PeaZip

Open Source, Portable File and Archive Manager

Document Revision: 2025 11 29

Author: Giorgio Tani

Translation: Giorgio Tani

This document refers to: PeaZip 10.8.0

Licensing: Present documentation is released under GNU GFDL License; PeaZip executable implementation is released under GNU LGPL License.

PeaZip official website: https://peazip.github.io , and https://peazip.sourceforge.io

For more information about the licenses:

GNU GFDL License, see https://www.gnu.org/licenses/fdl-1.3.txt

GNU LGPL License, see https://www.gnu.org/licenses/lgpl-3.0.txt

Content:

		Basic
• What is PeaZip	3	
• How to…	4	
• Context menu	7	
• Dark mode	9	
		Advanced
• File manager	11	
○ Set password	31	
○ Password manager	33	
○ Create keyfile	35	
○ Set advanced filters	36	
○ File tools	38	
• Extract archives	40	
• Create/convert archives	46	
		Expert
• Save and schedule tasks	55	
• Task progress window	57	
• Settings and help system	58	
		Additional information
• Supported formats	69	
• PeaUtils	71	
• Customisation and scripting, command line syntax examples	72	
• Translations	75	
• Notes	76	

What is PeaZip

PeaZip is a general purpose file and archive manager application for Linux, macOS, BSD, and Windows, aiming to provide a crossplatform graphical interface for many Open Source archiving and compression utilities in order to handle most of available archiving formats like 7Z, RAR (extraction), TAR, ZIP and many other ones, see Supported formats chapter for more information. The program features powerful and flexible inclusion/exclusion filters and search tools, provide optional two factor authentication through password and keyfile, and allows deeply fine tune the task's definitions, exposing through a single, consistent frontend GUI the options of underlying applications. The list of objects to be archived or extracted (Layout) can be saved for future use, to speed up backup and restore tasks. Also the resulting command for archive creation and extraction operations can be saved as script, to get the full control on task's definition, helping the user in bridging the gap between GUI and console applications to get the best of both worlds. A detailed log is available after each operation. PeaZip also collects a set of handy file management tools: robust file copy, split and join files, fast or secure file deletion, calculation of a wide set of checksums and hashes over selected files, byte-to-byte comparison of two files, web search etc. PeaZip can be used as file manager, or can be used from context and SendTo menu; file associations and menu entries (both for context and SendTo menu) can be changed running the setup program any time it is needed. If no system integration is

preferred, PeaZip Portable is available as standalone application, not needing installation and not modifying the host system; both packages are available on application's website. All the open source backend binaries included in base packages contain only open source software released under OSI-approved licenses.

Optional plugin based on closed source binaries are available as separate add-ons only, and can be browsed on PeaZip official website Add-Ons page.

How to…

This mini-tutorial introduces the most common operations that can be performed through PeaZip, following chapters contains a more detailed explanation.

Extract archive(s)

- **From the system**
    - Right-click on the archive(s) and in system's context menu click "Extract here", "Extract here (smart)" or "Extract here (in new folder)" to extract with no further interaction.
    - Otherwise use "Extract..." menu entry for having more options: output path, password, extract files to a new directory, chose to skip, rename or overwrite existing files etc.
- **Open an archive in PeaZip,** i.e. with double-click or drag an archive on PeaZip's window or icon
    - click on the quick extraction link "Extract here" on the right of the tool bar
    - drag and drop content from file/archive manager to folders in navigation bar on the left of the app
    - drag and drop files and folders to the desired destination on the system (if supported by the host OS)
    - click on "Extract" in toolbar or in context menu to extract only selected objects trough the confirmation dialog allowing to set all options (output path, password, naming policy, extract to a new directory etc)
    - click "Extract all to..." button (shortcut Ctrl+E or F12) to directly extract the entire archive to the specified folder, or click the quick extraction dropdown button on its right to extract into most common destinations skipping confirmation dialog, and to set most important options; shortcuts: Ctrl+Alt+E extracts archive in its current folder; Ctrl+Shift+E extracts to desktop; Ctrl+Alt+Shift+E extracts to documents; Ctrl+0 extracts to the previous path, Ctrl+1..8 extracts all to 1..8 bookmark's path (if defined)
    - **Open PeaZip,** select one or more archives and click "Extract" in toolbar or context menu, or use quick extraction destinations as explained at the previous point.

When a **folder** is sent to extraction procedure, PeaZip browses it recursively to find and extract all the archives it contains.

*Hint: to extract **multi-volume (spanned)** archives, **place all archives in the same folder** and use "Extract All" (from context menu or program's GUI) on the first volume, which is usually identifiable for the extension .001, or .r01, .z01 and similar.*

Extract selected content from archive

Open an archive in PeaZip, i.e. with double click, or drag an archive on PeaZip's window or icon

- Click "Extract" in toolbar: only selected items will be extracted
- Drag and drop content from file/archive manager to folders in navigation bar on the left of the app
- Drag files and folders to the desired destination on the system (if supported by the host OS
- right-click and in "Extract" group of the context menu click "Extract selected" (to extract only selected items) or "Extract displayed" (to extract content of current directory or of current search filter)

Browse or extract password protected archive

Click on the padlock icon to enter a password and optionally a key file. The icon is featured both in PeaZip's file browser's status bar and in the archive extraction interface; once the password is set the icon will change colour.

Create archive(s)

From the system, right click on files/folders and click on "Add to archive" in context menu or SendTo menu, it will open the archive creation confirmation dialog, more options are available in "Advanced" tab; click OK to create the archive. Alternatively, drag files/folders to PeaZip's window or shortcut; the same archive creation interface will be shown. **From PeaZip** select objects to be archived and click on "Add" button; the same archive creation interface will be shown.

Add content to existing archive (write-supported formats)

Open an archive in PeaZip (i.e. with doubleclick, or drag an archive on PeaZip's window or icon), then drag files and folders to be added in the archive (or click on "Add" button and use application's context menu to add objects to the archive). It will open the archive creation confirmation dialog, click OK to update the existing archive.

Edit files inside existing archive (write-supported formats)

Open the archive in PeaZip, double click to preview desired file(s), which will be extracted to a temporary work path. If a file is modified on the disk, the application will automatically ask to update the archive with the new version. If synchronization is declined, choosing “No”, temporary file(s) will be cleared from disk. “Yes / Clear” will update the archive and delete preview file(s) from disk, choosing “Yes” will update the archive and preserve preview files, to be able to re-open them faster for further edits. From context menu More, “Update edited files in archive” group of entries it is possible to manually request to update the archive and/or clear preview files.

Span archive in smaller files of a pre-defined size

While creating an archive as explained in previous points, click on “Single volume” dropdown menu to select a size for output files (volumes) the archive will be split in. Most common types of archives supports this option.

Create encrypted archives

Click on the padlock icon to enter a password and optionally a keyfile; the icon is featured both in PeaZip’s file browser’s status bar and in archive creation interface. To hide names of files and directories contained in the archive check “Encrypt also filenames”, please note that it will be applied only if the archive will be created in a format supporting this function, like 7Z and ARC. Note that in archive creation interface, alongside the padlock icon, there is a colour line (red for weak password, green for strong password) at the bottom of the form to visually inform if encryption is set and if the current archive format supports encryption.

Archive items in separate archives at once

Add objects to be archived (with PeaZip’s "Add" button, or from system’s context menu or SendTo menu) and before confirming with "Ok" check "Add to separate archives" option

Convert single or multiple archives

From PeaZip select archives to be converted and click on "Convert" on toolbar or context menu, non-archive files and folders can be added as well, the difference being in archives being extracted before compression stage. Using the “Convert existing archives” switch in conjunction with “Add to separate archives" (default) it performs a mass conversion of listed archives, without that switch archive conversion is meant to consolidate input data in a single archive, improving compression efficiency because it allows recompression of the original data from its uncompressed form.

Directly create an archive of specified format with given compression tool

It is possible to directly add files/folders to a new archive of a specific format, and with specific compression level, from presets in the dropdown menu on the right of Add button. Most common presets are also available during installation / configuration as context menu entries

Export tasks as scripts

PeaZip is meant to allow to easily export the tasks defined in the GUI as **command line scripts**, from Console tab in archive creation and extraction screens, see “Save and schedule tasks” chapter for more information. This allows to thorough inspection, later re-use, customization beyond the capability of the UI, and helps learning the syntax of the underlying binaries. Also PeaZip itself is easily scriptable, see “Customisation and scripting” chapter for directly accessing PeaZip functions from the command line.

Quick access to PeaZip functions

Drag & drop files and folders to PeaZip or a shortcut to a function, e.g. Add to archive, or Extract here You can drag and drop single or multiple items (files and folders) on PeaZip, or to a shortcut of PeaZip (which can point to a specific function), to easily pass the entire list of files & folders to the application. Dropping files & folders on a PeaZip’s shortcuts which points to a specific function, e.g. Add to archive, Add to ZIP, Extract here, Open as archive, Extract here (smart new folder), Extract…, etc **immediately send the item(s) to such function**. This method works on all supported platforms, and also works for PeaZip Portable packages as it does not require any kind of system integration – and does not leave tracks on the host system.

