary as two original sources: primary dictionary & secondary dictionary2 (see the figure). After these dictionaries have been processed, at the output we have the following phrases (they will be used when checking the password sought): 'actionaction', 'actionbad', 'actioncomputer' 'badaction', 'badbad', 'badcomputer' 'computeractio', 'computerbad', 'computercomputer'.

In the second case, we have got two different dictionaries. For example, the first dictionary consists of three words: action, bad, and computer. The second one also has three words: date, eagle, fail. In this case, we are going to have the following phrases: 'actiondate', 'actioneagle', 'actionfail'
 'baddate", 'badeagle', 'badfail'
 'computerdate', 'computereagle', 'computerfail'.

The example is plain but demonstrative. The idea is that for multiple sources you can successfully use both a single dictionary and multiple ones. It all depends on your imagination. The last example shows that a special attention should be paid to the order of the dictionaries if they are different. The order of the words in the phrases to be created depends directly on the order of the source dictionaries. In our second example, if we swap the primary and the secondary dictionaries, at the output we will obtain a completely different set of phrases.

Mutation rules

combined dictionary atta	ack options			×
d Dictionaries	Creation rules	Dictionary generator	Online dictionaries	⊳
Pass-phrase	creation rules			
Here you can create all p passwords by simply con spaces. However, you c words with caps, append Read more about combin	possible combination icatenating words fr an set your rules as d numbers, etc. <u>ned dictionary attac</u> operate different co	ons of phrases generated. By rom the source dictionaries, s well. For example, have it o	default the program will crow WITHOUT separating then reate phrases with spaces,	eate n with , begin
string1=firstupper stri	ngN=lower	mbinduons of the prinase		Add
string1=firstupper stri	ngN=firstupper			
string1=lower stringN	=lower delimiter =t	20		Modify
string1=firstupper stri	ngN=lower delimit	er=t20		Remove
string1=firstupper stri	ngN=firstupper de	elimiter =t20		
			H	emove all
✓ Insert words from sec For example: 12345Admi Note! This rule is active i Statistics	cond dictionary into in, A12345dmin, Ar if only 2 dictionarie:	every position of the every v d12345min, etc. s were set, it doesn't work fo	vord from dictionary 1 r 3 or 4 dictionaries.	
Output passwords	1 200 585 16	5		
Rules/combinations	6			
Size (strings)	Prim1=1414	5, sec2=14145		
			K Cancel	

Passwords created by the combined attack are generated according to special rules that are to be set on the second tab. By default, when password generation rules are disabled, the program generates passwords by simply gluing up the words from the dictionaries, without separating them with a space. For example, of the two words are 'my' and 'computer', you will get 'mycomputer'.

If word insertion option is set, the program additionally creates passwords by inserting words from second dictionary into every position of the word from dictionary 1. For example, if the first dictionary's word is **Admin**, and the word from the second dictionary is **12345**, the program will generate the following passwords:

12345Admin A12345dmin Ad12345min Adm12345in Adm12345n

And so on for all words of the second dictionary. Then goes another word from dictionary 1, etc. The option is active if only 2 dictionaries were set.

The generation rules are made to extend the password search options. For example: Mycomputer, MyComputer, MY COMPUTER, my-computer, etc. There are special rules available for this purpose; you don't have to know the syntax of them, for the mutation rule creation dialog is simple and intuitive.

A	dd/modify pass-phrase generation rule	x				
Set/change p	Set/change pass-phrase generation rule here					
Phrase properties		h				
Prefix	none					
First word	first character is upper, the rest are lower					
Word delimiter	constant text					
The rest words	first character is upper, the rest are lower					
Postfix	none					
Rule: string1 Example: I-Love	=firstupper delimiter=t2D00 stringN=firstupper e- My-Computer, total variants=1					

Each mutation rule consists of five elements:

- 1. *Prefix* text that will appear before each phrase. This element can be a character, plain text string, one digit between 0 and 9 or a number. For instance, if you set a one-digit prefix, the phrases created with this rules will look as follows: '0 aaa bbb', '1 aaa bbb': '9 aaa bbb'.
- 2. *First word* the action to be performed over the first word of each phrase. There are only four options. Namely: leave intact as is in dictionary, convert all characters to lowercase, convert all characters to uppercase or capitalize only the first letter of the word.
- 3. *Word separator*. It may be absent. Then all the words will be concatenated. Example: 'aaabbb', 'aaaccc','aaaddd', etc. You can otherwise set a custom separator; e.g. the '-' character: 'aaa-bbb', 'aaa-ccc','aaa-ddd'. Or you can set a range of characters.
- 4. Other words. With this attribute, similarly to 2., you can set rules for the other words of a phrase.
- 5. *Postfix* text that will finalize each phrase. For example, if you set Postfix to the '?' or ' ?', all phrases created with this rule will have the question mark at the end.

Certainly, the more password generation rules you set, the more chances you have to pick the right password. But, on the other hand, the more time you will have spent on the attack.

The 'Statistics' group shows the average and recommended average size of a dictionary, number of words in source dictionaries, total number of passwords being generated and other helpful information.

Dictionary generator
	Combined d	lictionary attack op	tions		x
Dictionaries	Creation rules	Pass-phrase generat	or Online dicti	ionaries	⊳
Dictionary g	enerator - create a	pass-phrases wordlis	:		
Generated by this attact use it in another program wordlists given and crea <u>Read more about combin</u>	<pre>< passwords can eas n. Be careful, diction tion rules set. ned dictionary attack</pre>	ily be saved to file. So ary creation may take	you can create your quite some time dep	own dictionary a ending on the so	and ource
Dictionary generator -					
Dictionary size	233 401 014 st	rings		Generate	
Statistics Output passwords Rules/combinations Size (strings)	233 401 014 6 Prim1=6237, se	ec2=6237			
			OK Car	ncel	

The third tab of options serves for creating combined attack-based dictionaries (available not for all editions).

You can also download additional dictionary modules from the Passcape Software Web site.

2.8.2.9 Pass-phrase attack

More and more users choose to make up their pass phrases of entire phrases, passages from poems, movie aphorisms, Latin aphorisms, etc. Attempting to recover such passwords using the traditional techniques is unthinkable, even with the reference to the advancement of the computing power of modern computers. Therefore, the recovery help comes with the predefined and known phrase attack.

Pass-phrase attack is by much similar to the simple dictionary attack, except that here the password search goes phrase by phrase instead of going word by word. The main idea of the attack is to guess the right password by searching through predefined frequently used expressions, phrases and word combinations.

For example, if the sought password is made of the widespread phrase 'To be or not to be', it is obvious that this is the only attack that has the virtue to cope with such a password. In order to do that, you are

to specify a special pass-phrase dictionary. A simple phrase dictionary comes with the software, but you can also <u>download the online dictionaries</u> that were compiled specifically for this attack.

It wouldn't be an overestimation to say that 99 percent of the success in the recovery of a password with a dictionary attack depends on the quality of the dictionaries. Most likely, that is the reason why this type of attacks doesn't appear in just about any password cracker. Passcape Software allows utilizing a whole set of online and offline dictionaries (totally over 500 MB) compiled specially for this type of attack.

For example, many users make their passwords of excerpts from their favorite songs or music bands. That's why we have created special, unique (you won't find anything like that anywhere on the Net!) music-oriented key phrase sets. There's also a biblical set, movie phrases, proverbs, etc.

Windows Password Recovery comes with a short dictionary of phrases and aphorisms.

Phrase dictionaries

Pass-phrase attack options						
Phrase dictionaries Phrase mutation			Dictionary generator	Online dictionaries		
Pass-phrase dictionaries list						
The main idea of the pass-phrase attack (developed by Passcape) is to guess the right password by searching through predefined frequently used expressions, phrases and word combinations. Similar to the simple dictionary attack, from the source dictionary we sequentially take a phrase and attempt to match with the original password. Read about Passcape Wordlist Collection						
Dictionary name	Dictionary size	Phrases	Full path			
	73 372	9 286	E: \Program Files \Passo	ape \WPR \dic \phrases.pc	d	
Add	Remove		Мот	ve up Move down		

The password-phrase attack options almost completely repeat the simple dictionary attack options: here, you also are to select one or several dictionaries for the phrase source, it also allows loading additional dictionaries from the Passcape website, and it has the same way for setting phrase mutation rules (creating alternative options).

Phrase mutation

Pass-phrase attack options			×
Phrase dictionaries	Phrase mutation	Dictionary generator	Online dictionaries
Pass-phrase muta	ation		
Weak mutation is normally just of large sizes. Medium mutatio generated password phrases. range of all possible combinati using the national keyboard la <u>Read more about pass-phrase</u>	ified in only one case: find in only one case: for n is a normal balance by Strong mutation allows ons, to the prejudice of yout, abbreviations, etc attack	or increasing the attack spee etween the operating speed finding more difficult passwor the search speed. For instan	ed or when using dictionaries and the number of ds by generating the widest ce, English phrases typed
Mutation level			
✓ Enable smart mutation		0.1 h h i i i i i	
() Weak (fast)		Oltra light (extremel)	ly slow)
Normal (slow)		O Ultra normal (extrem	nelly slow)
Strong (very slow)		Oltra hard (extreme	lly slow)
Phrase limitation Imit input phrase Maximal phrase length	100		
Maximal words in obras	se: 10	1	
Maxina words in philas	10		
		ОК	Cancel

Mutation is worth saying more, since as you should have known strong mutation significantly raises chances for the successful recovery. Weak mutation is normally justified in only one case: for increasing the attack speed or when using dictionaries of large sizes. Medium mutation is a normal balance between the operating speed and the number of generated password phrases. Strong mutation allows finding more difficult passwords by generating the widest range of all possible combinations, to the prejudice of the search speed. The greater is the mutation level, the more passwords the attack will cover. For instance, English phrases typed using the national keyboard layout, abbreviations, etc.

Major difference in mutation levels:

- Weak simplest thus fastest mutations.
- Normal the same as Weak, but generates several additional mutations and case combinations.
- Strong the same as normal plus more mutations and national passwords (according to the installed keyboard layouts, if any).
- Ultra light this is a 2-step mutation because every generated in Weak mode password goes through the second mutation round (one used in Weak mode of the simple dictionary attack).
- Ultra normal 2-step mutation. Every password generated in Normal mode is used as a source to generate additional combinations by implementing additional Normal mutation level.

Be careful! Ultra modes generate a great number of passwords, thus the attack may be ran extremely slow. To speed up the attack, consider setting up input phrase limits. For example, you can limit input phrases to 10 words and 100 characters.

Dictionary generator

Pass-phrase attack options	x
Phrase dictionaries Phrase mutation Dictionary generator Online dictionaries	Þ
Dictionary generator	
Generated by this attack passwords can easily be saved to file. So you can create your own dictionary and use it in another program. Be careful, dictionary creation may take quite some time depending on the source wordlists given and creation rules set.	
Read more about pass-phrase attack	
Dictionary generator	
Dictionary size unknown Generate	
OK Cancel	

The third tab uses for creating pass-phrase dictionaries.

2.8.2.10 Rainbow tables attack

A rainbow table is a lookup table offering a time-memory tradeoff used in recovering the plaintext password from a password hash generated by a hash function, for example Windows passwords.

This is quite a sophisticated password audit tool. This method was developed by Philippe Oechslin for quick recovery of password using precalculated tables. It's enough to say that the sought password can be recovered within minutes or even seconds.

	Rainbow	attack optio	ns		
LM rainbow tables option	ns NTLM ra	inbow tables o	ptions		
LM rainbow tables lis	st				
Rainbow attack is an implementati method developed by Philippe Oe several minutes or even seconds, can be recovered, although with a Read more about Rainbow table a	on of the Faster chslin. Recoverir You should awa a success probal <u>Ittack</u>	r Cryptanalytic T ng complex pass re that due to th pility up to 100%	ime-Memory Trad words with the ra he nature of the a b. You should set i	le-Off passw inbow attack attack, not a LM specific t	vord auditing k is a matter of Ill passwords ables here.
Table name	Table size	Chains	Success rate	Full path	
Im_alpha-numeric#1-7_0_24 Im_alpha-numeric#1-7_1_24 Im_alpha-numeric#1-7_2_24 Im_alpha-numeric#1-7_3_24	976 479 408 966 925 008 983 924 640 804 082 528	61 029 963 60 432 813 61 495 290 50 255 158	72.541% 72.284% 72.740% 67.271%	C:\Passca C:\Passca C:\Passca C:\Passca	pe\Rainbows\R1 pe\Rainbows\R1 pe\Rainbows\R1 pe\Rainbows\R1
Add Remove	1	11	Mor	/e up	Nove down
			Tabalance		
Limit RAM usage			l otal succ	ess race;	99.321%

The program supports the standard *.rt, indexed *.rti and hybrid tables. Multithreading is supported as well.

It must be mentioned that rainbow attack does not guarantee the recovery of all passwords, but the probability of the recovery is close to 100%, depending on the tables you've got.

A specific rainbow table can be implemented for the hash it was created for. Eg. LM specific tables should be used for breaking LM hashes only.

The attack options allow limiting the amount of RAM that can be utilized by the attack when using old computers (the attack assumes using large volumes of RAM for its calculations).

2.8.2.11 Hybrid dictionary attack

Hybrid dictionary attack is a form of <u>simple dictionary attack</u>. However, unlike the latter, hybrid attack allows user to set his own word mutation (variation) rules and attempt to validate the modified words as source passwords. For example, user could capitalize the first letter of a password being validated, append '2' to it, replace the number 8 in it with the letter B, O with 0, etc.

Actions, performed on source words from the dictionary, are called rules. Multiple rules can be applied to each source word. The rule definition syntax is compatible with John the Ripper and PassworsPro software. The author of the latter has kindly provided an extended set of rules, slightly edited, which comes with the distribution kit for Windows Password Recovery.

Hybrid dictionary attack settings are grouped in 7 tabs:

- 1. Dictionaries for setting up source dictionaries.
- 2. Rules files with set of rules.
- 3. Super-rules ones to be applied over the top of regular rules
- 4. Dictionary generator, where you can create files of words obtained from the hybrid attack.
- 5. Online dictionaries for downloading new dictionaries to the application.
- 6. Hybrid syntax complete description of all rules with examples.
- 7. Rule tester, where you can test your rules.

Wordlists to be used in the attack are set on the first tab. Traditionally, the application supports wordlists in ASCII, UTF8, UNICODE, PCD, RAR and ZIP format. The position of the files on the list can be altered. For example, you may want to move smaller dictionaries up the list or the other way. During the attack, they will be used one after another, according to their position on the list.

	Hybr	rid dictionary atta	ack options				
	Rules Supe	r-rules Dictio	nary generator 🔰 Online dictionaries 🚺 🕨				
Hybrid at	Hybrid attack dictionaries						
Lisheid attacks is simila			t use devide at his own word to take a day				
Setup at least one w	ordlist here in order to	continue.	at user should set his own word mutation fules.				
Read more about Hy	brid dictionary attack						
Dictionary name	Dictionary size	Strings	Full path				
WPR.pcd	6 290 880	2 150 822	C:\Program Files (x86)\Passcape\W				
Danish.dic	7 653 774	564 295	C:\SYS\PWC01\01 - Languages\Dani				
Dutch.dic	3 120 165	245 688	C:\SYS\PWC01\01 - Languages\Dutc				
English.dic	38 302 216	3 540 673	C:\SYS\PWC01\01 - Languages\Engli				
French.dic	3 822 281	320 033	C:\SYS\PWC01\01 - Languages\Fren				
German.dic	21 367 379	1 786 211	C:\SYS\PWC01\01 - Languages\Ger				
Japanese.dic	1 050 978	115 650	C:\SYS\PWC01\01 - Languages\Jap				
Norwegian.dic	12 658 942	809 552	C:\SYS\PWC01\01 - Languages\Nor				
Add	Add folder Rem	ove	Move up Move down				
OK Cancel							

On the '**Rules**' tab, define at least one file with password mutation rules. The format of the rules file is quite trivial; it is a plain-text ASCII file with the '**[Rules]**' string. Anything above this string is considered as comments and ignored by the program. Whatever goes below this string is considered as rules. Each string can contain several rules, applicable to a source word. The exclusion is the **aN** rule. This rule must not be on the same line with other rules. If a string contains multiple rules per word, those rules are parsed left to right. For example, if you apply the rule '@pc\$a\$b\$c' to the source word 'password', at the output you will get 'Asswordabc'. The maximum length of an output word may not exceed **256** characters.

