

# Rar Archiving

# Introduction

Rar is a powerful and effective archiver created by Eugene Roshal. This extension gives you possibility to read Rar archives but doesn't support writing Rar archives, because this is not supported by UnRar library and is directly prohibited by it's license.

More information about Rar and UnRar can be found at [» http://www.rarlabs.com/](http://www.rarlabs.com/).

# Installing/Configuring

## Requirements

No external libraries are needed to build this extension.

## Installation

Rar is currently available through PECL » <http://pecl.php.net/package/rar>.

Also you can use the PECL installer to install the Rar extension, using the following command: *pecl -v install rar*.

You can always download the tar.gz package and install Rar by hand:

### Example #1 - Rar installation

```
gunzip rar-xxx.tgz
tar -xvf rar-xxx.tar
cd rar-xxx
phpize
./configure && make && make install
```

Windows users will enable *php\_rar.dll* inside of *php.ini* in order to use these functions. The DLL for this PECL extension may be downloaded from either the » [PHP Downloads](#) page or from » <http://pecl4win.php.net/>

## Runtime Configuration

This extension has no configuration directives defined in *php.ini*.

## Resource Types

There is one resource used in Rar extension: a file descriptor returned by [rar\\_open\(\)](#).

# Predefined Constants

The constants below are defined by this extension, and will only be available when the extension has either been compiled into PHP or dynamically loaded at runtime.

**RAR\_HOST\_MSDOS** ( [integer](#) )

**RAR\_HOST\_OS2** ( [integer](#) )

**RAR\_HOST\_WIN32** ( [integer](#) )

**RAR\_HOST\_UNIX** ( [integer](#) )

**RAR\_HOST\_BEOS** ( [integer](#) )

# Examples

## Example #2 - Rar extension overview example

```
<?php

$rar_file = rar_open('example.rar') or die("Can't open Rar archive");

$entries = rar_list($rar_file);

foreach ($entries as $entry) {
    echo 'Filename: ' . $entry->getName() . "\n";
    echo 'Packed size: ' . $entry->getPackedSize() . "\n";
    echo 'Unpacked size: ' . $entry->getUnpackedSize() . "\n";

    $entry->extract('/dir/extract/to/');
}

rar_close($rar_file);

?>
```

This example opens a Rar file archive and extracts each entry to the specified directory.

# Rar Functions

# **`rar_close`**

`rar_close` -- Close Rar archive and free all resources

## **Description**

bool **`rar_close`** ( resource `$rar_file` )

Close Rar archive and free all allocated resources.

## **Parameters**

*`rar_file`*

A Rar file resource, opened with [rar\\_open\(\)](#).

## **Return Values**

Returns **TRUE** on success or **FALSE** on failure.

# rar\_entry\_get

rar\_entry\_get -- Get entry object from the Rar archive

## Description

[RarEntry](#) **rar\_entry\_get** ( resource \$rar\_file, string \$entry\_name )

Get entry object from the Rar archive.

## Parameters

*rar\_file*

A Rar file resource, opened with [rar\\_open\(\)](#).

*entry\_name*

Path to the entry within the Rar archive.

## Return Values

**rar\_get\_entry()** returns entry object or **FALSE** on error.

## Examples

### Example #3 - [rar\\_entry\\_get\(\)](#) example

```
<?php

$rar_file = rar_open('example.rar') or die("Failed to open Rar archive");

$entry = rar_entry_get($rar_file, 'Dir/file.txt') or die("Failed to find
such entry");

print_r($entry);

?>
```



# Rar::extract

Rar::extract -- Extract entry from the archive

## Description

<b>Rar</b>
------------

bool **extract** ( string \$dir [, string \$filepath ] )

[Rar::extract\(\)](#) extracts entry's data to the *dir*. It will create new file in the specified *dir* with the name identical to the entry's name.

## Parameters

*dir*

Path to the directory where files should be extracted.

*filepath*

If parameter *filepath* is specified instead *dir*, [Rar::extract\(\)](#) will extract entry's data to the specified file.

## Return Values

Returns **TRUE** on success or **FALSE** on failure.