Set custom default action double-clicking an associated file, e.g. extract here instead of open
From Options > Settings, Archive manager tab is possible to define **Default action on input** for **the application, the action which is performed when a file associated with PeaZip is opened with double click**, or when single or multiple files associated with PeaZip are opened from “Open” context menu. By default action is set to open (browse) the input file(s), but it is also possible to directly send them to the extraction screen (selecting Extract… action instead of Open), or even to set the application to directly extract input files as it usually done by archive managers on macOS or Gnome, selecting “Extract here”, or “Extract here (smart new folder)”. In this way, double-clicking a file associated with PeaZip, or using a generic “Open” context menu item, the content of the file(s) will be immediately extracted without requiring further interaction.

Use system-wide keyboard shortcuts

On Windows you can create **system-wide keyboard shortcuts** pointing to any of PeaZip’s internal functions made available as shortcuts, as ones mentioned in the previous paragraph. Righ-click on one of shortcuts, click Properties, in Shortcut Key field press the key combination (e.g. Ctrl+Alt+E for Extract…) you would like to use to directly access the function. Unfortunately this method does not allow to pass selected items to the function, but it is still useful to open a specific PeaZip’s function directly from the keyboard – e.g. to open an empty archive creation

layout, or an empty archive extraction screen, in which input files & folders can be dropped from the system. For the installable version it is also possible to access to PeaZip's shortcuts from the context menu, or from application's folder (open command prompt and type: explorer shell:AppsFolder).

Context menu

PeaZip can be integrated with Linux, macOS, and Windows contextual menu (right click). Sample scripts to customize system integration are available in (peazip)/res/share/batch folder, being (peazip) the folder where PeaZip is installed, or where PeaZip Portable is placed – more information is available on Customization and scripting chapter. Fore reference usually PeaZip is installed in:

- Linux: /usr/share/peazip for non-binary files as the aforementioned batch scripts
- macOS: /Applications/peazip.app (right-clicking on peazip.app it is possible to browse PeaZip's directory)
- Windows: C:\Program Files\PeaZip\ (or different drive letter if system disk is not C:)

Image 1: examples of menu integration in Linux; see (peazip)/res/share/batch/ for sample scripts; freedesktop_integration/ subfolder contains .desktop files and other scripts allowing to integrate the application on Gnome, KDE, Xfce, Cinnamon and other DE following freedesktop standards.

On **macOS**, .workflow Automator scripts for Service menus are made available in the root of the DMG, installation is optional. To install a .workflow item, double-click on it. To uninstall a .workflow item, follow system's instruction from context menu customization entry, or simply delete it from ~/Library/Services/ directory inside user's home. A backup copy of the scripts is available in the aforementioned (peazip)/res/share/batch folder.

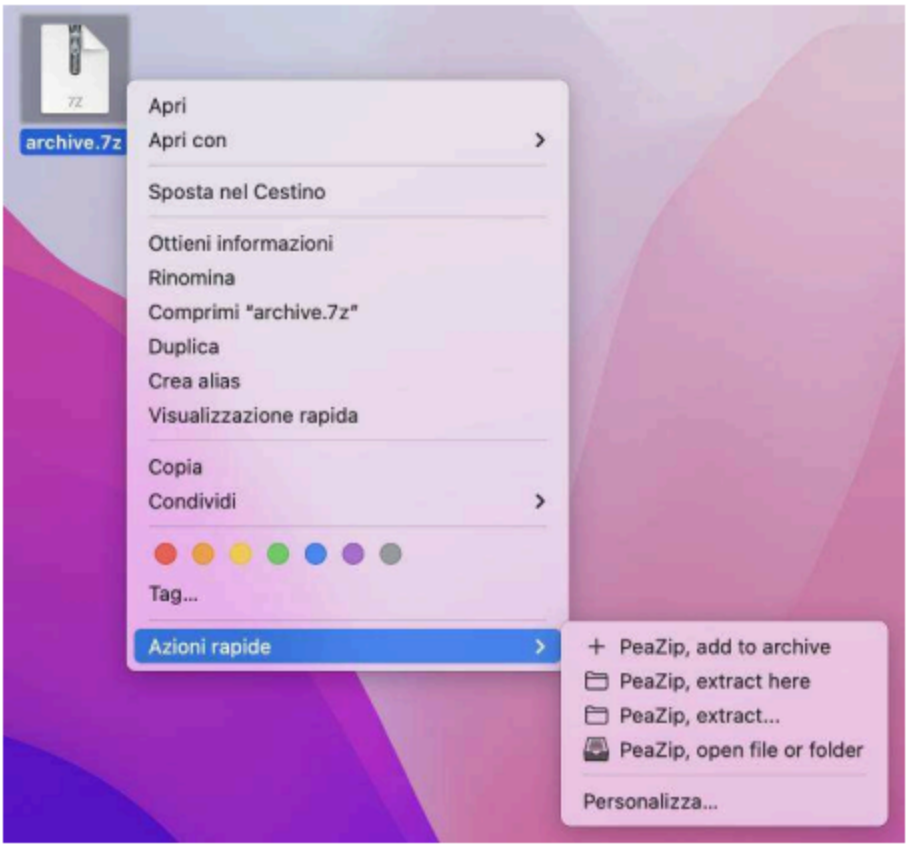


Image 2: Service menus on macOS Monterey

On Windows platform, the installer takes care to create context menu and SendTo menu items, which can be customized during the installation, or later running main menu > Options > File associations and system menu integration wizard. The Windows context menu items type created by PeaZip can accept up to 99 selected items at once as input, and should not be displayed by the system otherwise (but this latter behaviour may change on different Windows versions and updates). As alternative, you can either drag and drop 100+ items from the system to PeaZip, or (faster alternative) select them from PeaZip's own file manager - hint: to directly open current path in PeaZip use system context menu entry "Browse path with PeaZip". On Windows 11 it is possible to add PeaZip entries to the mini context menu (Image 3b) with simple .reg scripts which uses whitelisted IDs, e.g. SetDesktopWallpaper. The scripts are provided as example in the aforementioned (peazip)/res/share/batch folder, and are natively capable to pass multiple input items to a single program's instance. The scripts can be modified to be used with PeaZip Portable too, pointing them to the actual path of the application (as well as the Linux and macOS context menu integration scripts). It is possible to localize the language of the menu entries changing the text in MUIVerb entries of the scripts. All context menu and SenTo menu entries are available from "Show more options", or Shift+F10, in Windows 11 full context menu, and are not affected by mini context menu scripts.

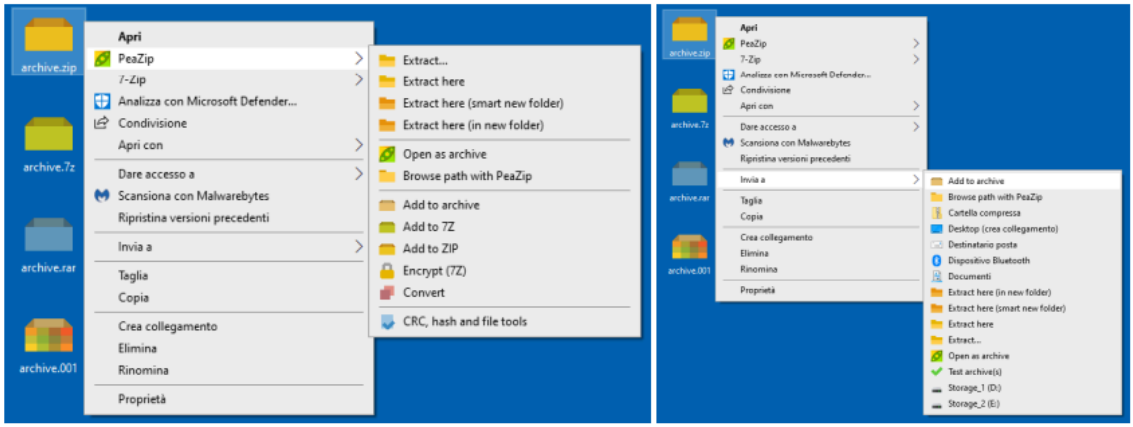


Image 3: high resolution icons, SendTo and context menu integration. The context menu is optionally cascaded, only for Windows 7 or newer systems that supports cascading menus and display icons through static registry entry, which is now recommended over COM methods. System integration can be customized running the installer or in PeaZip in Options > System integration, choosing “Custom installation”.