	Hybrid	d dictionary attack options	
♦ Dictionaries	Rules Super-	rules Dictionary generator Online dictionaries I. >	
Hybrid attack rules You should set at least one file with word mutation rules. Rule file is a simple ASCII text file with [Rules] section in it. Everything below the string is considered as rules. See the syntax of the rules at the Syntax tab. You can save all rules into a single file (rules will be sorted, duplicates and errors will be skipped).			
File name	Rules count	Full path	
yurets.ini	74585	C:\Program Files (x86)\Passcape\WPR\hybrid_rules\yure	
Add R Import Hashcat/Inside	emove pro rules	Move up Move down Save	
		OK Cancel	

'Super-rules' is a rule (or several rules) to be applied over the top of all other regular ones, before or after them. For example, you can set 'a8' tail super-rule to create all possible case combinations after a common mutation has been done. So '/asa4' rule from I33t.ini file will become '/asa4a8', '/csc(' will become '/csc(a8', etc. Yet another one example: setting the '>6<G' head rule allows you to skip all words of less than 6 or greater than 16 characters, before starting a common mutation. This is a helpful feature once you decide to add the same rule to all text lines of the selected *.ini files. There's no need to modify them all. Be careful though, the 'aN' super-rule may increase the total number of generated passwords drastically.

Hybrid dictionary attack options
Dictionaries Rules Super-rules Dictionary generator Online dictionaries
Super-rules to be automatically append or prepend to every line of common rules
'Super-rule' is a rule (or several rules) to be applied over the top of all other regular ones, before or after them. If you set a HEAD super-rule, it is prepended to every line from the given *.ini file. If you set a TAIL super-rule, it will be automatically appended to the every line of common rules.
Read more about Hybrid dictionary attack
HEAD super-rule
This super-rule is applied BEFORE every line of common rules
TAIL super-rule
This super-rule is applied AFTER every line of common rules >6 <g< td=""></g<>
OK Cancel

The '**Dictionary generator**' tab is designed for generating dictionaries obtained from an attack. Further on, those dictionaries could be used, for example, in other applications. To generate a dictionary, specify a source dictionary and a set of mutation rules for it. The size of a target file may exceed 2 GB. Be careful, the dictionary generation process may take considerable time!

Hybrid dictionary attack options
Dictionaries Rules Super-rules Dictionary generator Online dictionaries
Dictionary generator
Passwords which were generated by this attack can easily be saved to file. Therefore, you can create your own dictionary and use it in another program. Be careful, dictionary creation may take quite some time depending on the source wordlists given and creation rules set.
Read more about Hybrid dictionary attack
Dictionary generator
Dictionary size unknown Generate

You can download additional wordlists for the attack using 'Online dictionaries' tab.

If you want to create your own set of rules, you can use the last two tabs as sources of help. While the '**Hybrid syntax**' tab gives mere descriptions of available rules, on the last tab you can actually test them by specifying a source word and a rule for the hybrid attack. Forward your rule sets to us; if we find them interesting/useful, we will include them in the default distribution of the program.

Program's interface

Hybrid dictionary attack options						
Dictionary generator Online dictionaries Hybrid syntax Rule tester						
Hybrid rules	Hybrid rules tester					
Read more about Hybrid	dictionary attack					
Dula tastas						
Input word	password					
Hybrid rule	@pc\$a\$b\$c	Þ				
Output word	Asswordabc					
	OK Cancel					

Rules description for the hybrid dictionary attack

Several rules at a line are allowed to be set.

Rules (if any) are processed from the left to the right.

Maximal line length is limited to 256 characters.

Maximal output word length is limited to 256 characters.

White space is ignored as long as it is not used as a parameter.

A line started with # character considered as a comment.

All text before the '[Rules]' line is considered as comment.

N and M always start at 0. For values greater than 9 use A..Z (A=10, B=11, etc.).

The following rules should be at the last position of a line: aN, ?iN[C], ?i[C], ?oN[C], ?o[C], ?iZ[C], ?oZ[C].

Don't change the names of the standard rule files. Some ones are used by the program.

?iN[C], ?i[C], ?oN[C], ?o[C] ?iZ[C], ?oZ[C] rules use the following predefined charsets (you can use custom character sets though):

digits	- 0123456789
loweralpha	- abcdefghijklmnopqrstuwxyz
upperalpha	- ABCDEFGHIJKLMNOPQRSTUVWXYZ
alpha	 abcdefghijkImnopqrstuwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
special	- !@#\$%^&*()+=~`[]{} \:;"'<>,.?/ "

loweralphanumeric - abcdefghijklmnopqrstuwxyz0123456789

 $upper alpha numeric\ -\ ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789$

alphanumeric - abcdefghijkImnopqrstuwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 printable

abcdefghijkImnopqrstuwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@#\$%^&*()-_+=~`[]{}| \:;;"'<>,.? /

<u>I (ui</u>	53			
Rul e	Exa mple	Input	Output	Description
:	:	passwoi d	password	Do nothing to the input word
{	{	passwoi d	asswordp	Rotate the word left
}	}	passwoi d	dpasswor	Rotate the word right
Ľ	[passwor d	assword	Delete the first character
]]	passwor d	passwor	Delete the last character
С	с	passwor d	Password	Capitalize
С	С	passwor d	pASSWOR D	Anti-capitalize (lowercase the first character, uppercase the rest)
d	d	passwor d	passwordp assword	Duplicate word
f	f	passwor d	passworddr owssap	Reflect word
k	k	passwoi d	gfhjkm	Convert word using alternative (first after default) keyboard layout. The rule works in both directions. For example, if there's Russian keyboard layout installed previously in the system, the rule should convert word 'password' to Russian ' ', and Russian word ' ' to 'gfhjkm'. This is very helpful when looking for non-English passwords. If only one language is installed in the system, the rule does nothing.
К	К	passwoi d	passwodr	Swap last two characters
I	I	passwoi d	password	Convert all characters to lowercase
q	q	passwoi d	ppaassssw woorrdd	Duplicate all symbols
r	r	passwoi d	drowssap	Reverse word
t	t	PassW ord	pASSwOR D	Toggle case of all characters
u	u	passwor d	PASSWOR D	Convert all characters to uppercase
U	U	my own passwoi d	My Own Password	Capitalize all words delimited with space (upper-case the first character and every character after a space)
v	V	passwoi d	PaSSWoR D	Vowels elite

۰.,	~~	, .	•
Ru	امع		

Program's interface

Rul e	Exa mple	Input	Output	Description			
v	V	passwor d	pASSWoR D	Vowels noelite			
'N	'4	passwor d	pass	Fruncate the word to N character(s) length			
+N	+1	passwor d	pbssword	Increment character at position N by 1 ASCII value			
-N	-0	passwor d	oassword	Decrement character at position N by 1			
.N	.4	passwor d	passoord	Replace character at position N with character at position N+1			
,N	,1	passwor d	ppssword	Replace character at position N with character at position N-1. Where N > 0 .			
<n< th=""><th></th><th></th><th></th><th>Reject (skip) the word if it is greater than N characters long</th></n<>				Reject (skip) the word if it is greater than N characters long			
>N				Reject (skip) the word if it is less than N characters long			
aN				Check all possible symbol cases for the word. N is a maximal length of the word to apply this rule for.			
DN	D2D2	passwor d	paword	Delete the character at position N			
рN	р3	key	keykeykey	Copy word N times			
TN	T1T5	passwor d	pAsswOrd	Toggle case of the character at position N			
уN	у3	passwor d	paspaswor d	Duplicate first N characters			
YN	Y3	passwor d	paswordord	Duplicate last N characters			
zN	z3	passwor d	ppppasswo rd	Duplicate the first character of the word N times			
ZN	Z3	passwor d	passwordd dd	uplicate the last character of the word N times			
\$X	\$0\$0\$ 7	passwor d	password0 07	Add character X to the end of the word			
^Х	^3^2^ 1	passwor d	123passwo rd	Insert character X at the beginning of the word			
@X	@s	passwor d	paword	Remove all characters X from the word			
!X				Reject (skip) the word if it contains at least one character X			
/X				Reject (skip) the word if it does not contain character X			
(X				Reject (skip) the word if the first character is not X			
)X				Reject (skip) the word if the last character is not X			
éX	e@	mike@y ahoo.co m	mike	Extract a substring starting at position 0 and ending up before first occurrence of X character (do nothing if X is not found)			
EX	E@e.	mike@y ahoo.co m	yahoo	Extract a substring starting right after first found X character and till the end of the string (do nothing if X is not found)			
% MX				Reject (skip) the word if it does not contain at least M instances of the character X			

Program's interface

Rul e	Exa mple	Input	Output	Description
*ХҮ	*15	passwor d	possward	Swap characters at positions X and Y
=N X				Reject (skip) the word if the character at position N is not equal to the X
iNX	i4ai5b i6c	passwor d	passabcwo rd	Insert the character X in position N
oN X	o4*o5 *	passwor d	pass**rd	Overwrite a character in position N with the character X
sXY	ss\$so 0	passwor d	pa\$\$w0rd	Replace all characters X with Y
xN M	x4Z	passwor d	word	Extract a substring of up to M characters length, starting from position N.
INX -Y	r10/-/r	google. com	google.com /	Insert the character X at position N if previous character at position N is not Y.
INX +Y	rl0.+.r	passwor d.	password	Insert the character X at position N if previous character at position N is Y .
ON X-Y	О0- +р	passwor d	-assword	If the character at position N is not Y, overwrite it with X character.
ON X+ Y	00Р+ р	passwor d	Password	If the character at position N is Y, overwrite it with X character.
RN M+ Y	R01+ a	passwor d	assword	Remove character at position N if character at position M is Y
RN M-Y	R40-b	passwor d	passord	Remove character at position N if character at position M is not Y
? iN[C]	? i0[digi ts]	passwor d	0password, 1password 9password	Insert a character from a charset [C] into position N of the word. Where C should be either a predefined charset name or a custom character set itself.
? iZ[C]	? iZ[digi ts]	passwor d	password0, password1 password9	Insert a character from a charset [C] into last position of the word. Where C should be either a predefined charset name or a custom character set itself.
? i[C]	? i[spec ial]	passwor d	~password, !password password_, password+	Insert a character from a charset [C] into every position of the word. Where C should be either a predefined charset name or a custom character set itself.
? oN[C]	? o1[up peralp ha]	passwor d	pAssword, pBssword pZssword	Overwrite a character at position N with a character taken from a charset [C]. Where C should be either a predefined charset name or a custom character set itself.
? oZ[C]	? oZ[up peralp ha]	passwor d	passworA, passworB passworZ	Overwrite a character at last position with a character taken from a charset [C]. Where C should be either a predefined charset name or a custom character set itself.

Rul	Exa	Input	Output	Description
е	mple			
?	?o[-	passwor	-assword,	Overwrite a character at every position of the word with a character taken
o[C	=.]	d	=assword	from a charset [C]. Where C should be either a predefined charset name
]			passwor.	or a custom character set itself.

Additional

Windows Password Recovery distribution kit comes with extended sets of password mutation rules: **hybrid rules/english words.ini** file contains basic rules for English passwords.

hybrid_rules/nonenglish_words.ini holds common rules for non-Eglish passwords.

hybrid rules/simple dates.ini - a lot of rules with dates, months, seasons, etc.

hybrid_rules/I33t.ini - rules to freak words (based on leet dictionary). For example, password->p@\$\$w0rd

•••

Looking for a convenient way to handle as much passwords as possible? Downloading the <u>full set of</u> more than 180000 sorted and duplicate-free rules.

2.8.2.12 Online recovery

Online recovery (developed by Passcape Software) finds passwords using Internet search engine servers. It deals fairly well with short and frequently-used passwords. Among its drawbacks are low operating speed and poor suitability for handling large hash lists.

Online recovery has been developed by Passcape Software and is an improved online password finder. To find passwords, the program consecutively submits a special search request for each hash to a search engine and then downloads the password files found and analyzes their contents. Online recovery is relatively slow; therefore, it is appropriate for small hash lists. In addition, the passwords found are usually limited to simple vocabulary and short combinations. One way or the other, this attack can be quite useful; for example, when auditing passwords, as a simple vulnerability detector for certain systems.

Online recovery optior	ıs			۲
Online	recovery performs v	vorldwide password	d search	
Online recovery search well-known passwords,	es passwords in onlin for small list of hashe	e databases. This att s. It also may acts as	ack is effective against simple, short or a password weakness indicator.	
Read more about online	e recovery			
Attack settings				
Search full LM ha	shes (recommended),	halves otherwise		
Maximize lookup e	effeciency			
Skip unnecessary	rtiles			
V Set response time	out (in seconds)			
	for a second data	manta		
	ze for pages and docu	Iments		
1	MD			
Use proxy				
Server				
Port	0			
Min delay between se	earch queries	10	seconds	
Max delay between s	earch queries	20	seconds	
	ОК	Cance	2	

Online recovery options

- Search full LM hashes use the entire 16-byte hash when searching LM hashes. If this option is not set, the search will be carried out over the 8-byte halves. To ensure more efficient search and get rid of some stray traffic, it is recommended that this option is set. It is ignored when searching NT hashes.
- **Maximize lookup efficiency** increase password lookup efficiency not affecting the attack speed. It is also recommended to always set this option.
- Skip unnecessary files do not check some unnecessary files if they are suspected to not contain passwords.
- Response timeout set the maximum allowed Web resource response time.
- Limit download size limit download file size. Some hash databases have enormous size, even despite that often they do not contain passwords. Therefore, for slow Internet connections and to restrict stray traffic, it is recommended to set a limit on the size of download pages. Unfortunately, there is no way to figure out what is in the data to be downloaded; therefore, this option is determined exclusively by your preferences and capabilities.
- Use proxy use proxy server for looking up passwords
- Min/max delay between search queries minimum and maximum delays between two consecutive requests to the search server. Some search servers may reject search requests if they go in series from the same IP address with a very short time interval (normally less than 10 seconds). Despite that Windows Password Recovery has an internal request randomizer, which allows to drop this delay significantly (to as little as 1 and 2 seconds respectively), the safe values when a search request will be definitely processed by the server are min=15 and max=30 seconds. Certainly, the attack speed depends is in direct relation to these two options.

Be careful! Online recovery may generate a lot of Internet traffic!

2.8.2.13 Passcape table attack

Passcape Rainbow Tables are the next logical development of simple pre-calculated tables. They are most suitable for the recovery of meaningful combinations and complex passwords of literally unlimited length.

The original method of simple rainbow tables

The operating principle of simple rainbow tables consists of setting a character range (for example, a..z) and maximum password length, followed by the calculation of all the possible variants and the generation of millions of chains. Each chain is calculated by the formula:

P0 -> hash(P0) -> H1 -> R(H1) -> P1 -> hash(P1) -> H2 -> R(H2) -> P2 ...

where P – password, **hash** – hashing function, R – reduction function. Thus, from the original password, the hashing function produces a hash, which the reduction function then converts into the next password, and the process repeats all over again and generates chains. Each chain stores only the original and final value. Storing only the first and the last hash is an operation leading to compromise and saving memory at the cost of time spent on cryptanalysis.

To recover a sought password, it undergoes hashing and the reduction function and then is looked up in the table. For that purpose, a key chain is generated beginning with R(Hn) up until the maximum chain length. If Hn is obtained with the password used when creating the table, we finally get the key that matches the key of the respective chain. This last key was saved in the table along with the first key of the chain. Using the first key of the chain, we can recover the entire chain, in particular, the value right before R(Hn). That is actually the key that was used for generating Hn, our sought password.

Operating principle of Passcape rainbow tables

Recovery using Passcape rainbow tables is pretty much the same as recovery using simple rainbow tables. However, unlike the latter, it is sort of a hybrid of <u>Fingerprint</u> and <u>simple table</u> attacks, where instead of setting a specific character range passwords are validated within a so-called 'word footprint' range. The idea of the Fingerprint attack developed at Passcape comes down to taking the source dictionary and creating a bank of word footprints (fingerprints), necessary for validating the password, out of that dictionary; then, during the attack, we search for all possible variants of words that consist of two such footprints.