## Examples

<b>Example #4 - <a href="#">Rar::extract()</a> example</b>
------------------------------------------------------------

<pre>&lt;?php  \$rar_file = rar_open('example.rar') or die("Failed to open Rar archive");  \$entry = rar_entry_get(\$rar_file, 'Dir/file.txt') or die("Failed to find such entry");  \$entry-&gt;extract('/dir/to'); // create /dir/to/Dir/file.txt \$entry-&gt;extract(false, '/dir/to/new_name.txt'); // create /dir/to/new_name.txt  ?&gt;</pre>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Example #5 - How to extract all files in archive:

```
<?php

/* example by Erik Jenssen aka erix */

$filename = "foobar.rar";
$filepath = "/home/foo/bar/";

$rar_file = rar_open($filepath.$filename);
$list = rar_list($rar_file);
foreach($list as $file) {
    $entry = rar_entry_get($rar_file, $file);
    $entry->extract("."); // extract to the current dir
}
rar_close($rar_file);

?>
```

# Rar::getAttr

Rar::getAttr -- Get attributes of the entry

## Description

Rar
-----

int **getAttr** ( void )

[Rar::getAttr\(\)](#) returns attributes of the archive entry.

## Return Values

Returns the attributes or **FALSE** on error.

## Examples

### Example #6 - [Rar::getAttr\(\)](#) example

```
<?php

$rar_file = rar_open('example.rar') or die("Can't open Rar archive");

$entry = rar_entry_get($rar_file, 'dir/in/the/archive') or die("Can't find
such entry");

$host_os = $entry->getHostOs();
$attr = $entry->getAttr();

switch($host_os) {
    case RAR_HOST_MSDOS:
    case RAR_HOST_OS2:
    case RAR_HOST_WIN32:
    case RAR_HOST_MACOS:
        printf("%c%c%c%c%c%c\n",
            ($attr & 0x08) ? 'V' : '.',
            ($attr & 0x10) ? 'D' : '.',
            ($attr & 0x01) ? 'R' : '.',
            ($attr & 0x02) ? 'H' : '.',
            ($attr & 0x04) ? 'S' : '.',
            ($attr & 0x20) ? 'A' : '.');
        break;
    case RAR_HOST_UNIX:
    case RAR_HOST_BEOS:
        switch ($attr & 0xF000)
        {
```

```

        case 0x4000:
            printf("d");
            break;
        case 0xA000:
            printf("l");
            break;
        default:
            printf("-");
            break;
    }
    printf("%c%c%c%c%c%c%c%c%c%c\n",
        ($attr & 0x0100) ? 'r' : '-',
        ($attr & 0x0080) ? 'w' : '-',
        ($attr & 0x0040) ? (($attr & 0x0800) ? 's':'x') : (($attr &
0x0800) ? 'S':'-'),
        ($attr & 0x0020) ? 'r' : '-',
        ($attr & 0x0010) ? 'w' : '-',
        ($attr & 0x0008) ? (($attr & 0x0400) ? 's':'x') : (($attr &
0x0400) ? 'S':'-'),
        ($attr & 0x0004) ? 'r' : '-',
        ($attr & 0x0002) ? 'w' : '-',
        ($attr & 0x0001) ? 'x' : '-');
    break;
}

rar_close($rar_file);

?>

```

## See Also

- [Rar::getHostOs](#)

# Rar::getCrc

Rar::getCrc -- Get CRC of the entry

## Description

Rar
-----

int **getCrc** ( void )

[Rar::getCrc\(\)](#) returns CRC of the archive entry.

## Return Values

Returns the CRC of the archive entry or **FALSE** on error.

# Rar::getFileInfo

Rar::getFileInfo -- Get entry last modification time

## Description

Rar
-----

string **getFileInfo** ( void )

Gets entry last modification time.

## Return Values

Returns entry last modification time as string in format *YYYY-MM-DD HH:II:SS*, or **FALSE** on error.

# Rar::getHostOs

Rar::getHostOs -- Get entry host OS

## Description

<b>Rar</b>
------------

int **getHostOs** ( void )

[Rar::getHostOs\(\)](#) return code of the host OS of the archive entry.

## Return Values

Returns the code of the host OS, or **FALSE** on error.

## Examples

### Example #7 - [Rar::getHostOs\(\)](#) example

```
<?php

$rar_file = rar_open('example.rar') or die("Failed to open Rar archive");

$entry = rar_entry_get($rar_file, 'Dir/file.txt') or die("Failed to find
such entry");

switch ($entry->getHostOs()) {
    case RAR_HOST_MSDOS:
        echo "MS-DOS\n";
        break;
    case RAR_HOST_OS2:
        echo "OS2\n";
        break;
    case RAR_HOST_WIN32:
        echo "Win32\n";
        break;
    case RAR_HOST_MACOS:
        echo "MacOS\n";
        break;
    case RAR_HOST_UNIX:
        echo "Unix/Linux\n";
        break;
    case RAR_HOST_BEOS:
        echo "BeOS\n";
        break;
}
```

?>



# Rar::getMethod

Rar::getMethod -- Get pack method of the entry

## Description

Rar
-----

int **getMethod** ( void )

[Rar::getMethod\(\)](#) returns number of the method used when adding current archive entry.