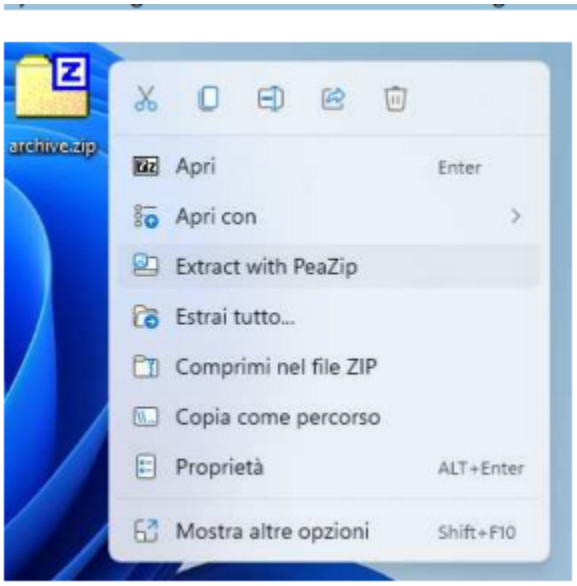
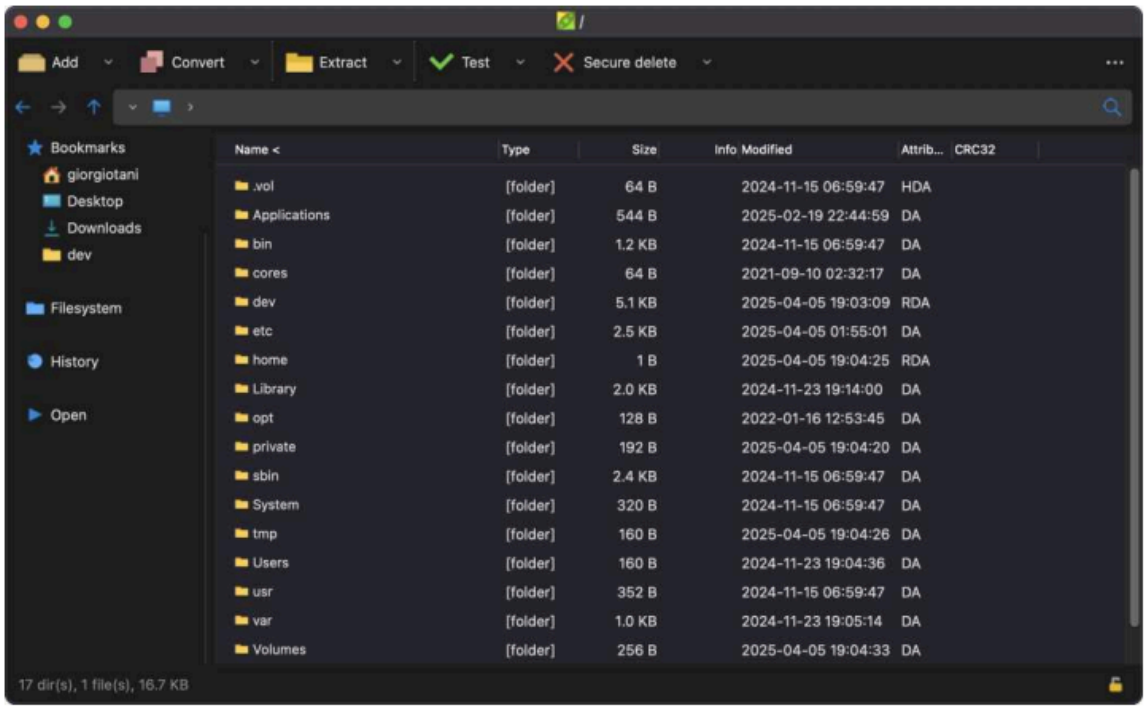


Image 3b: Windows 11 mini context menu integration

Dark Mode

PeaZip uses system’s colors, so it does natively support dark mode on all platforms. On **BSD, Linux, macOS** and **Windows** it is sufficient to set the system to use a dark theme to use same colours on PeaZip. In some instances this may not happen, if the app is running in a way which prevents it to receive correct system colors - virtualization, emulation, containers, some package formats (e.g. Flatpak), depending on how the host system is configured.

Image 4: Peazip in dark mode on macOS



Please note that on Windows 10 and more recent systems the dark mode is set separately for ”modern” apps (from Personalization > Colors) and for Win32 apps (from Settings > Ease of Access > High contrast) and for PeaZip supports both methods, the first provides better look&feel, while the second allows to apply dark color scheme to all Win32 apps at once, even ones not supporting the new dark mode method.

On older Windows systems it is sufficient to set a dark Windows Theme (or a dark high contrast theme) and PeaZip will apply the same dark colors scheme.

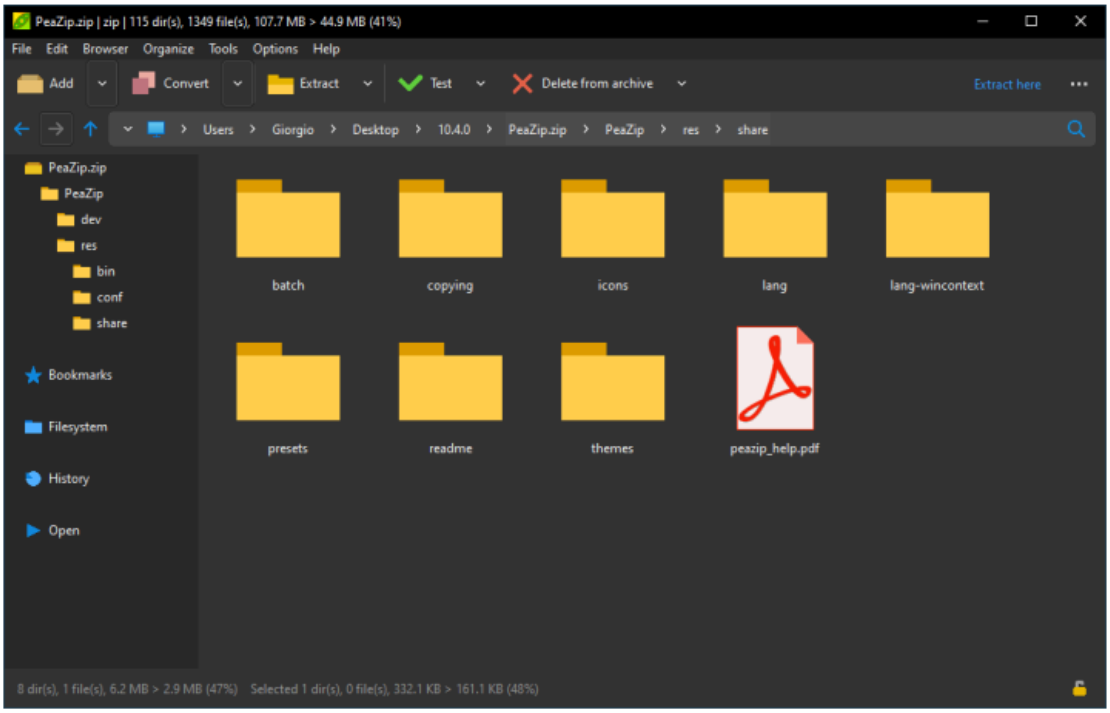


Image 6a: Peazip on Windows in dark mode (large icons mode, browsing archive)

It is also possible to **manually force PeaZip** to **run in light or dark mode** regardless the system settings, creating an empty file named "light" or "dark" in the same path of the executable (this mechanism is not integrated with Themes due to technical limitations). As alternative method to reduce the brightness of PeaZip's app it is possible to change background color from main menu Options > Settings, Theme tab, from Color control in Advanced (hidden) group. On the right of the Color button it is available a list of pre-set values to quickly reduce brightness of the app, either with a neutral grey tone or applying tints like sepia, plum, etc.

Tools submenu contains:

- **Password Manager** (Shift + F9) encrypted password manager
- **Set password / keyfile** (F9) sets the default password (and optionally keyfile, if two factor authentication is desired) to be used in browsing, testing, extraction and archive creation
- **Create random password / keyfile** (Ctrl+F9) allows to sample entropy from the system and from user's actions to generate a random keyfile; this utility can be also used to generate random passwords to be used in any other application/website/etc
- **Verify hash of binaries** check SHA256 hash (which is cryptographically strong) of backend binaries invoked by PeaZip against known values for known versions of those binaries, in order to identify if any one of the binary files has been replaced. Executable files, libraries (.dll, .so, etc), and sfx modules are verified for integrity in this test.

Please note end users are still able to freely replace backend binaries with custom builds best fitting their own needs, this tool will simply report what binaries have been modified (unless PeaZip binary is modified from sources and recompiled to recognize the new hash values). In this way users with stricter security needs can know on demand if unknown binaries are in use in their copy of the app.

• It is also possible to harden the PeaZip at compile time defining fixed paths for binaries, configuration, and nonbinary resources, setting constants HBINPATH, HCONFPATH, and HSHAREPATH. In this way it is possible to force the application to use resources from paths enforcing the desired set of security rules, so binaries cannot be modified unless said security rules are fullfilled.