Similar to the Fingerprint attack, Passcape rainbow tables first create a bank of footprints for words from a user's wordlist. The word footprint bank is an analog to character set in simple rainbow tables. It is used for both creating Passcape tables and validating passwords. Thus, a Passcape rainbow table consists of one or more ***.prt** files (the actual tables) and a bank f word footprints (***.prti**), which can be engaged only with tables that were created with it.

There are a number of advantages in using word footprints instead of character sets when creating tables:

• The length of passwords validated with Passcape tables is literally unlimited. Unlike with simple rainbow tables, which practically cannot be created for passwords longer than 9 characters, with Passcape tables one can recover both one-character and 50-character password with same probability.

- Character set in the regular table greatly affects its critical parameters: the wider the character range, the greater the chain length or the total number of chains for storing success rate (percentage of success in finding password) of the table must be. In a Passcape table, a character set does not affect the critical parameters of the table.
- Plain tables have certain difficulties when generating tables for validating passwords in national character sets; not all programs properly handle such tables, and not all can create them. With Passcape rainbow tables, when generating tables, for example, for Russian passwords, one can simply specify the source dictionary in Russian.
- With Passcape tables, passwords are searched for using more meaningful combinations; however, that largely depends on the source dictionary.

These can be referred to as drawbacks of Passcape rainbow tables:

- Not all source dictionaries are equally suitable for the tables. Using large dictionaries (normally greater than 1 MB) generates too large of a footprint bank; respectively, creating tables may require significant time and resources.
- Using dictionaries with long words or phrases is discouraged due to the above mentioned reason.
- Rainbow table attack consumes a great deal of resources: the footprint bank must fully fit the computer's RAM.

Passcape rainbow table attack settings

Passcape rainbow table attack settings are rather trivial. Specify one or several *.prt tables, which should reside in the same directory as the footprint bank (*.prti file). Since this attack consumes more RAM than the attack that uses simple rainbow tables, it is recommended to limit the amount of RAM that can be consumed by adjusting the respective option.

Passcape rainbow tables attack op	otions				×
Passcape LM rainbow tab	les list Pas	scape NT rainb	ow tables list		⊳
Passcape NT rainbow	tables list				
Passcape rainbow attack is a smar unlike simple rainbow tables, Passo passwords. You should aware that depending on the initial wordlist use specific tables here. <u>Read more about Passcape rainbo</u>	t implementation cape rainbow att due to the natu ed to create the <u>ww table attack</u>	of the Time-Mer ack can recover re of the attack, tables and table	mory Trade-Off pa r long (up to 64 ch not all passwords generation option	ssword audit laracters) and can be recov s. You should	ing method. But d complicated vered, d set NTLM
Table name	Table size	Chains	Success rate	Full path	
nt#0_6400x40000000#word	640 000 000	40 000 000	84.713%	G:\PRT\wo	rds-english\nt#
nt#1_6400x40000000#word	640 000 000	40 000 000	84.713%	G:\PRT\wo	rds-english\nt#
nt#2_6400x40000000#word	640 000 000	40 000 000	84.713%	G:\PRT\wo	rds-english\nt#
nt#3_6400x40000000#word	640 000 000	40 000 000	84.713%	G: \PRT \wo	rds-english \nt#
•	1	1			F.
Add Remove			Mov	ve up	Move down
🔽 Limit RAM usage			Total succ	ess rate:	99.945 %
128 Mb 🔹			Total tal	oles size:	2.4 Gb
			ок	Cancel	

Tables can crack only the hash function they were created for, i.e. NT tables can crack only NT hash!

To create your own tables, you can take advantage of the <u>respective tool</u>. You can download sample Passcape tables for this attack from our website.

2.8.2.14 Batch attack

Since each attack covers its own password range, sometimes, in order to fully recover password hashes, you have to run several different attacks one after another. The basic idea behind the batch attack (developed by Passcape Software) is to create a list/batch of attacks to be run one after another, so that you could launch all those attacks with a single click of the mouse and not hassle with configuring each of them individually every time you need them.

	Batch at	ttack ×
Setup your own bur	dle of preferred attacks that v	vill be executed subsequently, one-by-one
Batch attack is a set of several at different options. <u>Read more about batch attack</u>	tacks that will be executed one-	-by-one. You can include any attack more then once with
Attack type	Complexity	Attack properties
Preliminary attack	1	Estimated time: instantly
Mask attack Mask attack Combined dictionary attack Phrase attack Brute-force attack Brute-force attack Brute-force attack Brute-force attack	0 0 4 0 3 3 4 6	Simple brute-force attack: Yes Simple dictionary attack: No Extended dictionary attack: No Attack repeatable sequences: Yes Attack simple patterns: Yes Attack complex patterns: Yes Attack keyboard combinations: Yes Attack heyboard combinations: Yes Attack national keyboard combinations: Yes Complex keyboard attack: Yes Passcape Password Prediction attack: No User name mutation attack: Yes
		<u>نې</u>
	ОК	Cancel

The batch attack options are available as a list that you can extend or cut (buttons [+] and [-]). Each attack on the list can be moved up or down (buttons $[^{\wedge}]$ and [v]), and its settings can be edited. A batch can include several attacks of the same kind, but of the attacks can have different settings. The pane to the right of the selected entry displays the properties of the selected entry; brief specifications of the attack and the estimated time the attack will take to complete.

2.8.2.15 GPU: Brute-force Attack

A GPU brute force attack is fully identical to a <u>regular brute force attack</u>, except that passwords are searched by the graphics processing unit of your PC instead. It is no secret that the performance of modern graphics cards is an order of magnitude greater than that of CPUs; this makes them a convenient tool for heavy calculations, such as password recovery. It is important to understand that calculations using graphics cards have a number of disadvantages. For example, some algorithms with a great number of conditional jumps and other checks demonstrate extremely poor performance on GPU, and in certain cases it may be even lower than on a regular CPU.

Anyway, the software supports brute force password search using GPU. You can compare the performance indicators of GPU vs. CPU calculations through the respective menu item of the application or present it visually through the '**Reports**' menu.

The configuration of GPU brute force attack consists of three parts:

- 1. Choosing a character set for the search.
- 2. Specifying password length.
- 3. Configuring the graphics processing unit.

Choosing a character set for the search

When choosing a character set for a brute force attack, you are normally guided by empirical considerations. For example, if the expected password consists of lower-case Latin characters and digits, it makes sense to choose the range 'a-z, 0-9'. The smaller the character set, the sooner the attack completes.

On the other hand, there is always a chance to make a wrong choice of the expected character set. If at least one character of the password to be recovered is not included in the specified character set, the password will not be found.

At the bottom of the attack settings dialog, you can see the total number of passwords that match the specified character set and password length.

It is important to know that LM passwords in Windows are always converted to upper case; that significantly cuts the range of passwords to be searched!

Specifying password length

On the second tab of the options page, set the minimum and maximum length of searched passwords. As an alternative to minimum length, you can set the source password, which the search would begin with. The maximum length of LM in Windows operating systems is 7.

Configuring graphics processing unit

Before you can use it in an attack, you must first select the graphics card on the respective menu item.

GPU brute-force attack op	tions			×
	cter set Password lengt	h and position	GPU settings	Þ
GPU specific	options			
You can edit GPU specific simultaneously (in most ca sets the number of passw	e settings here. Thread blocks ses each block incorporates 2 ords to verify from within a sing	option sets the numb 56 GPU threads). W le GPU thread.	per of GPU blocks hile passwords pe	to be run r thread
Read more about GPU br	<u>ite-force attack</u>			
Thread blocks	25600			
Passwords per thread	Automatic			
		ОК	Cancel	

GPU configuration consists of only 1 parameter: the number of thread blocks to run on GPU. Each block consists of 256 threads. Thus, if you set the number of blocks to 25600, the GPU will run 25600*256=6553600 threads. Each GPU thread can check multiple passwords. The total number of checked passwords greatly depends on other options. Setting the **ThreadBlocks** parameter smaller than 10000 on modern graphics cards, in the majority of cases, leads to poor performance. To avoid performance degradation, after setting up the parameter and running the attack, make sure the GPU load chart has close to 100% plain graphic without peeks (see the screenshot below).



When running password recovery for domain cached credentials type 2, you may need to play around with this option to get better performance.

Setting too big value(s) may cause your GPU to hang or raise an error, depending on your <u>GPU kernel</u> timeout settings.

2.8.2.16 GPU: Fingerprint attack

Fingerprint attack is a brand-new tool for recovering complex passwords, which could not be decrypted in a common way. The idea of the attack is that here, to recover a password, we take neither individual words from the source dictionary, like in the Dictionary attack, nor even word combinations, like in the Combined attack, but so-called "fingerprints". So every word from the source dictionary is used for generating several fingerprints. If some password is found during the attack, it participates in generating new fingerprints, and the attack goes another round. Implementing GPU computing power allows to increase the recovery speed drastically. Fingerprint options consist of 4 parts:

General options

Before launching the attack, specify the source dictionary to be used for creating the fingerprints. The software comes with common.pcd dictionary, optimized for this attack, but you can use yours or download one off the Internet ('*Online dictionaries*' tab). There are no certain requirements to the source wordlist, except one: it must not be too large; otherwise, the attack will take significant time. You can use dictionaries with national passwords, if you suspect that the sought password contains characters in a national encoding.

GPU fi	ngerprint attack opt	ions			×
4	General options	Dictionary generator	Online dictionaries	GPU settings	Þ
	Fingerprint ge	neral options			
Finge passv effect <u>Read</u>	rprint attack uses inpu word is found, it then u tive if all additional mu I more about Fingerprin	It dictionary to generate all p Ises in further identification o tation options are set on. I <u>nt attack</u>	ossible variations for comp f more complicated passw	licated passwords. ords. The attack is v	Once a very
C:	al dictionary \Program Files\passca	ape\WPR\dic\common.pcd			
Ada V V	ditional mutation option Use "Passcape Pass Use keyboard and fre Use dates Use numbers and con Use extra word mutati Maximize efficiency w	ns word Prediction" engine to g quently used sequences nmon sequences ons (time-expensive) hen generating fingerprints	enerate additional source	passwords	
	Loop until no more pa Maximal password len	sswords are found Igth 16 ;			
			ОК	Cancel	

Here is the way the fingerprints are generated: first, a word from the source dictionary is broken into onecharacter passwords, then - into 2-character, etc. For instance, the source word **crazy** is broken into one-character fingerprints. So we get:

С
r
а
Z
У
Now, into two-character: cr ra az zy
Next, three-character: cra raz azy
And, finally, four-character:

craz razy

We have got 5+4+3+2=14 fingerprints, not counting the source word.

All word from the source dictionary are broken into fingerprints. After this, all the fingerprints are dumped into a single database, naturally, discarding duplicates. So we have got a database of fingerprints that would be used for checking passwords by gluing all the fingerprints with each other and finding the match.

The real fingerprint generation algorithm is a bit more sophisticated. Moreover, there is an option in the attack settings, **Maximize efficiency when generating fingerprints**, which maximizes the efficiency (at the expense of speed) by generating additional fingerprints.

Let's take a look at the remaining options.

- Use PPP engine to generate additional passwords use passwords found in other attacks when generating fingerprints.
- Use keyboard and frequently use sequences add keyboard combinations and common sequences to fingerprint bank.
- Use dates add dates to fingerprints.
- Use numbers and common sequences use digits and simple combinations of letters.
- **Maximal password length** this option allows limiting the maximum length of generated passwords. As a result, it reduces the time it takes for the attack.

The most careful attention should be paid to the option **Loop until no more passwords are found**. That is where fingerprint attack can really show itself off. Here is how it works: if at least one password is found during an attack, when the attack is over, the password participates in generating new fingerprints, and the attack runs again. This option works great on large lists of hashes and on password history hashes. However once the option is set, you will not be able to proceed the attack from the last saved position.

Dictionary generator

The second tab with the settings allows to create and save a custom dictionary using current options of the fingerprint attack. Be careful; the dictionary may take up a lot of space on your PC's hard disk drive.

GPU fingerprint attack options	×
d General options Dictionary generator Online dictionaries GPU settings	⊳
Dictionary generator - create fingerprint wordlist	
Generated by this attack passwords can easily be saved to file. So you can create your own dictionary an use it in another program. Be careful, dictionary creation may take quite some time depending on the sour wordlists given and creation rules set.	id ce
Read more about Fingerprint attack	
Dictionary generator	
Dictionary size unknown Generate	
OK Cancel	

Online dictionaries

On the third tab, you can download source wordlists for fingerprint attack from the Internet. Be careful, not all the dictionaries suit fine for the attack.

Passcape or	line dictionarie	5			
irst the program should	attempt to establ	ish a connection	with the Passoa	ane server an	d retrieve the list of
ictionaries available for	downloading. Se	elect the dictiona	ry you need and	click 'Downl	oad' button to retrieve
	- the state of the				
iead about Passcape o	niine dictionaries				
Dictionary name	D/L size	Real size	Strings	Ratio	Group
√ African.pcd	268 Kb	1 Mb	137821	16%	01 - Languages
American.pcd	82 Kb	399 Kb	45392	20%	01 - Languages
🗸 Australian.pcd	53 Kb	926 Kb	79409	5%	01 - Languages
Srizilian.pcd	89 Kb	1 Mb	129338	6%	01 - Languages
Chinese.pcd	1 Kb	5 Kb	802	32%	01 - Languages
Croatian.pcd	80 Kb	607 Kb	56558	13%	01 - Languages
Czech.pcd	377 Kb	3 Mb	309365	11%	01 - Languages
Janish.pcd	614 Kb	7 Mb	564295	8%	01 - Languages
Outch.pcd	393 Kb	2 Mb	245688	12%	01 - Languages
English.pcd	6 Mb	36 Mb	3540673	17%	01 - Languages
manufacture and	0.24175	C AND	400570	4.407	
•				T 1 1 (1 - OF	
lease select a dictionar	y to download			l otal files: 25	9, total size: 1 894 M
Update list	Download	Use loca	version		

GPU settings

Before you can use it in an attack, you must first select the graphics card in the General Options menu.

GPU fingerprint attack optic	ons	×
General options	Dictionary generator Online dictionaries GPU settings	Þ
GPU specific op	otions	
You can edit GPU specific s simultaneously (in most case sets the number of passwor	settings here. Thread blocks option sets the number of GPU blocks to be run es each block incorporates 256 GPU threads). While passwords per thread ds to verify from within a single GPU thread.	
Read more about GPU fing	erprint attack	
Thread blocks	256	
Passwords per thread	1000	
	OK Cancel	

GPU configuration is pretty simple and consists of only two parts:

- Setting the number of parallel graphics card's blocks, where passwords would be searched. Typically, each block consists of 256 threads. Thus, if you set the number of blocks to 256, the GPU will run 256*256=65536 threads. The total number of checked passwords for one call to GPU kernel will be 256*ThreadBlocks*PasswordsPerThread. In our case 256*256*1000 = 65 536 000 passwords. Setting the **ThreadBlocks** parameter smaller than 256 on modern graphics cards, in the majority of cases, leads to performance degradation.
- 2. Setting the number of passwords to be search from a single thread. The greater the value, the lower the overhead associated with launching threads, and the higher the search speed. However, setting too great a value may hang the computer or cause significant fluctuations in the current search speed, displayed on the attack status tab. This is caused by the fact that task completion time on the GPU exceeds the time required for refreshing the current state of the attack. Setting too big numbers may cause a system failure.

The **Password per thread** is not used and is always set to 1 when recovering domain cached credentials type 2.

When running password recovery for domain cached credentials type 2, you may need to play around with the **Thread blocks** parameter to get better performance.

2.8.2.17 GPU: Mask attack

Mask options

Mask attack is an irreplaceable tool when you know a fragment of the password or have any specific details about it. For example, when you know that the password consists of 12 characters and ends with the *qwerty*, it is obvious that searching the entire 12-character range of passwords is unreasonable (and useless, for it takes ages to complete). All what would be required in this case is to guess the first 6 characters of the sought password. That is what mask attack is for.