## Return Values

Returns the method number or **FALSE** on error.

## Examples

<b>Example #8 - <a href="#">Rar::getMethod()</a> example</b>
--------------------------------------------------------------

<pre>&lt;?php  \$rar_file = rar_open('example.rar') or die("Failed to open Rar archive");  \$entry = rar_entry_get(\$rar_file, 'Dir/file.txt') or die("Failed to find such entry");  echo "Method number: " . \$entry-&gt;getMethod();  ?&gt;</pre>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

# Rar::getName

Rar::getName -- Get name of the entry

## Description

Rar
-----

string **getName** ( void )

[Rar::getName\(\)](#) returns full name of the archive entry.

## Return Values

Returns the entry name as a string, or **FALSE** on error.

## Examples

<b>Example #9 - <a href="#">Rar::getName()</a> example</b>
------------------------------------------------------------

<pre>&lt;?php  \$rar_file = rar_open('example.rar') or die("Failed to open Rar archive");  \$entry = rar_entry_get(\$rar_file, 'Dir/file.txt') or die("Failed to find such entry");  echo "Entry name: " . \$entry-&gt;getName();  ?&gt;</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

# Rar::getPackedSize

Rar::getPackedSize -- Get packed size of the entry

## Description

Rar
-----

int **getPackedSize** ( void )

Get packed size of the archive entry.

## Return Values

Returns the packed size, or **FALSE** on error.

## Examples

<b>Example #10 - <a href="#">Rar::getPackedSize()</a> example</b>
-------------------------------------------------------------------

<pre>&lt;?php  \$rar_file = rar_open('example.rar') or die("Failed to open Rar archive");  \$entry = rar_entry_get(\$rar_file, 'Dir/file.txt') or die("Failed to find such entry");  echo "Packed size of " . \$entry-&gt;getName() . " = " . \$entry-&gt;getPackedSize() . " bytes";  ?&gt;</pre>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

# Rar::getUnpackedSize

Rar::getUnpackedSize -- Get unpacked size of the entry

## Description

Rar
-----

int **getUnpackedSize** ( void )

Get unpacked size of the archive entry.

## Return Values

Returns the unpacked size, or **FALSE** on error.

## Return Values

<b>Example #11 - <a href="#">Rar::getUnpackedSize()</a> example</b>
---------------------------------------------------------------------

<pre>&lt;?php  \$rar_file = rar_open('example.rar') or die("Failed to open Rar archive");  \$entry = rar_entry_get(\$rar_file, 'Dir/file.txt') or die("Failed to find such entry");  echo "Unpacked size of " . \$entry-&gt;getName() . " = " . \$entry-&gt;getPackedSize() . " bytes";  ?&gt;</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

# Rar::getVersion

Rar::getVersion -- Get version of the archiver used to add the entry

## Description

Rar
-----

int **getVersion** ( void )

Get version of the archiver used to add the archive entry.

## Return Values

Returns the version or **FALSE** on error.

## Examples

<b>Example #12 - <a href="#">Rar::getVersion()</a> example</b>
----------------------------------------------------------------

<pre>&lt;?php  \$rar_file = rar_open('example.rar') or die("Failed to open Rar archive");  \$entry = rar_entry_get(\$rar_file, 'Dir/file.txt') or die("Failed to find such entry");  echo "Rar (WinRAR) version used: " . \$entry-&gt;getVersion();  ?&gt;</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

# rar\_list

rar\_list -- Get entries list from the Rar archive

## Description

array **rar\_list** ( resource \$rar\_file )

Get entries list from the Rar archive.

## Parameters

*rar\_file*

A Rar file resource, opened with [rar\\_open\(\)](#).

## Return Values

[rar\\_list\(\)](#) returns array of entries or **FALSE** on error.

## Examples

### Example #13 - [rar\\_list\(\)](#) example

```
<?php
$rar_file = rar_open('example.rar') or die("Failed to open Rar archive");
$entries_list = rar_list($rar_file);
print_r($entries_list);

?>
```

# **rar\_open**

rar\_open -- Open Rar archive

## **Description**

resource **rar\_open** ( string \$filename [, string \$password ] )

Open specified Rar archive and return Rar file resource.

## **Parameters**

*filename*

Path to the Rar archive.

*password*

A plain password, if needed.

## **Return Values**

[rar\\_open\(\)](#) returns Rar file resource or **FALSE** on error.