- **Comment** (ALT + M) shows an editor which read and allows to modify the archive-level comment in RAR (if WinRar is featured on the system), ZIP, and ZIPX files. The comment editor can read the comment of the (first) selected file, and can write the comment to multiple archives at once (number of selected archives is shown in window title bar) even of mixed types, and load / save comment to text file. Comments can contain UTF-8 characters and can be up to 64KB in size. Comments are not supported in multi-volume ZIP/ZIPX files (consistently with7-Zip and WinRar), and RAR comments cannot currently being read from (but can be written to) multi-volume RAR files or legacy RAR4 format. *Comment cannot be currently set for RAR files with encrypted TOC – but can be correctly read.*

Image 8a: comment editor

o Use advanced filters (Shift+Ctrl+F9) sets multiple inclusion and exclusion filters to be used in browsing, testing, extraction and archive creation; filters are applied only to archive formats managed through 7z backend interface, see “Supported file types” chapter o Load Layout, Save Layout, Saved Layouts allows to work with current Layout definition, which for file manager only contains Advanced filters. o Saved and scheduled task definition scripts opens folder containing task definitions saved as script files / scheduled as system task, created from archiving or extraction interfaces, that can be run for unattended repetition of user-defined operations

- **System benchmark (7Z)** LZMA-based test to rate the host system in terms of MIPS (millions of integer instructions per second) and Core 2 Duo equivalent speed in MHz
- **PeaUtils**, standalone GUI for file management tools, if a reduced scope user interface is preferred.

“Help” submenu opens settings & help system interface, pointing to check for updates, Plugin, Themes, and Translations in project’s website, and providing access to most up to date documentation available online, and the offline localized PeaZip tutorial (F1), and help file (Alt+F1).

From this submenu it is also possible to visit PeaZip’s Donations page, which links to funding pages of a few suggested charitable organizations. This allow to turn the perceived project’s value in real help to whom it need it most.

Tool Bar

On the right side of application's toolbar the "…" Style button allows to customize application's GUI and zoom level. In this menu it is possible to toggle the toolbar between Archive Manager, File Manager, and Image manager, to display large, medium, or small version of tool bar, change side navigation bar to filesystem treeview, change status bar functionalities (show history, bookmarks, clipboard…), change breadcrumb bar style (plain text, Windows/KDE; Gnome, macOS) and to customize file browser style (details, list, icons) and size. Style submenu allows to change multiple parameters at once to fit one of the pre-set file manager styles (Classic, Modern, etc…). It also serves as alternative way to show the Main menu - which can be optionally hidden - as a popup menu. If the tool bar is hidden (size set to None or Compact), the Style button is automatically shown on the right side of the Address bar. If the tool bar size is set to Compact, a small selection of the tool bar buttons is shown on the right side of the address bar. On the right of the tool bar (before Style button) it is shown "Extract here" quick extraction link, when browsing archive files. The link extracts selected content, or entire archive if nothing is selected. Extraction is directed to new/smart folder depending on settings, and the link text is adjusted accordingly. Right-clicking on the link it is possible to choose to extract here, or to smart/new folder, on the fly without changing saved options.

Image 9: Archive Manager tool bar.

The Archive Manager toolbar features: • Add o While browsing the filesystem, the button adds selected files and folders to the current archive layout; before confirming the creation of the archive with "Ok" it is possible to modify the list of objects to be archived (dragging them or using the context menu) as well as other options, output name etc, see "Create archive" chapter for more information. o While browsing an existing (writeable) archive, the button brings to the archive update interface; it is possible to add files and folders to be added to the archive as in the previous case, dragging them or using context menu. • On the right of add button a dropdown menu features most common compression tasks and most used formats, allowing to send items from the file browser to compression screen, loading in the meantime the chosen custom compression settings. "Immediate execution" option, if flagged, allows to start compression without requiring to click on "OK" button for confirmation, but as downside it does not allow further customization of the task. • Convert o Convert selected archives in the desired format (non-archive files and folders can be added as well, the difference being in archives being extracted before compression stage). Using with "Add to separate archives" (default) it performs a mass conversion of listed archives, without that switch archive conversion is meant to consolidate input data in a single archive, improving compression efficiency because it allows recompression of the original data from its uncompressed form.

On the right of convert button, a dropdown menu similar to archiving one allows to directly send items to conversion operation with most common pre-set parameters. • Extract o While browsing the filesystem, extracts all selected archives; before conforming the extraction with

“Ok” it is possible to modify output path and other options, and to add other archives to be extracted. o While browsing an archive, extracts the selected items or all the displayed items if nothing is selected, like WinRar/7-Zip (it is possible to be sure to extract the whole archive using context menu’s Extract all entry, or clicking “Extract all” link in extraction interface); before confirming the extraction with “Ok” it is possible to modify output path and other options. o Right clicking Extract button shows a menu to extract all and (when browsing archives) displayed or selected objects, for partial extraction from archives. • On the right of extract button, an arrow shows a menu with functions to directly extract all the content of the archive to most common destinations without further interaction with extraction confirmation interface o In the upper area are listed most common extraction operations: extract all here (to new folder, smart new folder), extract all to allowing to select output directory and proceed with extraction without leaving the file browser UI, extract all to desktop, extract all to documents, extract to 1..3° bookmarked paths (if defined), and extract to default extraction path. Only first, second and third bookmarks are shown as quick extraction destination to not clutter the menu, but bookmarks 1..8 are accessible with keyboard shortcuts Ctrl+1..8. Ctrl+0 extracts to the previous output path.

- In the lower area it is possible to set most common extraction options: extract to new folder, smart new folder, naming policy (only for formats supported through 7z frontend), open output path after the task completes, set default extraction path…

- **Test**

- Test selected archives for integrity • On the right of test button, a submenu shows more informative functions: quick info or full listing of archive, system’s file properties dialog, analysis of files and directory (showing recursively directory content and size), check Zone.Identifier (Mark of The Web) in selected files, file checksum/hash including functions to save reports and to search hash values online (for possible known malware matches), and hexadecimal view of file content. In “File browser” section are shown functions displaying the result in the file manager windows, as “Find duplicate files”. On the bottom of the menu are featured software configured for antivirus / antimalware scan (automatically detected or as user-defined in advanced applications list)

- **Secure delete**
    - Erase selected files and folders (see File Tools chapter). Inside archives it becomes Delete from archive button. Please note that in both cases the files being securely deleted, or deleted from archive, cannot be recovered from the recycle bin.

- On the right, an arrow shows a menu with other functions replicating some ones featured in context menu, explained in details in context menu section of this chapter:
    - File / archive manager functions
        - Copy to and Move to, create new folder, calculate file checksum, Quick delete, Zero delete, and Secure delete, zero or secure delete free space.

- Misc functions: Add to bookmarks, Web search (search selected item's name on the web), Explore path and Open command prompt here, to open the path being currently browsed with Explorer (or other default file manager) and command prompt respectively, Properties.

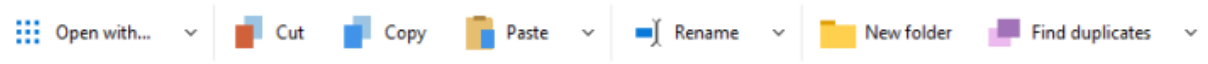


Image 10: File Manager tool bar.

The File Manager toolbar features 4 areas. The first area contains Open with button, and dropdown menu showing custom applications (can be defined by the user with Settings > Applications). If nothing is selected, this dropdown becomes a launcher for the custom applications. The second button group contains Copy/Paste buttons; Cut, Copy to, Move to, Copy path, and Select… (advanced selection dialog) are featured in the dropdown menu on the right. Third group contains Rename button and a dropdown menu featuring a set of common renaming operations: convert the filename(s) in lowercase or uppercase, cut filenames at specified length, append or prepend directory name, number files accordingly to current sorting order, add a string (or delete n characters) at specified position, replace or remove all occurrences of a string (optionally case sensitive), replace or remove all non-AXCII characters, change file extension. Rename functions are also featured in context menu (File manager > Rename submenu). The last area contains various file management features, similar to ones featured in the last dropdown button of Archive manager toolbar; create new folder and find duplicate files (see below File tools chapter) are featured as buttons.

Image 11: Image Manager tool bar.

Image 11: Image Manager tool bar.

Navigation side bar

To quickly jump to desired directory or archive PeaZip offers a Side bar on the left side of the file browser (see Image 7 on page 11, left area), it can be toggled with ALT+4 or can be set from Style menu clicking "…" button on the top right of the tool bar. Available modes are: **Navigation** showing shortcuts to system paths, user-defined bookmarks, recently visited archives, and most common app's functions, **Treeview** showing the classic directory tree, **Compact to** shows icon links to most commonly used filesystem paths, and Hidden.

While browsing an archive file, the Navigation bar will show in-archive directory tree structure on the top of usual navigation shortcuts (bookmarks, system, recent, operations), that can be used

to quickly navigate the archive. A context menu item allows to expand / collapse the structure in a single click in order to immediately explore the structure of the archive.

Status bar

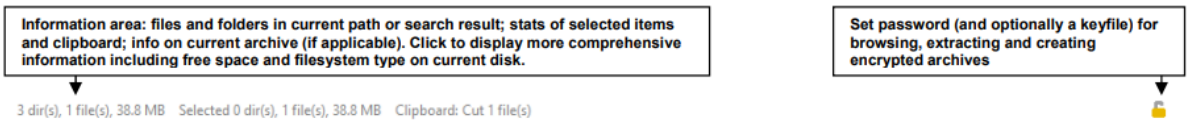


Image 14: status bar; can be right-clicked to display alternative views (bookmarks, Clipboard, etc...)

Keyfile is not mandatory, it can be used if two factor authentication is preferred to password-only authentication; keyfile creation utility can be launched from Tools menu or with Ctrl+F9.