GPU mask attack options
Mask options Dictionary generator Mask tips GPU settings
Brute-force attack by a given mask
The mask attack is used if there's some information about the password. For example, you know that the password begins with 'loveme' and followed by a name. So you can set the mask 'loveme%c%c%c%c%c%c' to check all possible variants starting with 'lovemeaaaaaaa' and up to 'lovemezzzzzz'. You can set multiple masks within one attack. Read more about Mask syntax
Password masks
%c%c%c%c%cc%cqwerty
Custom character sets
Statistics
Password range aaaaaaqwerty zzzzzqwerty
Total passwords 308 915 776
OK Cancel

In our case, we could define the following mask: **%c%c%c%c%c%cqwerty**. That means that the program would successively check the following combinations: aaaaaaqwerty, aaaaabqwerty, aaaaacqwerty ... zzzzzqwerty. If the original password is *'secretqwerty'*, it perfectly hits the range.

The group of the Password Masks options is aimed to set a mask (or several ones), that will be used to generated passwords by. In most cases, if you know a part of the password, it is enough to specify a single mask. When a mask is selected, the statistics group shows the range of output passwords and the number of passwords generated by this mask. You can save your masks to disk to use it in another project, for example. The program also allows you to generate dictionaries by given masks.

Suppose that you have a list of decrypted passwords and you would like to generate mask templates

out of these passwords. Nothing easier. Run the mask generator and show the path to your password list there. You can sort the resulting masks alphabetically, popularity, or by search range (the fastest go first).

Generating multiple password masks	<
Generating multiple password masks	
To generate all possible masks, please supply a wordlist with real passwords. If the wordlist contains passwords in different languages, worth splitting it into several ones using the wordlist tools for example.	
Input wordlist with real passwords	
D:\Passcape\1\1.txt	
Sort output masks	
◯ Sort by alphabet	
 Sort by popularity (most frequently-used first) 	
◯ Sort by password range (fastest masks first)	
✓ Limit the maximal number of generated masks	
Max number of masks 100	
Generate Cancel	

Dictionary generator

By switching to **Dictionary generator** tab, you can generate your own dictionary by a given mask, and save it to disk. This feature available in Advanced edition of the program only.

PU mask attack option	s	X
Mask options	Dictionary generator Mask tips GPU settings	Þ
Dictionary	generator	
Often the mask attack of that the password begin mask loveme%c%c%c% Read more about Mask	ised if there's some information about the password to recover. For example, you kn is with "loveme' and followed by a word or a name. You can then set the following cc%c%c' to check all possible variants from "lovemeaaaaaa" to "lovemezzzzz". syntax	ow
Dictionary generator		
Dictionary size	~ 3 830 Mb Generate	
Statistics		
Password range	iloveAaaaaa iloveZzzzzz	
Total passwords	308 915 776	
	OK Cancel	

Mask tips

Third tab of the mask options contains a short description of the mask syntax and a couple of examples. The mask syntax is pretty simple and consists of static (unmodifiable) and dynamic (modifiable) characters. Dynamic characters always have a leading %. For example, if you set the mask **secret%d% d%d%d**, the program will generate 10000 passwords (secret0000, secret0001, secret0002 ... secret9999).

Windows Password Recovery supports the following dynamic mask sets:

- %c lower-case Latin characters (a..z), 26 symbols
- %C upper-case Latin characters (A..Z), 26 symbols
- %# full set of special characters (!..~ space), total 33 symbols
- %@ small set of special characters (!@#\$%^&*()-_+= space), 15 symbols
- %? all printable characters with ASCII codes of 32..127
- %* all ASCII characters (codes 1 through 255)
- %d one digit (0..9)
- %r(x-y) user-defined characters with serial ASCII codes between x and y

- %r(x1-y1,x2-y2...xn-yn) set of several non-overlapping sequences of ASCII characters. Useful for defining custom character sets; e.g., of OEM characters.
- %1[2,3..9] a character from user defined charset 1..9
- %% standalone static character %

Examples: test%d - password range test0..test9, 10 passwords total test%d%d%d~test0000..test9999, 10000 passwords test%r(0x0600-0x06ff) - test_.. test_, 256 passwords with Arabic character at the end %#test%# - _test_..~test~, 1089 passwords %1%1%1pin%2%2%2 - aaapin000.. zzzpin999, %1 is user-defined charset 1 (a..z), and %2 - the second user-defined charset 0..9 ilove%1%1%1%1%1 - iloveaaaaa .. iloveZZZZZ, %1 is user charset (a..z, A..Z)

The GPU mask attack syntax differs slightly from one used in a regular mask attack. The main difference is that in GPU-based attack you can not set numbers between x and y and can not set user-defined range of variable length, i.e. the following syntax will not work for GPU mask attack: %d(x-y) %1[2,3..9](min-max)

GPU settings

Before you can use it in an attack, you must first select the graphics card in the General Options menu.

PU mask attack options	-				×
Mask options	Dictionary generator	Mask tips	GPU settings		Þ
GPU specific	options				
You can edit GPU specific simultaneously (in most ca the number of passwords	c settings here. Thread blo ses each block incorporat to verify from within a single	cks option sets the es 256 GPU threa e GPU thread.	e number of GPU ds). While passwo	blocks to be run ords per thread se	ts
Read more about GPU fin	gerprint attack				
Thread blocks	10000				
Passwords per thread	Automatic				
			ок С	ancel	

GPU configuration for the Mask attack consists of only 1 parameter: the number of thread blocks to run at a single call to GPU. Each block consists of either 128 or 256 threads. Thus, if you set the number of blocks to 10000, the GPU will run 10000*256=2560000 threads for one call to GPU kernel. Each GPU thread can check multiple passwords. The total number of checked passwords depends greatly on other options. Setting the **ThreadBlocks** parameter smaller than 10000, in the majority of cases, leads to poor performance. To avoid performance degradation, after setting up the parameter and running the attack, make sure the GPU load chart has close to 100% plain graphic without peeks (see the screenshot of NVidia GTX 750Ti running with 15000 blocks).



When running password recovery for domain cached credentials type 2, you may need to play around with this option to get better performance.

Setting too big value(s) may cause your GPU to hang or raise an error, depending on your <u>GPU kernel</u> timeout settings.

If you feel lost and sick of all these stuff, then use the Mask Builder (in Password Recovery Wizard) that has a much more user-friendly graphical interface.

Pass	word recove	ery Wi	zard													\times
0		Mask	c build	ler											Step 3/4	
A ma eithe progr	ask consists o er from predefi ram will use th	of knov ined or he mas	vn and r from c sk to bi	l unkr custor rute-fo	nown pa n charse orce all p	rts (ele et. Build oossible	ments). d your pa e charad	Every ur assword cter com	nknowr mask t binatior	n eleme by simp ns you	ent ca bly add speci	n be rep ding one fied.	resent or mo	ted by a re mas	a character k elements	. The
Pa	ssword mask															
	a z		a :	z	а	z	а.	. z	а	. z	a	z	q	werty		
	Add know	wn	8	Ren	iove		ŧ۵.	Edit			+	Move le	ft	•	Move right	t
	Add unkno	own	Ē	Dup	icate											
Raw	mask:				%c%	c%c%	c%c%cc	qwerty								
Tota	l passwords:				308 9	15 77	5									
Pass	word range:				aaaaa	aaqwe	rty z	zzzząw	erty							
Pass	word example	e:			wvzy	gbqw	erty									
													<u>N</u> ext	>	Cano	el

2.8.2.18 GPU: Dictionary-force Attack

Oftentimes, when creating passwords, users add certain characters in the beginning, end or even middle of the word. To recover passwords of this specific kind, we have come up with a GPU-based dictionary attack, which is something between simple dictionary attack and brute force attack.

This attack works as follows:

- Reads the first word from the dictionary.
- According to the defined character set and the minimum/maximum length of the search range, generates all the possible variants.
- Those variants (characters) are then added to the beginning, end or middle of the word. The position within the word, where the generated sequences are to be inserted, can be specified at your discretion.
- Then goes next dictionary word, etc.

For example, if we specify a search character range between **0** and **9**, and the range length between **1** and **2**, the program will generate 100 combinations: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 ... 99. Then these sequences will be added to the beginning, middle or end of the word. Thus, for the word **test**, if the sequences are going to be inserted to every listed position, the program will check the following passwords:

Otest, 1test .. 99test t0est, t1est .. t99est te0st, te1st .. te99st tes0t, tes1t .. tes99t test0, test1 .. test99 Total - 100*5=500 variants.

Let's take a closer look at the attack settings.

Dictionaries

On the **Dictionaries** tab, you can specify the list of dictionaries to be used in the attack. The program supports text wordlists in the following formats: ASCII, UNICODE, UTF8, RAR, ZIP, as well as encrypted/packed dictionaries in the native PCD format, developed by our company. To deactivate a dictionary, simply clear the check box by its name. Thus, although the dictionary remains on the list, it will be ignored by the attack. The software comes with the default 400000-word dictionary. You can <u>order the full set of dictionaries</u>, that's over 6 GB in size, on CD or take advantage of the dictionaries available <u>online</u>.

PU dictionary force	options					×
1 Dictionaries	s Brute-force	Online	dictionaries	GPU settings		Þ
Diction:	aries list					
The GPU dictionary attack generates all combination to ever goes another word, <u>Read about Passca</u>	r is a hybrid attack I possible combinat ry position of the w etc. For complete ape Wordlist Collect	which actual ions using a r ord from dictio list of dictiona tion	y consists of 2 of range of symbol onary and check aries, check out	ones: dictionary and s from a given chars k the resulted word our 'Wordlist Collec	l brute-force. First th set, then it inserts ea as a password. The tion'.	e ich n
Dictionary name	Dictionary size	Strings	Full path			
wpr.pcd	756 682	416 713	E:\Program	Files \Passcape \WPI	R\dic\wpr.pcd	
Add	Remove			Move u	p Move dov	m
				ок	Cancel	

Search range

On this tab, you should set up the range of characters to be inserted into base words, its minimum and maximum length. When setting up a range, you can use the existing templates or, having checked the respective check box, define your own one. When selecting the maximum range length, keep in mind that specifying a too wide or too small value is inadvisable. While in the first case the password search speed may drop down to 0, specifying a too narrow range of characters to be searched raises the overheads related to the irrational use of the computing power of the GPU.

In the second group of options, specify the position in the word, where the characters of the searched range would be inserted.

And, finally, the third group of settings - these are in charge of preprocessing the words from the source dictionary. Selecting the **As is** option makes the program use the source word as is, not converting to upper or lowercase. The number of passwords to be searched grows in direct proportion to the number of options specified in this group. On the other hand, the program is smart enough to not use repeat words. For example, the word **12345678**, even if all the conversion options are set, will be used only once.

GPU dictionary force optio	ons		X					
Dictionaries	Brute-force Online dict	ionaries GPU settings	⊳					
A C Brute-force se	ettings							
Set up a character set, its symbols to be inserted into be inserted at.	Set up a character set, its minimal and maximal length. The attack will use the character set to generate symbols to be inserted into dictionary words. Insertion position is the position in the word where the symbols will be inserted at.							
Read more about GPU did	ctionary force attack							
Brute-force range								
Predefined charsets	az, AZ, 09, symbol 14, s	space						
Use custom charset								
Minimal length	1	Maximal length	4					
- Position in the word the	bruteforce symbols should be	e inserted at						
At the beginning	brateroree symbols should be	At the end						
Everywhere in the mid	Idle							
Input word utilization								
As is (no conversion)		Convert to lower case						
Convert to upper case	•	Capitalize (first char is upp	er, the rest are lower)					
		ОК	Cancel					

The number of passwords to be searched for a single word can be calculated using the following formula:

passwords = R * L * K

where

R - character range, calculated using the formula: R = charset_length $^max_length - charset_length ^ (min_length-1) +1$

L - positions in word. Calculated as follows: if the insertion is made in the middle of the word, L = password_length - 1; then add plus one if the insertion is made to the beginning and end of the word. K - number of options specified in the group 'Input word utilization'.

For example, if the source word we have is **window**, and the options are specified as shown on the image above, i.e. character range **a..z,A..Z,0..9,symbol14,space**, insertion to all positions, conversion to lowercase and capitalizing (first letter in uppercase). Let's calculate how many password we are going to check for this word:

charset_length = 26+26+10+14+1 = 77R = $77^{4} - 77^{0} + 1 = 35153041$ L = (6-1) + 1 + 1 = 7K = 2 passwords = 35153041 * 7 * 2 = 492 142 574
Online dictionaries

On the third tab, you can download source wordlists for the attack. The program uses internal wordlist (400000+ words) by default.

GPU dictionary force options							
♦ Dictionaries	Brute-force	Online dictio	naries GPL	J settings		Þ	
Passcape online dictionaries							
First the program should attempt to establish a connection with the Passcape server and retrieve the list of							
dictionaries available for o	downloading. Se	lect the dictiona	ry you need and	click 'Downl	oad' button to retriev	ve	
Read about Passcape or	line dictionaries	illiple selection.					
Dictionary name	D/L size	Real size	Strings	Ratio	Group	<u> </u>	
African.pcd	268 Kb	1 Mb	137821	16%	01 - Languages		
American.pcd	82 Kb	399 Kb	45392	20%	01 - Languages	<u>.</u>	
🗸 Australian.pcd	53 Kb	926 Kb	79409	5%	01 - Languages	5	
Brizilian.pcd	89 Kb	1 Mb	129338	6%	01 - Languages	5	
Chinese.pcd	1 Kb	5 Kb	802	32%	01 - Languages	5	
Croatian.pcd	80 Kb	607 Kb	56558	13%	01 - Languages	s	
Czech.pcd	377 Kb	3 Mb	309365	11%	01 - Languages	5	
Danish.pcd	614 Kb	7 Mb	564295	8%	01 - Languages	5	
Jutch.pcd	393 Kb	2 Mb	245688	12%	01 - Languages	s	
Finglish.pcd	6 Mb	36 Mb	3540673	17%	01 - Languages	s_	
Final-Land	004145	C MAL	400570	4.407	01 1	·	
Please select a dictionan	to download			Total files: 25	9 total size : 1 00/	Mb	
Prease select a dictionary to download Total files: 259, total size: 1 894 Mb							
Update list Download Use local version							
OK Cancel							

GPU settings

Before you can use a GPU in the attack, you must first select it in the <u>respective item</u> of the main menu.

	1	8

GPU dictionary force option	s		×
	ute-force Online dictionaries	GPU settings	
GPU specific op	tions		
'Max thread blocks' sets the 256 GPU threads). While 'M within a single GPU thread. trying different combinations <u>Read more about GPU diction</u>	maximal number of GPU blocks to be n ax passwords per thread' sets the maxir Even though both parameters are comp to maximize GPU utilization. onary force attack	un simultaneously nal number of pass uted automatically	each block incorporates words to verify from during runtime, consider
Max thread blocks	256		
Max passwords per thread	1000		
		ок	Cancel

GPU configuration is pretty simple and consists of only two settings:

- The number of parallel graphics card's blocks, where passwords would be searched. Typically, each block consists of 256 threads. Thus, if you set the number of blocks to 256, the GPU will run 256*256=65536 threads. The total number of checked passwords for one call to GPU kernel will be 256*ThreadBlocks*PasswordsPerThread. In our case 256*256*1000 = 65 536 000 passwords. Setting the **ThreadBlocks** smaller than 256 on modern graphics cards, in the majority of cases, leads to performance degradation.
- 2. The number of passwords to be search from a single thread. The greater the value, the lower the overhead associated with launching threads, and the higher the search speed. However, setting too great a value may hang the computer or cause significant fluctuations in the current search speed, displayed on the attack status tab. This is caused by the fact that task completion time on the GPU exceeds the time required for refreshing the current state of the attack.

Depending on the options you have specified, a proper choice of GPU settings can dramatically, often by several times, increase the password search speed. We recommend playing around with GPU settings to achieve the maximum utilization of the GPU in this attack.

The **Max password per thread** is not used and is always set to 1 when recovering domain cached credentials type 2.