If the directory structure area of the archive is encrypted (i.e. .7z archives created with –mhe option), browsing is not possible until the correct password/keyfile is provided: the archive browser will be empty and "no matches" will be displayed in the status bar until it becomes possible to browse the archive, having the user provided the right password. Please note the same can happen when the archive cannot be browsed for other reasons, i.e. it was corrupted due to bad download or storage media failure. When strong encryption is involved, it may not be possible to determinate if the provided password is incorrect or if the file is corrupted, since resulting output will be random-looking in both cases.

Thumbnails

From "…" Style menu in tool bar it is possible to set file browser style (Details, List, and Icons, toggled with Alt+6) and size (Alt+7). Sizes are 16px, 24px, 32px, 48px, 64px, 72px, and 96px for the Details and List modes, and 48px, 64px, 72px, 96px, 128px, 144px, and 192px for the Icon mode. With any icon size of 48px or larger, PeaZip will **show picture thumbnails** instead of the generic picture icon. From main menu > Organize, it is possible to toggle on/off showing thumbnails, regardless the file browser icon size. On Windows, thumbnails calculation generates a set of 4 dedicated threads, if folder or search filter is changed a new set of threads is created and old threads are stopped as soon as possible (file level and thread level granularity: each thread can be stopped separately, at the completion of each image). As alternative, from context menu > File browser, "**View thumbnails**" entry (Ctrl+Space keyboard shortcut) it is possible to make the file manager showing pictures thumbnails on user request (regardless icon size and thumbnails settings). It is possible to stop on demand thumbnails generation with Esc key, refresh, or leaving the directory.

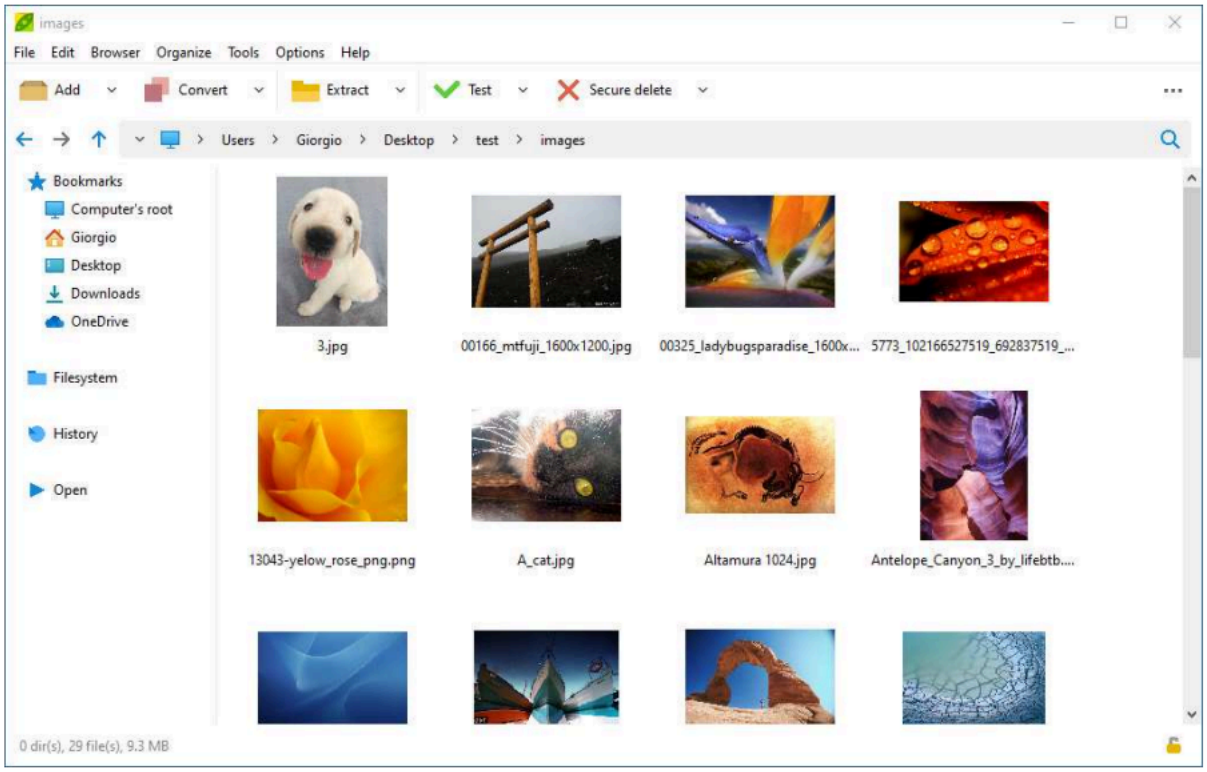


Image viewer

PeaZip provides an integrated image viewer, which can be accessed from context menu > File browser, “View images” entry (Ctrl+Space keyboard shortcut).

The image viewer can:

- zoom in (+ or up arrow) and out (- or down arrow) from 5% to 1000%, fit to screen (0), screen height (Shift+0), screen width (Ctrl+0), jump to 100% to 500% (1..5) and down to 20% (Ctrl+2..Ctrl+5) zoom
- toggle immersive mode on/off (double click or Enter); Esc exits immersive mode or exits the image viewer if not in immersive mode
- view previous (back arrow or clicking the left active border) and next image (forward arrow or Space or clicking the right active border), or jump to first (Home) or last (End) image in the current directory or search filter, applying the same sorting of the file browser
- rename and delete images from disk with all supported deletion routines (move to Trash, quick delete, zero delete, secure delete)

The component is meant primarily as a viewer: functions to copy / move the file, modify the image, and find duplicates, are featured in the file browser (see the “Image manager” section in Toolbar chapter) and are not replicated in the viewer.

The image viewer **can view images when file browser is in virtual mode only.**

Context Menu

The browser's **context menu**, activated right clicking on the archive browser area, is organized in 4 areas, and is context sensitive as provides different options while browsing the filesystem or archives of various types (which supports different operations).

Archive management

The top area features archive management-related entries: Add, Add to separate archives, Convert, Extract, Test, Info (synthetic information about the archive), List (applies also to uncompressed files/folders), and List (with details) for a in depth report of archive content. While browsing the filesystem Add will send selected files and folders to the archive layout, while browsing an archive it opens a submenu allowing to add files or folders or to open a search dialog from which files/folders can dragged into the archive itself. In this case objects will be archived starting from archive's root, compressed and encrypted accordingly to the archive's settings.

Please note that if it is possible for the archive format to store objects with different passwords into the same archive (i.e. in .7z format), you can set the different passwords each time you add objects.

*Note **adding files/folders to encrypted .7z archives**: .7z archive format can store objects encrypted with different passwords in the same archive, so when adding an object to an encrypted 7z archive the object will be encrypted with the password/keyfile currently set (in the form which popup clicking on the padlock icon). If the .7z archive is encrypted with "Content and filenames" option, objects can be added only using the same password / keyfile for the entire archive.*

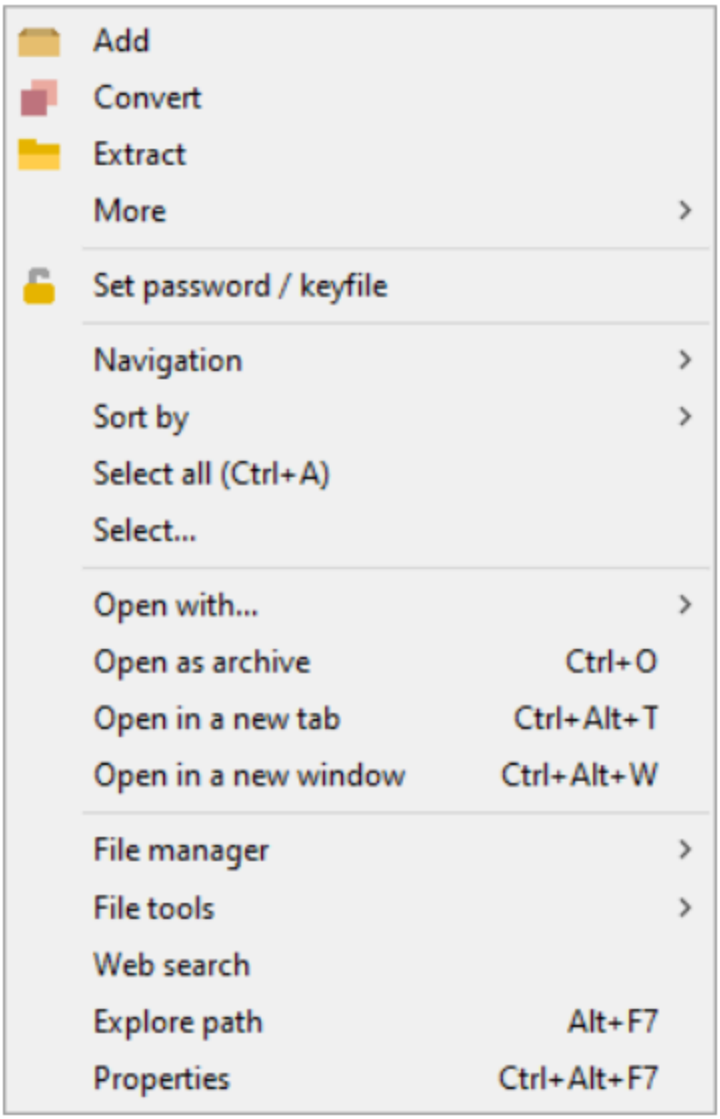


Image 17: Application's context menu

More submenu also contains menu items to **save edited files to archive** on demand, or to discard temp edited files so editing can start from scratch from last version of files saved to archive.

Editing files in archive, when a file is saved (to temp work path) PeaZip will ask to save edited files into the archive; options are:

- **Yes/clear:** save edited files to archive (may require time for large files, or solid archives) and delete temp work files; further edits will require files – which are now synced with edits - to be extracted again
- **No/clear:** do not save edited files to archive and delete temp work files: basically discard all changes and start from scratch (last save)

- **Yes** save edited files to archive (may require time for large files, or solid archives) and keep temp work files for further editing without needing to re-extract them
- **No** do not save edited files to archive but keep temp work files for further editing, edits can then be saved later i.e. from context menu > More group.

ZIP filenames encoding (in More submenu) allows to set a custom codepage to encode filenames in ZIP archives (please note this setting applies to .zip format only) choosing it from a list of known codepages The last menu element. **Text encoding** points to Options > Settings > Advanced, Text encoding options, which contains all text encoding related options, as freely editable custom codepage for ZIP filenames, change encoding for console commands, etc.

On the bottom of the archive management area is displayed "Set password / keyfile": clicking it will display password form in order to set password to open archives, or for creation of new encrypted archives.

Direct extraction

Direct extraction (extract here, to new folder, smart new folder) functions, and Extract to… submenu, are visible only when browsing an archive, or if at least one archive file is selected in the file browser. These entries allows to directly extract all or selected items without further confirmation. Extract to… submenu displays bookmarked paths and history (previously visited paths, session history, and breadcrumb paths) allowing to **directly extract all or selected items to any path saved in bookmarks or history with a single click.**

Browsing / Navigation

The second area contains features related to navigate, and to manage the file browser. "**Navigation**" submenu allows to add / manage Bookmarks, open archives and paths, toggle browse / flat view (show all), set flat view for archives, save paths of selected items, display information about current path, and save the list of files/dirs (including size, date, attributes, checksum if present, etc) to TXT or CSV file. It also provides an alternative access to Settings and to enable/disable main menu, Tool Barr, Side bar, Status Bar etc… This area also contains "**Sort by**" submenu, that allows setting / inverting sorting column (column's header is not available when the browser is in List or Icons mode), "**Select all**" (Ctrl+A) and "Select…" (**advanced selection dialog**).

PeaZip for Darwin / macOS replaces some of the keyboard shortcuts with other ones meant to be more natural for MacOS users:

Move to Trash	Command + Backspace
Toggle show hidden files (in filesystem)	Command+.
Search	Command + F
Select all	Command + A

Copy	Command + C
Cut	Command + X
Paste	Command + V
Create new folder	Shift + Command + N
Up one level	Shift + Command + Up Arrow
Previous item in history	Shift + Command + Back arrow
Next item in history	Shift + Command + Forward arrow
Go to Computer's or archive root	Shift + Command + C
Go to Desktop	Shift + Command + D
Go to Home	Shift + Command + H
Go to Downloads	Shift+Command+L
Go to Documents	Shift+Command+O

Alternative context menus

Ctrl+RightClick breadcrumb context menu
Shift+RightClick session history context menu
Ctrl+Shift+RightClick Navigation context menu

Set password (F9)

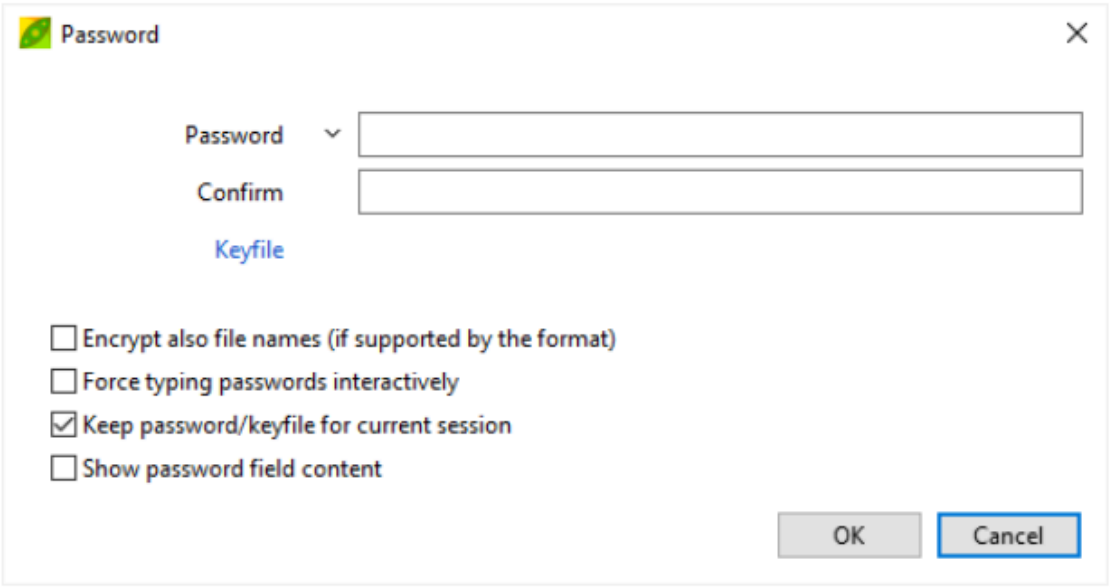


Image 19: Set password (and optional keyfile) prompt

Password form allows to set password and, optionally, a keyfile for two factor authentication. Once a password is entered, it is used by: • file manager, since some encrypted archives needs password for being browsed or tested • archive extraction, to open encrypted archives • archive creation, to create encrypted archive (if the selected format supports encryption); in archive creation interface it is specified if the password is set and if encryption is supported for selected archive format.

Two factor authentication is not applied to sfx archives; as self extracting modules are able to check for password only the keyfile field is ignored when compressing to self executable archives in order to not break compatibility.

If a keyfile is set for any other format than PEA (which has its own way to use keyfile) the SHA256 hash of the file encoded in Base64 (RFC 4648) will be prepended to the password. This convention allows any other file archiver to work on two-factor authentication archives built with PeaZip, even if keyfile parsing is not supported (or has different scheme) simply passing the Base64-encoded SHA256 hash of the keyfile as the first part of the password.

"Encrypt also filenames" option is used during creation of 7Z and ARC formats (PEA format always encrypt filenames regardless this option): if checked, the encrypted archive will need password for being browsed, else the content will be visible; in both cases extraction will require the password. The padlock icon in file browser's status bar (also shown in archiving and extraction interfaces) will change to visually inform that a password is set.

Password options

In the bottom area of the form are listed password options.

"**Encrypt also filenames**" option increases security for file formats (7Z, ARC) supporting this feature. If filenames are not encrypted, an unauthorized attacker may open the archive and get an idea of the content, even not being able to decrypt and extract it, which may be unwanted in many scenarios.

"**Force typing password interactively**" if this feature is supported by the backend for the selected format, PeaZip will require to type each time the password directly in the instance of the backend binary, running the in console mode. This option allows to create scripts that will not run unattended but will rather ask user for interactively providing the password. Please note this option will not allow to browse archives created with "Encrypt also filenames" option, and will set backend binaries to run in native console mode - backends not supporting encryption are not affected by this option.

Notes: 7z / p7zip, FreeArc, and Rar backends support this option, running in console mode. Pea backend supports this option even running in GUI mode. Zpaq backend supports encryption but does not support this option. Other backend does not support encryption so information in this chapter does not apply to them..

“**Show password field**” content makes the password field visible, making easier to set password and not requiring retyping it for verification. However, this way the password is also easier to read for unauthorized bystanders.

How are passwords handled in PeaZip

How are passwords handled in PeaZip Passwords are entered in PeaZip and kept only for the current session of the app - until the app is closed.. Unchecking option “Keep password for the current session” (in password prompt) is more restrictive as it resets the password each time a new archive is opened. Passwords are kept in memory, unless the system decides to save app’s memory in a paging file - preventing this is beyond the possibility of the app. By default passwords are then sent, ephemerally, to the backend binaries (handling each specific archive format) **stdin of the process**, without sending them as command line parameters. This is safer because sending passwords as command line parameters makes them visible, and logged, in the user’s process table and possibly console history, which (depending on the host system configuration) may not met the security requirements desired by the user.

Exceptions where the password is sent in as command line parameter:

- If it is not possible to use stdin input for the target backend binary
    - This exception does apply to FreeArc, Pea, and Zpaq, legacy UnACE and UnRar5 plugins, and custom arbitrary binaries
    - This exception does NOT apply to 7z/p7zip (7Z, 7Z sfx, ZIP, ZIPX, RAR extraction) and WinRar (external, RAR compression)
- Always when using Console mode or GUI+Console mode option (Settings, Advanced tab Backend binaries option group),
- Always when saving the task as command line script (Console tab in extraction and archiving screens). In this case it is also needed to securely handle the saved script file in order to not make it accessible to attackers.