When running password recovery for domain cached credentials type 2, you may need to play around with the **Max thread blocks** parameter to get better performance.

Setting too big value(s) may cause your GPU to hang or raise an error, depending on your <u>GPU kernel</u> timeout settings.

2.8.2.19 GPU: Hybrid dictionary attack

GPU Hybrid dictionary attack is pretty much the same as the <u>Hybrid Dictionary attack</u>, except that it utilizes your GPU power instead of CPU. That makes it extremely fast. Approximately 10 times faster than a simple Hybrid attack. The value greatly depends on options and hardware used though. The hybrid attack allows user to set his own word modification rules and attempt to validate the modified output words.

Actions, performed on source words from the dictionary, are called rules. Multiple rules can be applied to each source word.

GPU hybrid dictionary attack settings are grouped in eight tabs:

- 1. **Dictionaries** for setting up source dictionaries.
- 2. Rules files with set of rules.
- 3. **Super-rules** ones to be applied over the top of regular rules
- 4. Dictionary generator, where you can create files of words obtained from the hybrid attack.
- 5. Online dictionaries for downloading new dictionaries to the application.
- 6. Attack syntax complete description of all rules with examples.
- 7. Rule tester, where you can test your rules.
- 8. GPU settings is used to tune your GPU parameters.

Wordlists to be used in the attack are set on the first tab. Traditionally, the application supports wordlists in ASCII, UTF8, UNICODE, PCD, RAR and ZIP format. The position of the files on the list can be altered. For example, you may want to move smaller dictionaries up the list or the other way. During the attack, they will be used one after another, according to their position on the list.

Hybrid dictionary attack options							
Dictionaries Rules Super-rules Dictionary generator Online dictionaries							
Hybrid attack dictionaries							
Hybrid attack is simila	ar to mutation dictiona	rv attack, except th	at user should set his own word mutation rules.				
Setup at least one wo	ordlist here in order to	continue.					
Read many about the							
		Q. i.e.	C. H				
Dictionary name	Dictionary size	Strings	Full path				
WPR.pcd	6 290 880	2 150 822	C:\Program Files (x86)\Passcape\W				
Danish.dic	7 653 774	564 295	C:\SYS\PWC01\01 - Languages\Dani				
Dutch.dic	3 120 165	245 688	C:\SYS\PWC01\01 - Languages\Dutc				
English.dic	38 302 216	3 540 673	C:\SYS\PWC01\01 - Languages\Engli				
French.dic	3 822 281	320 033	C:\SYS\PWC01\01 - Languages\Fren				
German.dic	21 367 379	1 786 211	C:\SYS\PWC01\01 - Languages\Ger				
Japanese.dic	1 050 978	115 650	C:\SYS\PWC01\01 - Languages\Jap				
Norwegian.dic	12 658 942	809 552	C:\SYS\PWC01\01 - Languages\Nor				
Add	Add folder Rem	ove	Move up Move down				
OK Cancel							

On the '**Rules**' tab, define at least one file with password mutation rules. The format of the rules file is quite trivial; it is a plain-text ASCII file with the '**[Rules]**' string. Anything above this string is considered as comments and ignored by the program. Whatever goes below this string is considered as rules. Each string can contain several rules, applicable to a source word. If a string contains multiple rules per word, those rules are parsed left to right. For example, if you apply the rule '@pc\$a\$b\$c' to the source word 'password', at the output you will get 'Asswordabc'. The maximum length of an output word may not exceed **256** characters.

Hybrid dictionary attack options					
Dictionaries Rules Super-rules Dictionary generator Online dictionaries Image: Constructionaries Hybrid attack rules You should set at least one file with word mutation rules. Rule file is a simple ASCII text file with [Rules] section in it. Eventhing below the string is considered as rules. See the syntax of the rules at the Syntax tab. You can					
save all rules into a s Read more about Hy	single file (rules will be so /brid dictionary attack	orted, duplicates and errors will be skipped).			
File name	Rules count	Full path			
yurcusin		c. programmics (xoo) passcape (which produces (yare			
Add Remove Move up Move down Import Hashcat/Insidepro rules Save					
		OK Cancel			

'Super-rule' is a rule (or several rules) to be applied over the top of all other regular ones, before or after them. For example, you can set 'a8' tail super-rule to create all possible case combinations after a common mutation has been done. So the '/asa4' rule from I33t.ini file will become '/asa4a8', '/csc(' will become '/csc(a8', etc. Yet another one example: setting the '>6<G' head rule allows you to skip all words of less than 6 or greater than 16 characters, before starting a common mutation. This is a helpful feature once you decide to add the same rule to all text lines of the selected *.ini files. There's no need to modify them all. Be careful though, the 'aN' super-rule may increase the total number of generated passwords drastically.

Hybrid dictionary attack options
Dictionaries Rules Super-rules Dictionary generator Online dictionaries
Super-rules to be automatically append or prepend to every line of common rules
'Super-rule' is a rule (or several rules) to be applied over the top of all other regular ones, before or after them. If you set a HEAD super-rule, it is prepended to every line from the given *.ini file. If you set a TAIL super-rule, it will be automatically appended to the every line of common rules.
Read more about Hybrid dictionary attack
HEAD super-rule
This super-rule is applied BEFORE every line of common rules
TAIL super-rule
This super-rule is applied AFTER every line of common rules >6 <g< td=""></g<>
OK Cancel

The 'Dictionary generator' tab is designed for generating dictionaries obtained from an attack. Those custom-made dictionaries could be used, for example, in other applications. To generate a dictionary, specify a source dictionary and a set of mutation rules for it. The size of a target file may exceed 2 GB assuming that you save it to NTFS disk. Be careful, the dictionary generation process may take considerable time and disk space!

Hybrid dictionary attack options
Dictionaries Rules Super-rules Dictionary generator Online dictionaries
Dictionary generator
Passwords which were generated by this attack can easily be saved to file. Therefore, you can create your own dictionary and use it in another program. Be careful, dictionary creation may take quite some time depending on the source wordlists given and creation rules set.
Read more about Hybrid dictionary attack
Dictionary generator
Dictionary size unknown Generate
OK Cancel

You can download additional wordlists for the attack using 'Online dictionaries' tab.

If you want to create your own set of rules, you can use the next two tabs as sources of help. While the '**Syntax**' tab gives mere descriptions of available rules, on the '**Rule tester**' tab you can actually check them by specifying a source word and a rule. Forward your rule sets to us; if we find them interesting/useful, we will include them in the program.

Program's interface

Hybrid dictionary attack options							
Dictionary generator Online dictionaries Hybrid syntax Rule tester							
Hybrid rules to	Hybrid rules tester						
You can test your own hyb	rid rules here. Just type in a sample input word and the rule to be tested.						
Read more about Hybrid (lictionary attack						
Rule tester							
Input word	password						
Hybrid rule	@pc\$a\$b\$c	Đ					
Output word	Asswordabc						
	OK Cancel						

Rules description for the hybrid dictionary attack

Several rules at a line are allowed to be set.

Rules (if any) are processed from the left to the right.

Maximal line length is limited to 256 characters.

Maximal output word length is limited to 256 characters.

White space is ignored as long as it is not used as a parameter.

A line started with # character considered as a comment

All text before the '[Rules]' line is considered as comment.

N and M always start at 0. For values greater than 9 use A..Z (A=10, B=11, etc.)

The following rules should be at the last position of a line: aN, ?iN[C], ?i[C], ?oN[C], ?o[C], ?iZ[C], ?oZ[C]

Don't change the names of the standard rule files. Some ones are used by the program.

?iN[C], ?i[C], ?oN[C], ?o[C] ?iZ[C], ?oZ[C] rules use the following predefined charsets (you can use custom character sets though):

	U /
digits	- 0123456789
loweralpha	 abcdefghijkImnopqrstuwxyz
upperalpha	- ABCDEFGHIJKLMNOPQRSTUVWXYZ
alpha	 abcdefghijkImnopqrstuwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
special	- !@#\$%^&*()- +=~`[]{} \:;"'<>,.?/ "

loweralphanumeric - abcdefghijklmnopqrstuwxyz0123456789

upperalphanumeric - ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789

alphanumeric - abcdefghijklmnopqrstuwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 printable

abcdefghijkImnopqrstuwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789!@#\$%^&*()-_+=~`[]{}| \:;;"'<>,.? /

Itan	<u></u>			
Rul e	Exa mple	Input	Output	Description
:	:	passwoi d	password	Do nothing to the input word
{	{	passwoi d	asswordp	Rotate the word left
}	}	passwoi d	dpasswor	Rotate the word right
[[passwoi d	assword	Delete the first character
]]	passwoi d	passwor	Delete the last character
C	с	passwoi d	Password	Capitalize
С	С	passwoi d	pASSWOR D	Anti-capitalize (lowercase the first character, uppercase the rest)
d	d	passwor d	passwordp assword	Duplicate word
f	f	passwoi d	passworddr owssap	Reflect word
k	k	passwor d	gfhjkm	Convert word using alternative (first after default) keyboard layout. The rule works in both directions. For example, if there's Russian keyboard layout installed previously in the system, the rule should convert word 'password' to Russian ' ', and Russian word ' ' to 'gfhjkm'. This is very helpful when looking for non-English passwords. If only one language is installed in the system, the rule does nothing.
к	K	passwoi d	passwodr	Swap last two characters
I	I	passwoi d	password	Convert all characters to lowercase
q	q	passwoi d	ppaassssw woorrdd	Duplicate all symbols
r	r	passwoi d	drowssap	Reverse word
t	t	PassW ord	pASSwOR D	Toggle case of all characters
u	u	passwor d	PASSWOR D	Convert all characters to uppercase
U	U	my own passwoi d	My Own Password	Capitalize all words delimited with space (upper-case the first character and every character after a space)
V	V	passwor d	PaSSWoR D	Vowels elite

nuica

Program's interface

Rul e	Exa mple	Input	Output	Description
v	v	passwor d	pASSWoR D	Vowels noelite
'N	'4	passwor d	pass	Truncate the word to N character(s) length
+N	+1	u passwor d	pbssword	Increment character at position N by 1 ASCII value
-N	-0	passwor d	oassword	Decrement character at position N by 1
.N	.4	passwor d	passoord	Replace character at position N with character at position N+1
,N	,1	passwor d	ppssword	Replace character at position N with character at position N-1. Where N > 0.
<n< th=""><th></th><th></th><th></th><th>Reject (skip) the word if it is greater than N characters long</th></n<>				Reject (skip) the word if it is greater than N characters long
>N				Reject (skip) the word if it is less than N characters long
aN				Check all possible symbol cases for the word. N is a maximal length of the word to apply this rule for.
DN	D2D2	passwor d	paword	Delete the character at position N
рN	p3	key	keykeykey	Copy word N times
ΤN	T1T5	passwor d	pAsswOrd	Toggle case of the character at position N
уN	уЗ	passwor d	paspaswor d	Duplicate first N characters
YN	Y3	passwor d	paswordord	Duplicate last N characters
zN	z3	passwor d	ppppasswo rd	Duplicate the first character of the word N times
ZN	Z3	passwor d	passwordd dd	Duplicate the last character of the word N times
\$X	\$0\$0\$ 7	passwor d	password0 07	Add character X to the end of the word
^Х	^3^2^ 1	passwor d	123passwo rd	Insert character X at the beginning of the word
@X	@s	passwor d	paword	Remove all characters X from the word
!X				Reject (skip) the word if it contains at least one character X
/X				Reject (skip) the word if it does not contain character X
(X				Reject (skip) the word if the first character is not X
)X				Reject (skip) the word if the last character is not X
еX	e@	mike@y ahoo.co m	mike	Extract a substring starting at position 0 and ending up before first occurrence of X character (do nothing if X is not found)
EX	E@e.	mike@y ahoo.co m	yahoo	Extract a substring starting right after first found X character and till the end of the string (do nothing if X is not found)
% MX				Reject (skip) the word if it does not contain at least M instances of the character X

Program's interface

Rul e	Exa mple	Input	Output	Description
*XY	*15	passwor d	possward	Swap characters at positions X and Y
=N X				Reject (skip) the word if the character at position N is not equal to the X
iNX	i4ai5b i6c	passwor d	passabcwo rd	Insert the character X in position N
oN X	o4*o5 *	passwor d	pass**rd	Overwrite a character in position N with the character X
sXY	ss\$so 0	passwor d	pa\$\$w0rd	Replace all characters X with Y
xN M	x4Z	passwor d	word	Extract a substring of up to M characters length, starting from position N.
INX -Y	r10/-/r	google. com	google.com /	Insert the character X at position N if previous character at position N is not Y.
INX +Y	rl0.+.r	passwor d.	password	Insert the character X at position N if previous character at position N is Y.
ON X-Y	О0- +р	passwor d	-assword	If the character at position N is not Y, overwrite it with X character.
ON X+ Y	00Р+ р	passwor d	Password	If the character at position N is Y, overwrite it with X character.
RN M+ Y	R01+ a	passwor d	assword	Remove character at position N if character at position M is Y
RN M-Y	R40-b	passwor d	passord	Remove character at position N if character at position M is not Y
? iN[C]	? i0[digi ts]	passwor d	0password, 1password 9password	Insert a character from a charset [C] into position N of the word. Where C should be either a predefined charset name or a custom character set itself.
? iZ[C]	? iZ[digi ts]	passwor d	password0, password1 password9	Insert a character from a charset [C] into last position of the word. Where C should be either a predefined charset name or a custom character set itself.
? i[C]	? i[spec ial]	passwor d	~password, !password password_, password+	Insert a character from a charset [C] into every position of the word. Where C should be either a predefined charset name or a custom character set itself.
? oN[C]	? o1[up peralp ha]	passwor d	pAssword, pBssword pZssword	Overwrite a character at position N with a character taken from a charset [C]. Where C should be either a predefined charset name or a custom character set itself.
? oZ[C]	? oZ[up peralp ha]	passwor d	passworA, passworB passworZ	Overwrite a character at last position with a character taken from a charset [C]. Where C should be either a predefined charset name or a custom character set itself.

Rul	Exa	Input	Output	Description
е	mple			
?	?o[-	passwor	-assword,	Overwrite a character at every position of the word with a character taken
o[C	=.]	d	=assword	from a charset [C]. Where C should be either a predefined charset name
1			passwor.	or a custom character set itself.

GPU settings

Before launching the attack, make sure you setup up the GPU settings properly.

GPU hybrid dictionary attac	k options	x
Online dictionarie	es Hybrid syntax Rule tester GPU settings	Þ
GPU specific o	ptions	
You can edit GPU specific simultaneously (in most cas sets the number of passwo	settings here. Thread blocks option sets the number of GPU blocks to be run es each block incorporates 256 GPU threads). While passwords per thread rds to verify from within a single GPU thread.	
Read more about GPU hyb	rid attack	
Thread blocks	256	
Passwords per thread	1000	
	OK Cancel	

GPU configuration is pretty simple and consists of two parameters:

- The number of GPU blocks to be run at a single call to GPU. Each block consists of 256 threads. Thus, if you set the number of blocks to 256, the GPU will run 256*256=65536 threads. The total number of checked passwords for one call to GPU kernel will be 256*ThreadBlocks*PasswordsPerThread. In our case 256*256*1000 = 65 536 000 passwords per one call to GPU.
- 2. The number of passwords to be search in a single GPU thread. The greater the value, the lower the overhead associated with launching threads, and the higher the search speed. However, setting too great a value may hang the computer, make your GPU unresponsive or cause significant fluctuations in the current search speed, displayed on the attack status tab. This is

caused by the fact that task completion time on the GPU exceeds the time required for refreshing the current state of the attack.

Be careful setting 'heavy' rules like aN, ?iN, ?oN, etc. These rules may increase the number of generated passwords by hundred times and hang up your system or make your GPU device unresponsive.

The **Passwords per thread** is not used and is always set to 1 when recovering domain cached credentials type 2.

When running password recovery for domain cached credentials type 2, you may need to play around with the **Thread blocks** parameter to get better performance.

Setting too big value(s) may cause your GPU to hang or raise an error, depending on your <u>GPU kernel</u> timeout settings.