Possible attack vectors

type	Feasibility	Default	Force typing password interactively	Possible mitigation
Binaries and libraries hijacking	Easy to Difficult depending on the system settings	Minimally vulnerable, PeaZip checks binaries on request against known SHA256 values		Use PeaZip from safe paths
Keylogger and similar	Medium	Vulnerable	Vulnerable	Proper system update and maintenance
Memory, PeaZip	Difficult	Vulnerable	Immune	
Memory, backend	Difficult	Vulnerable	Vulnerable	
Pipe between PeaZip and backend	Very difficult	Vulnerable	Minimally vulnerable, only list/test output is passed and never when archive TOC is encrypted	
Process table (and possibly console history)	Easy to Difficult, depending on the system settings	Immune unless using a backend not supporting pipes or exporting tasks as scripts (as specified in documentation)	Immune	Secure system configuration, use "Force typing password interactively" option
Weak password	Unpredictable	Vulnerable	Vulnerable	Never use weak passwords, follow Diceware and similar guidelines, use integrated Password generation tool

Due the multitude of attack vectors in a real-life scenario, it is highly recommended to secure the machine as much as possible if it is intended to be used to handle encrypted data. By default PeaZip is hardened against some of the most common attack vectors, with "Force typing passwords interactively" being more secure and less prone to misuse.

Password manager (Shift + F9)

Password manager is accessible from main menu, Tools > Password manager or from Password form, from drop down menu on the left of Password field. To add or edit password and notes, and for other options, right click on the password list area, or click the menu icon on the right of the tab bar, this will display the password list's context menu Password list, contains passwords and notes (but can be used to store any kind of information as string, i.e. a keyfile name if desired), that can be copied and pasted to any application. First 8 saved passwords will also be displayed in the dropdown menu in Password form: items in the password list can be dragged (from first column) to be sorted so most used password are displayed directly in the dropdown menu. You can right click on the password list to open the options menu (add and remove items, reset, export…), doubleclick to edit an element, and use Ctrl+C to copy a string.

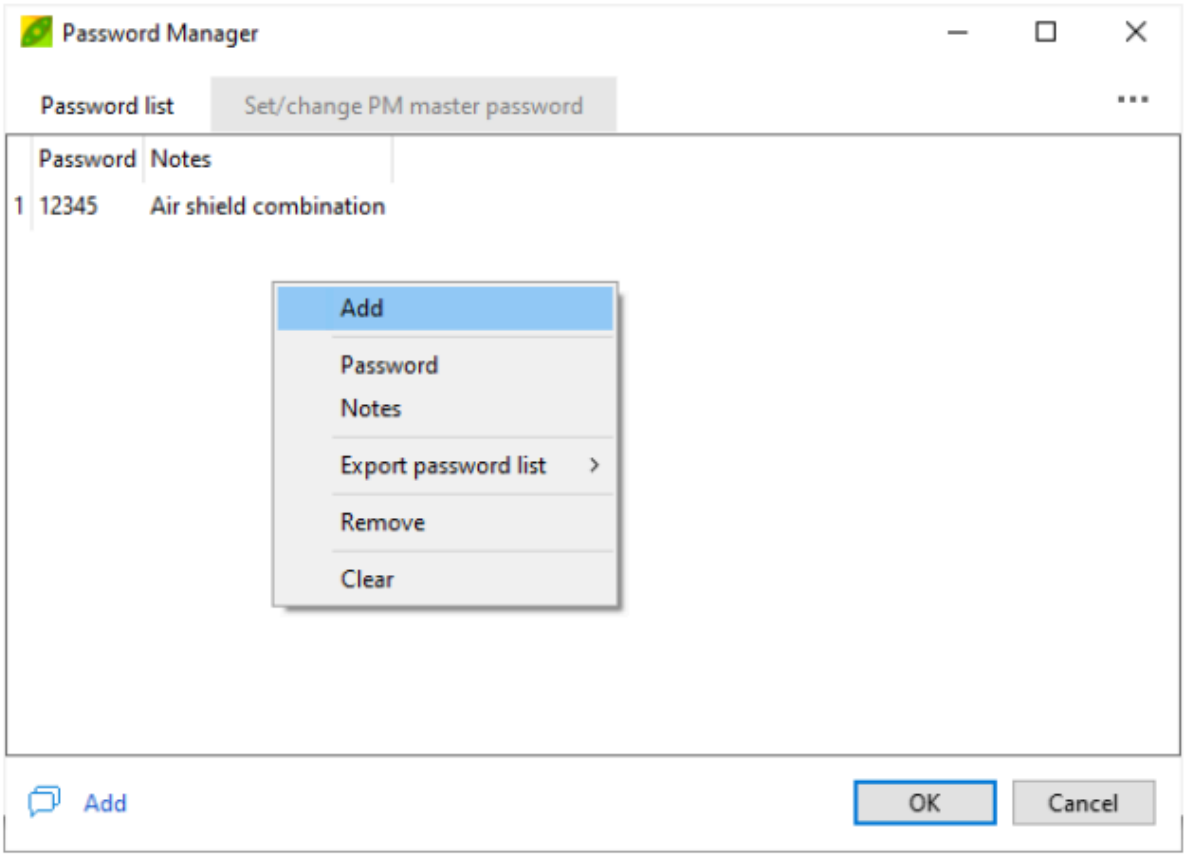


Image 20: Password manager’s Password list tab, showing context menu

Set/change master password

If no password is set the password manager will be accessible without prompting for authentication, otherwise password list tab (and passwords in dropdown menu in Password form) will not be displayed until the user correctly authenticates, accessing to the password manager providing correct master password/keyfile. The Password manager is protected by authenticated encryption (AES256 in EAX mode), and saved as encrypted file “pm” in PeaZip’s configuration folder, if the file is corrupted or tampered the user will be asked if trying to recover it or to delete it. Passwords can be exported as plain text (not recommended) or in its encrypted form, in this way a password list's .bak file can be manually imported as password manager file, renaming it as pm (no extension) and copying it to PeaZip's configuration path (linked in Options > Settings): this file will have the same content and password it had when it was originally saved. This mechanism allows in example to roll back to a previous password list, or maintain various off line password lists, or exchange protected password lists with other users or other machines. Note: Password Manager is encrypted by PeaZip binary, so its master password is never passed to any external binaries / libraries..

Multiple filters, one per line, can be written in the inclusion and exclusion fields; string delimiters (“ on Windows and ‘ on Linux and other *x systems) are not needed to be explicitly entered by the user. In example, if the user needs to extract (or display) only “myfile.txt” plus all files named “your file” and all .mp3 files, but not .mp3 starting with “a” and “m”, could write in the inclusion field:

Myfile.txt
your file
.* *.mp3

and in the exclusion field:

a*.mp3
m*.mp3

To exclude directories, use the syntax dirname\ which excludes all objects contained in the directory named “dirname”. This can be combined with wildcards: dirname*\ excludes all directories starting with “dirname” string *dirname*\ excludes all directories and subdirectories containing “dirname” in name or path *dirname* excludes all files and folders containing *dirname* in name or path To selectively exclude subdirectories, the filter in previous example can be modified as dirname\subfoldername\ to exclude all items inside “subfoldername” which is inside “dirname”.

Extract

This is the main screen for extraction tasks, containing most commonly used parameters and Ok and Cancel button to confirm start or to discard the current operation. Clicking Ok saves modifications to extraction options, while Cancel discards the modifications; it is also possible to confirm changes with an empty layout in order to modify PeaZip’s behaviour without the need to start a task.

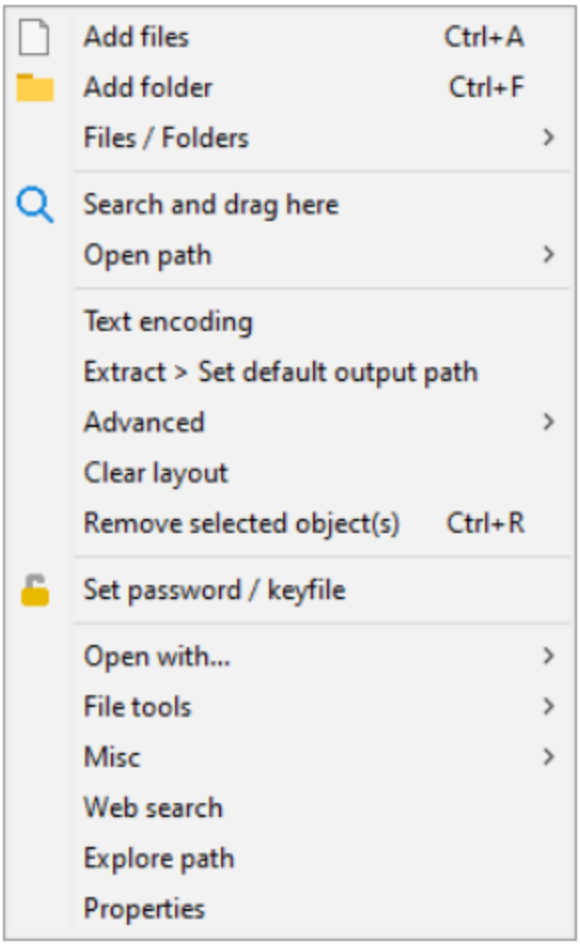
Hint: Remember that for simply changing extraction options, it is possible to click on “Extract” button with no input selected (empty layout), modify options and save with OK button

In the top area of the tab is shown the Input area, where are listed the archives that are going to be extracted (extraction layout), with the total number of archives and total size; additional archives can be dragged here to be added to the list. Objects in the archive layout can be sorted by name, full name, size, extension, type, attributes etc, clicking on titles in archive layout’s title bar. If extracting items which are stored inside an archive, the layout will display the items listed to be extracted from the archive, and will not accept modifications from drag and drop or context menu, and it will be available “Extract all” link below the input list to turn a partial extraction in extraction of the entire archive if desired. Please note that PeaZip will, by default, extract to a temporary work path (which can be set from Options > Settings) and then move extracted content to the output directory, interactively asking to overwrite existing files if needed. This

behaviour can be changed from Options > Settings > Archive manager, Extraction section, unchecking “Extract then move to destination” option. This will allow unattended extraction tasks, with existing files being handles as specified in the naming conflict policy (set in bottom dropdown menu on this screen) without requiring further interaction from the user.