2.9 View menu

The View menu enables/disables the auxiliary elements of the interface, change the interface language, minimize the application to the tray or run it in the invisible mode.

2.10 Themes menu

You can select here one of the themes you've liked or create your own theme.

2.11 Help menu

In this section of the menu, you can access the help articles on using the software, visit the program's home on the Web, check availability of updates, submit a bug report, register your copy of Windows Password Recovery, etc.

2.12 Hardware Monitor



On this tab, you can view current CPU load, RAM utilization, GPU temperature and load. By default, the refresh interval is set to 2 seconds. Be careful: gathering these statistics also takes CPU time; therefore, when running "heavy" attacks, such as brute-force, it is recommended to keep the system monitor disabled.

3.1 Attacking Windows hashes

Currently the program can decrypt Windows hashes in several ways:

<u>Preliminary attack</u> (developed by Passcape Software) is based upon a social engineering method and consists of several sub attacks. Preliminary attack is very fast and often it is used for guessing simple and short passwords when there's no need to launch a fully scalable attack.

<u>Artificial Intelligence attack</u> - is a brand-new type of attack developed in our company. It is based upon a social engineering method and allows, without resort to time-consuming and costly computations, to almost instantly and painless recover certain passwords.

Dictionary attack. It is the most efficient recovery method, when the program tries each word from the dictionary (or dictionaries if there are several dictionaries) you specify until it finds the original password or until the wordlist is out of words. This method is very efficient since many people use regular words or phrases for password. Besides this type of recovery is performed quite fast compared to brute-force attack, for instance. Additional dictionaries and word-lists can be <u>downloaded from our site</u> or can be <u>ordered on CDs</u>.

Brute-force attack tries all possible combinations from the specified range of characters. For example, for a three-character range of lower-case Latin characters, it will check all possible combinations, starting with 'aaa', 'aab', 'aac', and all the way through 'zzz'. This is the slowest attack, so it is really great for short passwords.

<u>Mask attack</u> is a variation of the brute-force attack, except that some characters for finding the password remain unchanged, and only a portion of the password may change. The special syntax is used for setting a mask or rule for finding a password.

Base-word attack (developed by Passcape). At the first glance, this type of attack reminds the one we just described. It is just as efficient if a portion of the password to be recovered is known to us. However, unlike in the previous attack, here you do not have to set a mask - just provide a basic word. The program will take care of the rest. The phrase attack is based upon the experience of the social engineering to generate a great number of possible combinations of the given password.

<u>Combined dictionary attack</u> (developed by Passcape) uses to find compound passwords. For example, 'nothingtodo' or 'I give up'. It is very similar to the dictionary attack, except that instead of using a single word for password verification it uses a combination of words created by combining words from several dictionaries. You can create your own password generation rules.

<u>Phrase attack</u> (developed by Passcape) is very efficient against complex passwords. The idea of it is to guess the right password by searching through frequently used phrases and combinations. You can download pass-phrase wordlists and dictionaries from our site only.

<u>Rainbow attack</u> (developed by Philippe Oechslin). It is a time-memory tradeoff used in recovering the plaintext password from hashes. This attack is quite fast and effective tool for auditing Windows hashes.

<u>Fingerprint Attack.</u> Developed by Passcape, original idea by Atom. The attack parses input wordlist to generate so-called "fingerprints" used to recover the password. The attack is quite effective in finding difficult passwords for big list of hashes or for password history hashes.

<u>Hybrid dictionary attack</u> is like a simple dictionary attack, but allows user to customize word mutation and set his own password mutation rules. The rule definition syntax is compatible with some other password recovery software.

Online recovery (developed by Passcape Software) searches passwords in Internet databases. It deals fairly well with simple and frequently-used passwords. Its drawback is pretty low operating speed and poor suitability for handling large hash lists.

<u>Passcape rainbow table attack</u> (developed by Passcape Software). It's the next generation of regular pre-calculated tables. Passcape table attack is most suitable for the recovery of complex passwords of literally unlimited length.

<u>Batch attack</u> (developed in Passcape Software) creates a list/batch of attacks to be run one-by-one, so that you could launch all those attacks with a single mouse-click instead of configuring each of them individually.

<u>GPU brute-force attack</u> is fully identical to simple brute-force except that to guess passwords, it uses video card instead of CPU. The GPU device to be run the attack on, should be set in *General Options*.

<u>GPU fingerprint attack</u> works exactly the same way the simple fingerprint attack does but uses GPU power.

<u>GPU mask attack</u>. This password recovery method is fully identical to the regular mask attack except that the password guessing is processed by a graphical card of your PC, thus the recovery speed is much higher.

<u>GPU dictionary-force</u>. Often, when creating passwords, users add certain symbols in the beginning, end or even middle of the word. To recover passwords of this specific kind, we have come up with a GPU-based dictionary attack.

<u>GPU Hybrid dictionary attack</u>. The same as a simple Hybrid dictionary attack but much faster because uses GPU.

3.2 Attack comparison table

Which attack is the best? How do you choose the attack? The answers to these questions should be found in the attack comparison table.

Attack	Description	Time required	Guara nteed	Pros	Contras	Limitations
Preliminary	A set of light and speedy mini- attacks for finding simple, short or common combinations	A couple of minutes	No	Great quick-find tool for quick recovery of common, simple, short passwords, keyboard combinations, repetitive sequences, etc. Good for finding weak passwords quickly:	Practically useless fo serious analysis, when recovering the majority of complex passwords	rFinds mainly simple passwords

doesn't require additional settings

Artificial Intelligence	The most advanced way of recovering passwords, based on the methods of social engineering.	Min: 2-3 minutes, Max: over an hour	No	The best tool for finding complex passwords, which other methods cannot cope with. Works great for passwords, words and combinations that the user stored in the system any time in the past.	During the most efficient analysis, when all the options are set to the maximum performance, the attack takes considerable time. Finds not all passwords.	Efficient only when run on the original system (where the passwords were taken)
Brute-force	Searches all possible combinations within a specified character set	Depends on options	Yes	The only attack (along with the mask attack) that is guaranteed to recover a completely unknown password. Good for any short and medium passwords	Searching long passwords takes considerable time. Hard to guess the right range of characters to be searched.	May take centuries to search long passwords. Does not find passwords when uses wrong character set or password length exceeds the one specified
Dictionary	Finds password by searching words from predefined dictionaries (word-lists)	Almost instantly	No	Good and speedy tool for recovering common passwords	Requires having good dictionaries, does not take into account peculiarities of the language and letter case	Finds only common passwords
Dictionary with smart mutation	Same as dictionary attack, except here each word from the dictionary undergoes all kinds of mutations. For instance, appending numbers, changing letter case, deforming (displacing) letters, etc.	Up to 1000000 times slower than a simple dictionary attack	No	Good for all sorts of variations of common passwords	The maximum (most effective) mutation takes considerable time	Fails to find strong (non- dictionary) passwords, mutation takes considerable time

Mask	Finds passwords by specified mask (password generation rule)	Depends on options	Yes	Guaranteed to recover the remaining portion of a password. Good option when some portion of the original password is known.	Requires having the exact known portion of the password and its length and specifying the right character set to be searched	Password will not be found if a wrong character set, incorrect password length or incorrect known portion of the source password is specified
Combined dictionary	Checks complex passwords (composed of two or more words) by gluing words from several dictionaries	Depends on options	No	The only attack that finds long and complex passwords	Limited set of field- specific dictionaries, does not take into account peculiarities of non-English passwords (endings, suffixes, etc.) With a large source dictionary, the attack may take considerable time	Requires to know in advance that the password being searched for consists of two or more words; relatively slow
Combined dictionary with smart mutation	Same as combined attack, plus mutations	Depends on options	No	Same as the previous attack	Same as the previous attack. Requires setting additional mutation rules for the passwords to be generated	Same as the previous attack; mutations require considerable time
Base-word	Takes advantage of a known base word used for making up the password	A couple of seconds if the base- word length is not exceeds 16 characters	No	Good for the cases when you had known the original password but have forgotten its variations, e.g., letter case or trailing numbers	Mutation for long passwords (over 16 characters) may take some time	Does not always work
Phrase	Same as dictionary attack, except that instead of a word this one checks a phrase, popular	From several minutes up to several hours	No	The only attack against password phrases.	Only a small percentage of users use pass-phrases as passwords. Phrase mutation is imperfect; the mutation and	Does not take into account peculiarities of the language; limited choice of mutations.

expression, excerpts from songs, books,

etc.

Uses

tables

Fingerprint Based on

wordlist

precalculated

fingerprints that several were generated hours up out of the given to several

Rainbow

tables

ograr	n				200
			analysis take	Difficulty in the	
			considerable time. Insufficient number of relevant dictionaries; in particular, with non- English phrases and expressions.	creation of specialized dictionaries.	
Usually several minutes (or even seconds) for each password	Up to 100% if the passw ord fits into the characc er set and passw ord length of the table(s)	Currently one of the best attacks for recovering the majority of passwords by the time/efficiency ratio	Requires tables. Precalculation tables may take much room on a hard drive. It is impossible to recover long passwords using this attack.	Cannot recover all passwords simultaneousl y, generating a new table takes longer than running a brute-force attack. Limited recovery capabilities for long and non- English passwords	
From several hours up to several days (depends on the initial dictionary)	No	Finds complex passwords that were impossible to recover in other attacks	Big input dictionary may generate too much fingerprints. The success depends on the input dictionary.	The attack take too much time to complete when setting a big input wordlist.	
Depend on the	No	Good for all sorts of variations of common	Cannot recover complex passwords.	Fails to find strong (non-	

Hybrid dictionary	It is much similar to simple dictionary attack, except that the password mutation rules are fully customizable and should be set by user.	Depend e on the source wordlist and rules counter. Usually up to several minutes for a small wordlist.	No	Good for all sorts of variations of common passwords	Cannot recover complex passwords	Fails to find strong (non- dictionary) passwords

Online recovery	Searches passwords via Internet	Depends on options set and internet connectio n speed. Usually less than 1 minute for a single hash.	No	Pretty nice alternative tool for finding out simple and frequently- used passwords.	Very slow, processes hashes subsequently, feeds a lot of Internet traffic.	Fails to find most strong passwords. Works only when internet is available.
Passcape rainbow tables	Uses specially formed precalculated tables to guess strong and complicated passwords	Several minutes (or even seconds) for each password, dependin g on table parameter s.	No	Actually it is very good and advanced attack for recovering strong and complicated pass words which cannot be cracked in other attacks	A good table precalculation may take much disk space and time. Password recovery success rate greatly depends on input wordlist.	Cannot recover all epass words simultaneousl ey; generating a new table takes longer than running a brute-force attack. Not all initial wordlists suit well for creating Passcape tables.

3.3 Recovering passwords from hashes

Use this simple instruction for the recovery of any passwords in Passcape programs. This instruction is offered in the format of recommendation and is meant primarily for the recovery of passwords encrypted with OWF; e.g., from Windows hashes.

When recovering certain types of passwords the major question is: How to organize the recovery process - which attack should I start with to raise the probability of its successful completion?

For choosing the type and the sequence of the attacks, we advise to follow this algorithm, which is applicable in the majority of cases to all types of passwords to be recovered:

First, enable the preliminary attack option, if it is available. It will help to recover simple and frequently used combinations.

Second, select one or several passwords you need to decrypt first of all and run Online recovery to find out simple and frequently-used passwords.

Third, if you are aware of any specifics of the password you are looking for, it's better to try mask attack or base-word attack first. Specifically, if you know a part of the password - using mask attack would be more effective. If you know the basic component of the password or, for example, know the password but don't remember the sequence of caps and lowercase characters in it, base-word attack would do the job better.

Fourth, if you there's no information on the password you are looking for, which occurs most frequently, be guided by the following sequence of steps:

- 1. Launch Artificial Intelligence attack with mutation and indexing options set to light.
- 2. If the password was not found, try once again with mutation option set to normal level and indexing set to deep.
- 3. Run a rainbow table attack if there are any tables
- 4. Run a Passcape rainbow table attack.
- 5. Run dictionary attack with the mutation option disabled.
- 6. Launch dictionary attack with the mutation option enabled; the depth of mutation depends on the amount of available time and the attack speed. When searching for passwords typed in the national keyboard layout, the depth of mutation should be set to strong.
- 7. Select and download online dictionaries and repeat steps 5 6.
- 8. Run Hybrid dictionary attack.
- 9. Repeat Hybrid attack using alternative wordlists.
- 10. Launch pass-phrase attack with the mutation option disabled.
- 11. Launch pass-phrase attack with the mutation option enabled and set to the maximum productivity. This will allow finding even passwords typed in the national keyboard layout.
- 12. Select and download online pass-phrase dictionaries and repeat steps 10 11.
- 13. Launch combined dictionary attack with defined phrase generation rules.
- 14. Select and download online dictionaries for combined attack and repeat step 13.
- 15. Run fingerprint attack with default dictionary.
- 16. Select and download new online dictionary for the fingerprint attack, adjust options, set the new dictionary and repeat step 15.
- 17. Select a charset and password length for brute-force attack, launch the attack.
- 18. If necessary, select a new or complete the old character set and repeat the brute-force attack; i.e. step 17.

Based on the given recommendations, it is easy to create your own rules for batch attack.

3.4 Windows passwords FAQ

Q. What is password protection?

A. Perhaps no one would argue that Windows NT-based operating systems today are the most popular all over the world. That makes them very vulnerable targets for various kinds of hackers, intruders and dishonest users. The spread of the global network only exacerbates the situation. To ensure the personalization of stored user or system data and to protect it from unauthorized access by third parties, it was proposed to use the password protection technology. Currently, the primary protection in Windows operating systems is password protection. Access to private data in this case is possible only when user knows the original password, which is normally a word or phrase. Here is what it looks like in the real life: the program or system, on an attempt to access private data, prompts user for the text passwords. That password is checked against the original password, and, if the values match, the system allows access to the private data; otherwise, it denies access. The primary

disadvantage of password protection is that the program or system must store the original password somewhere, in order to have something to compare the entered value with.

Q. How do operating systems store passwords?

A. But everything is not so bad; Windows NT was developed in a way that it wouldn't store the original text value of the password. "How is that?" You may ask. - Very easy. There are special cryptographic password wrapper algorithms that work one way only. That's why sometimes they are referred to OWF - one-way functions. Roughly, you can get the hash from a password, but there's no way to get the password from a hash. How does it work in Windows? When creating an account, user enters the original password, which, however, is not stored as plain text; instead, it is hashed with an OWF function. The password hash returned by the function will be stored in the system. Further on, when attempting to log on, the system will prompt user for the password; it hashes the password again and then compares the generated hash with the original one that is stored in the system. If the two values match, the passwords, naturally, match too. Thus, the original text password is not stored in the system. Moreover, there are new algorithms out there that do not even store hash, and the number of such algorithms keeps growing. An algorithm of such kind, for example, is used for encrypting passwords in Internet Explorer 7-8. You can learn more about it in our article.

Q. How do passwords become encrypted?

A. For hashing user passwords, Windows NT uses two algorithms: LM, which we have inherited from Lan Manager networks, which is based on a simple DES conversion, and NT, based upon the MD4 hashing function. LM, as the weaker and vulnerable one, is not supported by default by the latest Windows Vista and Windows 7; however, you can still enable it. Moreover, there is a tendency to completely eliminate or replace it. It is important to know that when the LM hashing option is on (it is enabled by default in Windows XP), all user passwords are considered quite vulnerable. Cracking the majority of such passwords normally takes just a few minutes. The NT hash is free from the disadvantages, common to the LM hash. Consequently, it is much harder to pick the right password to a known NT hash than to an LM hash. But the current trend of increasing the computing power of modern computers, especially when using GPU, possibly, will make this standard too vulnerable to potential attackers.

Q. Where are password hashes stored?

- A. So, we have found out that user passwords in Windows systems are converted to special values hashes. LM and NT hashes both have a fixed size 16 bytes and can be stored in two repositories: SAM for the regular accounts and Active Directory for domain accounts.
- **SAM:** The regular accounts that contain user name, password and other auxiliary information are stored in the Windows NT registry; precisely, in the SAM (Security Account Manager) file. That file is located on the hard disk, in the folder %windows%\system32\config. The %windows% stands for the path to your Windows folder. For example, :\Windows\System32\Config\SAM. The system has priority access to the SAM file, so access to the file is denied to anyone, even administrators, while the system is loaded; nevertheless, Windows Password Recovery bypasses that restriction with ease. Besides that, of great interest for a potential attacker would be the backup of the SAM.SAV file and the compressed archived copy of SAM in the folder %windows%\Repair. Another way to access the SAM file is to launch a special program from a boot disk and then copy the file. Anyway you need a physical access to the computer with password hashes. User passwords or, to be accurate, hashes are additionally encrypted with the SYSKEY utility, which stores its service data in the SYSTEM registry file. Thus, to extract hashes from SAM, you would also need the SYSTEM file, which is located in the same folder as SAM.
- Active Directory: Domain accounts are stored in the Active Directory database. Usually, the Active Directory database is located in the file %Windows%\ntds\NTDS.DIT; it is the core of Active Directory. The way user hashes are encrypted here is a bit different than that is in SAM, but the recovery would also require the SYSTEM file. Access to the database is also under the system's complete control; however, unlike SAM, the ntds.dit database is resistant to modifications from the outside.

- Q. If everything is so easy, why not simply deny access to SAM or Active Directory to all users?
- A. That's the way it's done. By default, only the system has access to those files. However, these restrictions can be easily overridden. For example, WPR can import hashes from the current (locked by the system) files SAM and AD. Besides that, the system stores hashes in the computer memory to speed up access to them, so dumping the computer's memory is also an option.
- Q. I didn't quite understand it; what do I need to copy from the computer to recover the passwords?
- **A.** If that's a regular computer, copy these files: SAM, SYSTEM (the SECURITY and SOFTWARE files are also desired). If that's a server, you will need the same files plus ntds.dit one.

Q. How long does it take to pick the password if the LM hash is available?

A. The greatest disadvantage of the LM algorithm is that it splits the password into halves of 7 characters long. If user enters a password that is shorter than 14 characters, the program trails it with zeros to get a 14-character long string. If user password exceeds 14 characters, the LM hash appears the same as for an empty password. Each of the 7-character halves is encrypted independently; that considerably eases and speeds up the password recovery process. Another major disadvantage of the LM hash relates to the fact that during the encryption all the alphabetic characters of the password are converted to uppercase. In other words, the hashes for PASSWORD, password, Password or pAsswOrd will be completely identical. By running a brute force attack against each half, modern personal computers can pick an alphanumeric LM hash within a few minutes (or even seconds, when using the Rainbow attack). Let's do a bit of calculation. To pick a password for any alphanumeric combination, we need to split the password into two 7-character long parts and then search 36+32²+..36⁷=80 603 140 212 combinations. Besides, all the hashes will be searched simultaneously. The search speed in Windows Password Recovery on a computer Intel Core i7 is over 100 million passwords per second. Let's round it downward to 100. 80 603 140 212 / 100 000 000 = 806 seconds. That means, we are guaranteed to get the right password within just a bit over 10 minutes using the brute force.

Q. Can I see the encryption sources?

A. Sure. Let's review a working password encryption program for the LM algorithm.

Q. How much time is it required to guess the password if its NT hash is known?

A. With NT hashes it's a bit more complicated. The NT hash does not have the disadvantages that are common to LM. Therefore, the probability of the recovery of the password completely depends on its length and complexity, and drops like a snowball. Even despite the fact that the NT conversion algorithm is faster. Let's take a look at the following table that demonstrates the how search time depends on password length and complexity. Assuming that the brute-force recovery speed is 10 Bln. p/s (1 top GPU in 2014).

Character set	Password length	Password sample	Time to crack
A Z	5	CRUEL	instantly
A Z	6	SECRET	instantly
A Z	7	MONSTER	instantly
A Z	8	COOLGIRL	22s
A Z	9	LETMEKNOW	~ 10m
AZ, 09	5	COOL3	instantly
AZ, 09	6	BANG13	instantly
AZ, 09	7	POKER00	8s
AZ, 09	8	LETMEBE4	~ 5m
AZ, 09	9	COOLGIRL1	~ 3h

AZ, az, 09	5	P0k3r	instantly
AZ, az, 09	6	S3cr31	10s
A Z, a z, 0 9	7	Didlt13	~ 6m
A Z, a z, 0 9	8	GoAway99	~ 6h
A Z, a z, 0 9	9	19Sample3	~ 16d

Q. How much time is it needed to guess NT password by it's LM hash?

A. Almost instantly.

Q. Why can't I just remove/drop the hash, i.e. set a blank password?

A. Who said you couldn't? You can. For instance, using this utility. This way is just fine for those who need to regain access to their (or someone else's - e.g., when talking about the respective authorities) account at any cost. Moreover, with the above mentioned utility, you can do the following: remember the hash, then reset the hash, log on to the account with an empty password, do necessary manipulations with it, and then restore the remembered hash back. But that's not as simple as it seems. Even if you have reset the password and gained access to the account, you still won't be able to recover the majority of other passwords. Why? - Because the user password participates in the creation of the user's master key, which is used in the DPAPI and EFS encryption and other Windows subsystems. In other words, even if you reset the password, you will not be able to recover any of the following data: EFS-encrypted files, Outlook account passwords, Internet Explorer 7-9 passwords, network connection passwords (RAS, DSL, VPN etc.), network passwords to other computers, wireless network keys, MSN Messenger credentials, Google Talk & Google Chrome passwords, Skype, etc.

Q. So, in order to recover, for example, an Internet Explorer password, I would need to get the account password first, right?

A. Exactly.

Q. Are there any backdoors?

A. Like anywhere else. For example, sometimes the account password can be stored in the plain-text form in the secrets. Passwords to many system accounts can also be recovered with ease.

Q. Is that what the SECURITY registry file is requested for when importing hashes from the local computer?

A. Yes. The Security's main purpose is to be a storage for the so-called LSA Secrets. These secrets (but not they alone) can store plain-text passwords. Artificial Intelligence attack implements a checkup for possible vulnerabilities in the system and, as the consequence, chances to recover some passwords.

Q. Can I tuck an existing hash instead of the password when logging on to the system?

. There are programs that do that. Here is how they work. Before booting up the system, they extract user password hashes from SAM. Then, when loading the account, they tuck the known hash instead of the password. However, the result of such manipulations is the same as of merely resetting the password; i.e. you won't be able to recover the majority of other passwords.

Q. What can I do if the SAM file is hopelessly corrupt? Is there a way to recover the original password in this case?

. Yes, there is. However, you will no longer have access to the system. You can, for example, pick the password using the user's master key. Passcape Software has means for doing that. If the computer belongs to a domain, the names and hashed passwords of the last ten users registered on the computer are cached in its local system registry, in the SECURITY\Policy\Secrets section. You can take advantage of <u>Reset Windows Password</u> for dumping those hashes (they are also referred to as MSCACHE) and then attack them using Network Password Recovery Wizard.

- Q. I need to regain access to my account. Would you draw a picture "for dummies" what's the best way to do that, and how do I do that?
- A. Briefly, there are two ways to regain access to an account.
- 1. Reset the password; e.g., make the password blank. There are special utilities for doing that; the most powerful one is Reset Windows Password. Its operation principle is simple. Run a boot disk creation program and create an Reset Windows Password boot CD/DVD or USB disk with it. Next, power on the computer with the account you need to regain access to and edit the BIOS settings to enable the computer to boot from CD/DVD /USB. Some computers have this option enabled by default. Now boot up from the Reset Windows Password boot disk and follow the wizard's instructions to reset the password to the account. However, resetting the password guarantees only access to the account. If you also need to regain access to EFS-encrypted files or recover other passwords (e.g., network ones), this method won't do for you.
- 2. Recover the original password. By the way, that can be done by that same Reset Windows Password, running the intellectual attack. However, its capabilities are limited by only weak and vulnerable passwords. For restoring the original password, it is recommended to use Windows Password Recovery. In this program, once the hashes are imported, select and launch one of the proposed attacks. If the attack did not succeed, you can alter the settings and run the attack over or replace it with another one. Read on to find out how to choose the best attack for your hashes.

Q. Where can I find word-lists for dictionary attacks?

A. It is not necessary to search it. You can <u>download dictionaries</u> from within the Windows Password Recovery. We have a huge set of dictionaries at our Web site.

Q. How do I make my password more secure?

- A. There are several ways how you can secure yourself from picking your passwords by potential attackers:
- Do not use dictionary words in any language, names, numbers, repetitive sequences of letters and numbers, abbreviations, keyboard combinations, personal information, etc. Such passwords can be guessed extremely fast and easy.
- Increase password length. However, there is a reasonable limit for everything. Remember that length is not the main thing (although not with passwords). Finally, making up a too long password will cause you to successfully forget it after a weekend party or vacation. Besides that, an average human's memory cannot hold more than 5-7 passwords at a time. Still, there are network password, Web password, etc. - that are to be remembered also.
- Extend the character set used in the password. For example, replace the ' ' characters in the password with the '@'. Using national characters also strengthens up passwords radically. Use uncommon characters; for instance, '~'. Do not use hard-to-remember passwords that consist of a random set of characters unless you are a genius.
- Do not use the same password for logging on to Windows, Web sites, services, etc.
- If you have trouble remembering all your passwords, save them in a separate password-protected file in a safe place. A good password protection is implemented, for example, in the Rar archiver. Do not keep that file on the local computer.
- Never enter your password on someone else's computer.
- It's not a good idea to write down your passwords on sticky notes and stick those on the monitor.
- Think about additional protection. For example, if you enable the SYSKEY startup password option, chances are close to 100% that not a single attacker will be able to break your passwords without having guessed the original SYSKEY password first.

3.5 Windows Password Recovery FAQ

Q. What do the question marks in LM passwords mean?

A. As you may have already known, an LM password consists of two halves. If an LM password has 7 leading question marks, that means that only the second half of the password is found. The trailing question marks indicate the first half of the password recovered.

Q. What's the difference between LM and NT passwords? I have found both passwords: MASTERGURU and MasterGuru. Which of them is the right one? Which one should I use? A.To log on to the system, you need to use the NT password.

Q. When brute forcing an LM password, the program complains and tells me that it truncates the password to 7 characters. Is that a bug?

A. No. As you know, an LM password is split into two 7-character halves. Therefore, the maximum length of brute forced LM passwords is 7 characters.

Q. I know my NT password, but the program fails to find it for some reason? Why?

A.The NT password is case sensitive. Perhaps, you have set an incorrect search range. Try checking the password manually in (Tools-Password Checker). Password Checker automatically checks all possible combinations of uppercase and lowercase characters.

Q. I have recovered the internal administrator password, but when attempting to log on with it, the system tells me that the password is incorrect. What's the matter?

A. Most likely, you have recovered the local administrator's password, while your computer belongs to a domain. Domain passwords are stored in Active Directory, including the domain Administrator's password. Try logging on to the system in the safe mode.

Q. During a dictionary attack, I have recovered a password that was not in the dictionary. How did that happen?

A. Most likely, you had set the maximum mutation level, when the program also checks dictionary words typed in a non-English, national character set, depending on the keyboard layout. For example, the word 'secret' typed with the Cyrillic layout will produce the word ' '. Besides swapping keyboard layouts, the active mutations can mutilate the words to the point where they are hard to recognize. Mutation is used in the preliminary, intellectual, dictionary, and combined attacks, as well as in the key word and phrase attacks.

Q. In a batch attack, can I set the same attack type but with different settings?

A. Yes, you can do that.

Q. I've got a question concerning online dictionaries. I've noticed that they are extremely compressed, to the level greater than those that are produced by the archivers. What is the PCD format?

A. That is a proprietary dictionary storage format developed in Passcape, which uses additional optimization and encryption algorithms. Some dictionaries can indeed be compressed harder than with a regular archiver. For example, the Australian.pcd dictionary in the original format takes 926 KB of space, while in the compressed format it's only 53 KB.

Q. I chose to run a dictionary attack and set the medium mutation level. When I launched the attack, I was unpleasantly surprised with the low speed, only a few thousand passwords per second. Why is it so slow?

A. The program shows the attack speed without mutations. For example, if 1000 words has been processed within a second, it shows 1000 p/s, although the mutation module could have generated 1000

additional words per each word during that time. Thus, the actual search speed is by hundreds or even thousand of times greater than what you see on the screen.

Q. Can I use the regular dictionaries in a combined dictionary attack?

A. Yes, you can.

Q. I know that the password begins with "blue". Which attack would be the best one to use?

A. You can try dictionary attack. For example, the mask blue%c%c%c%c%c%c would search the range from blueaaaaaa through bluezzzzz.

You can also try running a combined dictionary attack. In order to do that, open notepad, then type 'blue' and save the file as, for instance, 1.dic. Then open the combined attack options and set 1.dic as the primary dictionary and any other - as the secondary dictionary. This way the program would search for disyllable words like bluepig, blueberry, bluegirl, etc. If you add the third dictionary, the program will search through the combination of the three components. For example, bluecoolgirl, blueblackhash, bluebadboy.

Q. The Artificial Intelligence attack goes too slow. What's the matter?

A. It's either because the password cache is full. In this case, you need to try emptying it. Or because you have set too deep mutation, and the program has found quite many 'suspicious' words; i.e. the words that are considered as the potential passwords.

Q. I am launching the brute force, but the program complains that it has nothing to do. Why?

A. Before launching the brute force, you must first select the hashes. You can do that through the Edit - Select menu.

Q. What are Rainbow tables? And how can they be used for recovering passwords?

A. To launch a rainbow attack, in the attack options you need to load the *.RT or *.RTI files that contain Rainbow tables. The type of the tables must match the type of the hashes selected for the attack. Therefore, the names of the files with the tables must begin correspondingly: "Im_*.rt" for LM hashes, "ntlm_*.rt" for NT hashes. You can get some additional information and download rainbow tables at https://project-rainbowcrack.com.

3.6 GPU FAQ

Q: What are the system requirements for the program?

A: Currently the program supports NVidia video cards with CUDA compute capability 3.0 or higher, AMD Radeon at least 7xxx series and Intel HD Graphics 4xxx series or higher. The full list of CUDA supported devices can be found at https://developer.nvidia.com/cuda-gpus. Compatible AMD Radeon cards are shown here: https://en.wikipedia.org/wiki/Comparison of AMD graphics processing units. You should also have the latest video drivers installed.

Q: What versions of Windows the program supports?

A: GPU acceleration is supported starting up with Windows XP (NVidia GPUs) and Windows Vista (AMD GPUs) on both 32-bit and 64-bit systems.

Q: How do I know which architecture does my video card support?

A: For NVidia devices:

Launch the program, open the menu '*Options* - *General Options*,' select the '*GPU Settings*' tab, select '*NVidia CUDA*' platform and choose your video card here. The 'Compute capability' field in the description section should display your GPU architecture.

For AMD devices:

Launch the program, open the menu 'Options - General Options,' select the 'GPU Settings' tab, select 'AMD OpenCL' platform and choose your video card here. The 'CL_DEVICE_VERSION' and 'CL_DEVICE_OPENCL_C_VERSION' fields should display your GPU architecture supported.

Q: Where can I get the latest video drivers?

A: You can download the latest drivers from NVidia (<u>https://www.nvidia.ru/drivers</u>) and AMD (<u>https://support.amd.com/us/gpudownload/Pages/index.aspx</u>) web sites.

Q: Where can I read more info about CUDA?

A: <u>Wikipedia site</u> is a good starting point to start from.

Q: Where can I read more info about AMD Radeon cards?

A: https://en.wikipedia.org/wiki/Comparison of AMD graphics processing units

Q: Where can I read more info about Intel Graphics?

A: <u>https://en.wikipedia.org/wiki/Intel HD and Iris Graphics</u>

Q: After I launch a GPU-based attack, my computer freezes or crashes into BSOD. What's the problem?