Right-clicking on the input list shows the contextual menu, which features functions to add archives to extraction layout (add files & folders, add items from bookmarks and history, load layout…), and other related functions. Files can be simply dragged from the system to the application: to help file selection “Search and drag here” opens a standard system’s search prompt, and “Open path” submenu links system’s file explorer to paths of bookmarked and recent items. The layout can be saved to a UTF-8 text file (for maximum flexibility of use); when a layout is loaded each object is checked (must exist, duplicates are skipped). From the context menu it is also possible to remove objects from the archive’s layout (“Remove selected objects” and “Clear layout”) and to explore selected object’s path. ZIP filenames encoding replicates a control to set a custom codepage to encode filenames in ZIP archives (please note this setting applies to .zip format only) from a list of known codepages; last menu element brings to Settings, Advanced tab which allows to set all text encoding related options. Extract > Set default output path set all archive extraction tasks to point by default to the chosen path. “Go to file browser” can be used to return to file browsing without discarding the current list of archives, in example to navigate and search for other archives to be added to the list from file browser interface. “Open with…” submenu of context menu allows opening the selected object with PeaZip, associated application, or a custom applications. “File Tools” submenu allows quick access to some of PeaZip functions to be applied on selected objects (see “File tools” chapter).

When extracting entire archives (single or multiple files selected) the complete context menu is shown, while when extracting items from a single archive non relevant entries are hidden. In the bottom area of the form is featured Output group, that allows to select output directory; on the left of the address box, the arrow button popup a navigation menu similar to the one featured in file manager, to speed up the selection of most used paths. The extraction’s navigation menu is organized in Bookmarks, Filesystem and Recent submenus, containing only directory entries. The menu also features “Use default output path” entry to select an output path to be used by default in place of archive’s path, useful if the user often extracts to the same path.



When this feature is active, the “Use default output path” entry is flagged, and can be clicked again to remove the flag and deactivate the option.

On the right of Output field it is featured Set password / keyfile button to set the decryption password, if needed for extraction of encrypted archives. If password is not set, the padlock icon of this button is greyed, otherwise the icon is displayed with normal colours and the bottom of the form will be highlighted with a colour representing the strength (automatically evaluated entropy) of the password, red for easy password, green for more secure ones.

Dropdown menus (on the bottom left of output area) features output options for 7z backend, that is used to support most common file formats (“Advanced” tab contains parameters for other backend executables).

- **Function,** possible actions performed by the extraction routine:
    - “Extract” extracts archived objects with paths, replicating the directory structure of the input data;
    - “Extract (without path)” will extract all archived files to the same path;
    - “List” will show archive’s content;

    - “List (with details)” will give a more detailed report on archive’s content, the same given by “Info” entries in context menu. List functions will always be performed in pipe mode (even if 7z option is set to “console mode”), using graphical wrapper in order to make easier reading and saving the report.
    - “Test” will perform type specific tests to prove or disprove archive’s integrity.

- What the extraction routine will do **in case of naming conflict** while extracting data. Anyway, using “Extract in new folder” switch will assure avoiding naming conflict.

    - “Skip existing files” assure that pre-existing objects are not touched by the extraction operation, being the conflicting objects not extracted from the archive (default).
    - “Auto rename extracted files” assign a new unique name to objects being extracted from the archive each time a naming conflict is encountered; that policy assures that pre-existing objects will keep their names and new ones will get new unique names.
    - “Auto rename existing files” assures that extracted objects get the desired name while pre-existing objects are renamed with a new unique name.
    - “Overwrite existing files” make all pre-existing objects overwritten by extracted objects.
    - “Ask before overwriting (in console)”: no defined policy, the console will be shown alongside graphic process launcher allowing the user to decide interactively if overwriting or skipping files.

Format-specific options group for Zpaq, Brotli and Zstandard becomes visible when extracting file(s) of those types, or if at least one file of those types is listed for extraction. From this group it is possible to force overwriting existing files (off by default) for Zpaq Brotli, and Zstandard. Zpaq backend can also be set to extract to absolute paths, full paths, or to extract all revisions of files in the archive – option which is mutually incompatible with extract to absolute paths option. Please note that Zpaq extraction to absolute paths is set on by default for non-Windows systems, as on non-Windows systems the format is quite often used to create packages meant to be extracted to specific locations, rather than archives meant to be extracted in custom paths.

Checkboxes (on the bottom right of output area), can set following options:

Delete archives after extraction if no error was detected removes original archives after complete extraction process. Only enabled if “extract all” is selected. The user is asked for confirmation before proceeding with deletion unless confirmation request is unchecked in Options > Settings > Archive manager. If this option is in use, a dropdown menu on the right of the checkbox allows to set the type of deletion to be performed: move to recycle bin, quick delete (files are deleted without being moved to recycle bin), zero delete (files are overwritten by 0s, useful to optimize compressibility of disk images, if the host system is a virtual machine), secure delete (multiple pass overwriting of data).

Extract in new folder checkbox, when checked, triggers extraction to a fresh new folder, avoiding possible naming conflict and unwanted “tarbomb” effect (when an unexpectedly large number of files is extracted in current path getting mixed with existing files).

Extract to original path option can be used when multiple archives from different paths are listed for extraction, in order to perform extraction of each archive in its original input path.

Extract then move to destination option can be used when multiple archives from different paths are listed for extraction, in order to perform extraction of each archive in its original input path. option (default on) performs a two-step interactive process for all archive formats supported through the 7z / p7zip backend: extract to a fresh temporary path, and then move content to output path. Unchecking the “Extract then move to destination” extraction checkbox, it is performed a single step, policy based extraction: see “in case of naming conflict” section above.

If “Extract then move to destination” is checked 1) naming conflicts will be prompted interactively to the user, as it happens in drag & drop extraction 2) the tree structure of files/dirs in the archive will always been replicated, as with “Extract” switch, and extra nesting levels will be automatically removed, as in drag & drop extraction

The option is always overridden by “Extract to new folder” switch, which is sufficient to guarantee no naming conflict occur.

Please note that “Extract then move to destination” option is always ignored when composing the command line in Console tab, so the script which is generated will not need to rely on extracting to temporary work path first, and will not need user interaction in case it is needed confirmation to manage naming conflict issues.

In case of extraction errors:

For archives supported through 7z / p7zip backend a) if “Extract to new folder” is checked, the output will be kept in the fresh new output folder, where the user can check it to decide whether to keep or discard the content. b) if “Extract to new folder” is NOT checked a. if “Extract then move to destination” is checked, the faulty output will be automatically deleted, unless the user clicks on “Keep files” link in the extraction progress window. b. if “Extract then move to destination” is NOT checked, the output will be kept and in case of naming conflict the appropriate policy will be applied without requiring further user interaction (default skip existing files). For archives supported through different backend • for Pea archives files with errors are always preserved for further examination; extraction is always directed to a fresh new output folder to avoid conflict with existing files • for Zstd archives the option to keep files with errors can be set from the Advanced tab • for other backend (Brotli, Zpaq) the option is not available and if errors are detected extraction will not take place Please be aware that it is generally not recommended to keep files if the extraction procedures detects errors, more so overwriting existing files with them, as resulting output files may be unusable - or even have been

maliciously forged. The option is meant to be used with caution to save any recoverable content, when this is reasonably and safely applicable. Also please note the very same information above applies when extraction is halted by the user for any reason. Even in this case the output should be handled carefully, i.e. partially overwriting existing files with archived content with different level of update may have unpredictable consequences. Less commonly used options are available in the Advanced tab, and even more settings, which needs to be adjusted less frequently (on once for all), are available in Options > Settings, Archive manager page. The status bar contains, on the left, information of free space on current output disk, and input size, and on the right OK and Cancel buttons. Cancelling an operation also means discarding changes in option switches, while confirming with OK saves changes.