A: The problem may be caused by the following reasons:

- Your video card had been overclocked, and it was malfunctioning at high load. If that's the case, bring the frequencies of the video memory/cores to its defaults.
- Insufficient or ineffective cooling of your card. When you launch a GPU-based attack, the program
 utilizes the most of the GPU power, and the GPU temperature rises to a critical level. Make sure that
 your video card is well cooled, the GPU slot and your system unit are free from dirt and dust. An
 unwise use of some video settings may have a negative impact on the video card's temperature and its
 stability under high load conditions. For example, some applications reduce the fan speed to minimize
 the noise, which does result in noise reduction, but also increases the core temperature.
- Power supply problem. Your card can consumes a lot of energy at full load, and the power supply unit may be unable to handle such a high demand for power. If the video card has additional 6-pin or 8-pin power connectors, make sure they are all properly connected.

Q: When I launch a GPU attack, my computer slows down a great deal. How can I fix that?

A: By default, the application is set up for using video cards of medium performance. That's usually 256 threads per block, 256 blocks and 1000 passwords per thread. For older video cards such a configuration is too much and may cause a slowdown. Consider reducing the value of "*Passwords per thread*" to 100 or even less.

Q: What's the best way to find optimal values of "*Thread blocks*" and "*Passwords per thread*" in the GPU attack settings?

A: You can do that either empirically or by doing some maths. For example, if the values are 100 and 100, and the average speed of attack is 1 billion passwords per second, you can calculate that the GPU kernel is called about 390 times per second (the number of passwords calculated each time is usually 256 * ThreadBlocks * PasswordsPerThread). Naturally, the fewer calls, the less the overhead, and the higher the attack speed. On the other hand, you must call the GPU program at least a couple of times per second. So use a calculator, and adjust the parameters. You can also adjust them using a rule of thumb, that is, increasing their values until the speed of attack stops going up and the computer slows down. If you have a GPU monitor installed in your system, it should indicate a load of at least 98-99 percent. Besides, it's important to know some other things too. First, don't set the summary values of those parameters too high. Otherwise your system may malfunction or freeze. Second, you'd better not set the value of "*Passwords per thread*' at less than 100 as it will negatively affect the speed of attack regardless of what kind of video card is used.

Q: Does the PCI-Express bus have any impact on the performance?

A: Actually, this impact is negligible. It's usually masked by other factors. So the generation of your PCI-Express bus and its performance don't matter much.

Q: Does the amount of video memory matter?

A: No, it doesn't. However in most cases, your GPU should have at least 256 Mb of video memory.

Q: A GPU-based attack slows down my PC so I can barely use it. How can I fix it?

A: There are two ways to fix it: temporary and permanent. As a temporary fix to the problem, go to the attack settings and try reducing the number of GPU blocks used or the number of passwords checked per GPU thread. As a permanent fix, install a second video device, provided that you have a second slot on your motherboard and that your power supply unit can handle the additional load. For example, you can use some cheap card as the primary one (for displaying information on your monitor), and a second, more powerful one, for brute-forcing passwords.

Q: I have more than video cards in my computer. Can I use them all for brute-forcing?

A: Yes. You can use all or some of them. Just open general settings and specify the GPU device(s) to be used by the program.

Q: What's the maximal number of GPU devices does your program support?

A: It depends on your hardware. Even though the program supports up to 255 devices, typically, up to 8 devices can be installed into a 4 PCI-E slot motherboard (4 double-GPU cards).

Q: Can I brute-force passwords on devices which performance varies a lot?

A: Yes, you can.

Q: The program can not detect my video card. What can I do?

A: Update your video drivers. If it didn't help, try to extend your desktop to all devices (if you have more than one device). Re-plug your device into another PCI-Express slot.

Q: Your application can't use all of my GPUs.

A: You will have to disable SLI in order to be able to use all devices.

Q: Can I use both NVidia, AMD and Intel devices simultaneously?

A: Yes, you can use NVidia, AMD and Intel devices simultaneously.

Q: How can I check my GPU utilization?

A: Open 'Hardware Monitor' tab. In 'What to show' drop-box choose the device you need and select 'Show' to display it. You can then click 'Start' or 'Stop' buttons to manage the hardware monitoring. The GPU monitor shows device load (utilization), temperature and fan speed.

Q: My NVidia GPU is absent in hardware monitor.

A: You should install/reinstall NVAPI library. Download the library at https://developer.nvidia.com/nvapi

Q: My AMD GPU shows zeros in hardware monitor.

A: Install/reinstall the latest AMD drivers or ADL component. Make sure your AMD device is active (connected to active monitor). Non-active devices are not processed properly by ADL due to a bug in AMD drivers.

3.7 Online dictionaries

Dictionaries	Dic Filters Mu	tionary attac	k options line dictionari	es		Þ
Passcape on First the program attemp dictionaries available for it. You can use Ctrl or SH Bead about Passcape on	line dictionarie ots to establish a downloading, Se nift buttons for n	s connection with elect the dictiona nultiple selection	n the Passcape : ary you need ar	server and re	etrieves the list of load' button to retrieve	
Dictionary name	D/L size	Beal size	Strings	Batio	Group	1
	D/L 3/26	1 Mb	107001	1692		
	200 KD	1 MD	137021	10%	01 - Languages	
V American.pcu	OZ KD	399 KD	40392	20%	01 - Languages	
Australian.pcd	53 KD	926 KD	79409	5%	01 - Languages	
Brizilian.pcd	89 KD		129338	6%	01 - Languages	
Chinese.pcd	1 KD	5 KD	802	32%	01 - Languages	
Croatian.pcd	80 KD	607 KD	56558	13%	01 - Languages	
Czech.pcd	377 Kb	3 Mb	309365	11%	01 - Languages	
V Danish.pcd	614 KD	7 Mb	564295	8%	01 - Languages	
V Dutch.pcd	393 Kb	2 Mb	245688	12%	01 - Languages	
English.pcd	6 Mb	36 Mb	3540673	17%	01 - Languages	
Finnish.ncd	931 Kh	6 Mh	480570	14%	01 - Languages	1
Nesco coloct a disting an	u to download		Takal fil	an 254 kek	al cizer 1 200 Mb]
iease select a uictionar	y to download		TUCALITIK	55,204,100	arsize; 1 390 MD	
Update list	Download	Use loca	version			
			ОК		Cancel	

The online dictionary selection dialog is extremely simple. When it opens up, the program attempts to establish a connection with the Passcape server and then retrieves and displays the list of dictionaries available for downloading.

Select the dictionary you need and then click on the 'Download' button to retrieve it and use in the program.

Some of the dictionaries are large. For instance, the size of *'music_songs.pcd'* is more than 59 MB in the compressed format. Naturally, retrieving such a large amount of data may take some time, which depends upon file size, bandwidth of your Internet connection and net load.

All online (and some additional) dictionaries can be <u>ordered on CD</u>. The total size of all the dictionaries is over 7.5 GB. You can also share your own dictionary with us by e-mailing us the dictionary or the link where it can be downloaded.

The word-list are used in common dictionary attack, combined dictionary and pass-phrase attacks.

License and registration

4 License and registration

4.1 License agreement

SOFTWARE LICENSE AGREEMENT

IMPORTANT-READ CAREFULLY: This is the End User License Agreement (the "Agreement") is a legal agreement between you, the end-user, and Passcape Software, the manufacturer and the copyright owner, for the use of the "Windows Password Recovery" software product ("SOFTWARE").

All copyrights to SOFTWARE are exclusively owned by Passcape Software.

The SOFTWARE and any documentation included in the distribution package are protected by national copyright laws and international treaties. Any unauthorized use of the SOFTWARE shall result in immediate and automatic termination of this license and may result in criminal and/or civil prosecution.

You are granted a non-exclusive license to use the SOFTWARE as set forth herein.

You can use trial version of SOFTWARE as long as you want, but to access all functions you must purchase the fully functional version. Upon payment we provide the registration code to you.

Once registered, the user is granted a non-exclusive license to use the SOFTWARE on one computer at a time (for every single-user license purchased).

With the personal license, you can use the SOFTWARE as set forth in this Agreement for noncommercial purposes in non-business, non-commercial environment. To use the SOFTWARE in a corporate, government or business environment, you should purchase a business license. With the business license you can run the SOFTWARE on multiple computers of your organization - no matter where they are located.

The registered SOFTWARE may not be rented or leased, but may be permanently transferred together with the accompanying documentation, if the person receiving it agrees to terms of this license. If the software is an update, the transfer must include the update and all previous versions.

You may not create any copy of the SOFTWARE. You can make one (1) copy the SOFTWARE for backup and archival purposes, provided, however, that the original and each copy is kept in your possession or control, and that your use of the SOFTWARE does not exceed that which is allowed in this Agreement.

The SOFTWARE unregistered (trial) version may be freely distributed, provided that the distribution package is not modified. No person or company may charge a fee for the distribution of the SOFTWARE without written permission from the copyright holder.

You agree not modify, decompile, disassemble, otherwise reverse engineer the SOFTWARE, unless such activity is expressly permitted by applicable law.

Passcape Software does not warrant that the software is fit for any particular purpose. Passcape Software disclaims all other warranties with respect to the SOFTWARE, either express or implied. Some jurisdictions do not allow the exclusion of implied warranties or limitations on how long an implied warranty may last, do the above limitations or exclusions may not apply to you.

The program that is licensed to you is absolutely legal and you can use it provided that you are the legal owner of all files or data you are going to recover through the use of our SOFTWARE or have permission from the legitimate owner to perform these acts. Any illegal use of our SOFTWARE will be solely your responsibility. Accordingly, you affirm that you have the legal right to access all data, information and files that have been hidden.

You further attest that the recovered data, passwords and/or files will not be used for any illegal purpose. Be aware password recovery and the subsequencial data decryption of unauthorized or otherwise illegally obtained files may constitute theft or another wrongful action and may result in your civil and (or) criminal prosecution.

All rights not expressly granted here are reserved by Passcape Software.

4.2 Registration

The software is available in three editions: Light, Standard and Advanced. The detailed instruction for all kinds of orders is available online at the <u>program's order page</u>. Online orders are fulfilled in just a few minutes 24 hours a day 7 days a week. If you purchase our products online, you will receive an automatically generated e-mail message with registration details within several minutes (if the order passes the fraud check system). However some orders can be marked for manual checkout or as 'suspicious'. This may increase order time up to several hours.

Important: when completing the order form, please double-check that your e-mail address is correct. If it will not, we'll be unable to send you the registration code.



To complete the registration:

- Open the registration message and copy the registration code to the Windows clipboard.
- Run the program, select Help Enter Registration Code.

- Type in your registration name and paste the code here.
- Click Register button to confirm.

I	Registration X
Please, enter you	r registration code
Registration information —]
Your name:	Ivan
Registration code:	
Enter your registration code a If you experience any probler to the program's manual.	exactly as given to you in the registration e-mail. Ins during the registration process, please refer

4.3 Limitation of unregistered version

An unregistered version of **Windows Password Recovery** shows only first 3 characters of recovered passwords and has some functional limitations. Registered version of the program eliminates all restrictions. Please refer to <u>this page</u> to view restrictions of a certain edition of the program.

4.4 Editions of the program

Windows Password Recovery comes in three editions: Light, Standard and Advanced. The detailed list of features and compatibility chart is shown below.

FEATURE	Light	Standard	Advance d
Windows XP - 11 workstation support	+	+	+
Windows 2003 - 2022 server support	+	+	+
Windows 64-bit support	+	+	+
Non-US Windows support	+	+	+
International passwords support	+	+	+
Multithreaded recovery	+	+	+
Interface themes support	+	+	+
Load hashes from local computer	+	+	+
Load hashes from remote computer	-	+	+
Dump regular hashes	+	+	+
Dump password history hashes	+	+	+
FEATURE	Light	Standard	Advance d
---	-------	----------	--------------
Search for plaintext passwords	+	+	+
Load hashes from SAM	+	+	+
Load Active Directory hashes	+	+	+
Load domain cached credentials	+	+	+
Load Windows PIN	+	+	+
Load Microsoft or AzureAD cloud hashes	+	+	+
Password recovery for domain cached credentials	+	+	+
Password recovery for Windows PINs	+	+	+
Password recovery for Microsoft or AzureAD cloud accounts	+	+	+
Support for password-less accounts	+	+	+
Import hashes from other programs	+	+	+
Load hashes from system restore folders	+	+	+
Export hashes to PWDUMP file	+	+	+
Common attacks	+	+	+
Advanced attacks	+	+	+
Smart attacks	+	+	+
GPU-based attacks	+	+	+
Support for multiple GPU devices	-	+	+
Batch attack	-	-	+
View AI password cache	-	-	+
Smart password mutation	+	+	+
Online dictionaries	+	+	+
Support for SYSKEY	+	+	+
Support SYSKEY startup password decryption	+	+	+
Support SYSKEY floppy decryption	+	+	+
Custom wordlist generator tool in dictionary attack	-	-	+
Generate dictionaries by mask	-	-	+
Generate dictionaries by given base-word	-	-	+
Combined dictionaries generator	-	-	+
Pass-phrase dictionary generator	-	-	+
Fingerprint dictionaries generator	-	-	+
Create wordlists based on hybrid dictionary attack	-	-	+
Support for hybrid and indexed (*.rti) rainbow tables	+	+	+
Can restrict access to the program	+	+	+
Password strength measurement	+	+	+
Hash checker	+	+	+
Random hash generator	+	+	+
Multiple hashes generator	-	+	+
Rainbow table generation tool	+	+	+
Passcape rainbow table generation tool	+	+	+
Dictionary to hash generator	-	+	+
Backup system registry files	-	+	+
Backup Active Directory database	-	-	+

FEATURE	Light	Standard	Advance d
Asterisk password viewer tool	+	+	+
Offline password remover for regular user accounts	-	-	+
Offline password remover for domain cached accounts	-	-	+
Offline password remover for domain accounts	-	-	+
LSA secrets Dumper	+	+	+
Domain cached credentials explorer	-	+	+
SAM Explorer	-	+	+
Active Directory Explorer	-	-	+
Windows Vault Explorer	-	-	+
Wordlist tools: create a wordlist by indexing files	-	+	+
Wordlist tools: merge wordlists	+	+	+
Wordlist tools: wordlist statistics	+	+	+
Wordlist tools: sorting	+	+	+
Wordlist tools: conversion/compression	+	+	+
Wordlist tools: wordlist comparison	+	+	+
Wordlist tools: additional operations	+	+	+
Wordlist tools: indexing words/passwords of HDD sensitive areas	-	-	+
Wordlist tools: HTML links extractor	+	+	+
DPAPI: offline DPAPI blob recovery	*	*	+
DPAPI: DPAPI blob analysis	+	+	+
DPAPI: DPAPI blobs search	+	+	+
DPAPI: Master Key analysis	*	*	+
DPAPI: dump password history hashes	-	-	+
DPAPI: analyze password history	*	*	+
Windows Hello: recover user credentials	-	+	+
Windows Hello: decrypt biometric databases	-	+	+
Windows Hello: PIN bruteforcer	-	+	+
Windows Credentials Explorer	-	-	+
Hardware monitor	+	+	+
GPU health monitor	+	+	+
Password reports	-	+	+
Use AI when recovering password in Wizard mode	+	+	+
Run in hidden mode	+	+	+
Max. user accounts at a time	500	5000	unlimited
14-days money back guarantee	+	+	+
License	personal	personal	business
Price	\$65	\$345	\$895

* - uses some restrictions

Technical support

5

5.1 Reporting problems

If you have a problem, please contact us at <u>support@passcape.com</u>. Please inform us about the following:

- Full name and version of the program
- Windows version including service pack, OEM and language information, etc.
- Registration information if any
- Detailed description of the problem, whether it is a constant or spontaneous error
- If you're reporting a critical error, please attach Crash.log file that was saved during an unhandled exception session.

5.2 Suggesting features

If you have any questions, comments or suggestions about the program or would like more information, email us at info@passcape.com. Please don't forget to mention the program name and version. Also make sure you have the latest program version installed. Your feedback helps us to improve our products and work more effective.

5.3 Contacts

Please don't hesitate to send your questions regarding our products to e-mail <u>support@passcape.com</u>. You will get reply during one or two days. Note, that registered users have priority in technical support.

If you experience any problems during registration process, please send a letter to sales@passcape.com

We will be happy to assist you with the registration.

Please write in English!

You can find other password recovery utilities at https://www.passcape.com