

Commit 5a55fee



Erlang/OTP committed on Jun 13, 2025

Merge branch 'lukas/stdlib/zip-sanitize-filename/27/OTP-19653' into maint-28

* lukas/stdlib/zip-sanitize-filename/27/OTP-19653:
stdlib: Properly sanatzize filenames when (un)zipping

Conflicts:
lib/stdlib/test/zip_SUITE.erl

🔗 master · 📁 OTP-29.0-rc2 ... OTP-28.0.1

2 parents [4ae92fb](#) + [b49c0b4](#) commit 5a55fee 📄

📁 2 files changed +74 -12 lines changed

↑ Top ⚙️

🔍 Filter files... ☰

- 📁 lib/stdlib
 - 📁 src
 - 📄 zip.erl
 - 📁 test
 - 📄 zip_SUITE.erl

📁 2 files changed +74 -12 lines changed

🔍 Search within code ⚙️

```

lib/stdlib/src/zip.erl
@@ -1240,12 +1240,12 @@ get_filename({Name, _}, Type) ->
1240 1240  get_filename({Name, _, _}, Type) ->
1241 1241      get_filename(Name, Type);
1242 1242  get_filename(Name, regular) ->
1243      -   Name;
1243      +   sanitize_filename(Name);

```

```

1244 1244     get_filename(Name, directory) ->
1245 1245         %% Ensure trailing slash
1246 1246         case lists:reverse(Name) of
1247 -         [$/ | _Rev] -> Name;
1248 -         Rev         -> lists:reverse([$/ | Rev])
+         [$/ | _Rev] -> sanitize_filename(Name);
+         Rev         -> sanitize_filename(lists:reverse([$/ | Rev]))
1249 1249         end.
1250 1250
1251 1251     add_cwd(_CWD, {_Name, _} = F) -> F;
@@ -2368,12 +2368,25 @@ check_dir_level([_Dir | Parts], Level) ->
2368 2368     get_filename_extra(FileNameLen, ExtraLen, B, GPFlag) ->
2369 2369         try
2370 2370             <<BFileName:FileNameLen/binary, BExtra:ExtraLen/binary>> = B,
2371 -         {binary_to_chars(BFileName, GPFlag), BExtra}
+         {sanitize_filename(binary_to_chars(BFileName, GPFlag)), BExtra}
2372 2372         catch
2373 2373             _:_ ->
2374 2374                 throw(bad_file_header)
2375 2375         end.
2376 2376
+ sanitize_filename(Filename) ->
+     case filename:pathtype(Filename) of
+         relative -> Filename;
+         _ ->
+             %% With absolute or volumerelative, we drop the prefix and rejoin
+             %% the path to create a relative path
+             Relative = filename:join(tl(filename:split(Filename))),
+             error_logger:format("Illegal absolute path: ~ts, converting to
+             ~ts-n",
+                 [Filename, Relative]),
+             relative = filename:pathtype(Relative),
+             Relative
+         end.
+
2377 2390     %% get compressed or stored data
2378 2391     get_z_data(?DEFLATED, In0, FileName, CompSize, Input, Output, Op0, Z) ->
2379 2392         ok = zlib:inflateInit(Z, -?MAX_WBITS),

```


244	247	
245		- %% Verify contents of extracted file.
246		- {ok, Bin} = file:read_file(Name),
247		- true = match_byte_list(X0, binary_to_list(Bin)),
248		+ %% Verify that absolute file was not created
249		+ {error, enoent} = file:read_file(Name),
248	250	
251		+ %% Verify that relative contents of extracted file.
252		+ {ok, Bin} = file:read_file(TempRelName),
253		+ true = match_byte_list(X0, binary_to_list(Bin)),
249	254	
250	255	%% Verify that Unix zip can read it. (if we have a unix zip that is!)
251		- zipinfo_match(Archive, Name),
256		+ zipinfo_match(Archive, RelName),
252	257	
253	258	ok.
254	259	
		<div style="text-align: center;"> ↓ ↑ </div> @@ -1654,6 +1659,50 @@ uid_gid(Config) ->
1654	1659	
1655	1660	ok.
1656	1661	
1662		+ sanitize_filenames(Config) ->
1663		+ RootDir = get_value(pdir, Config),
1664		+ TempDir = filename:join(RootDir, "sanitize_filenames"),
1665		+ ok = file:make_dir(TempDir),
1666		+
1667		+ %% Check that /tmp/absolute does not exist
1668		+ {error, enoent} = file:read_file("/tmp/absolute"),
1669		+
1670		+ %% Create a zip archive /tmp/absolute in it
1671		+ %% This file was created using the command below on Erlang/OTP 28.0
1672		+ %% 1> rr(file), {ok, {_, Bin}} = zip:zip("absolute.zip",
		[{" /tmp/absolute", <<>>, #file_info{ type=regular, mtime={{2000,1,1},{0,0,0}},
		size=0 }}, [memory]], rp(base64:encode(Bin)).
1673		+ AbsZip =
		base64:decode(<<"UESDBAoAAAAAAAAAISgAAAAAAAAAAAAAAAAAANAkAL3RtcC9hYnNvbHV0ZVVU
		BQABcDvt0FBLAQI9AwoAAAAAAAAAISgAAAAAAAAAAAAAAAAAANAkAAAAAAAAAAckAQAAAAAwdG1wL
		2Fic29sdXRlVVQFAAFwNW04UESFBgAAAAABAAEARAAAADQAAAAAA==">>),
1674		+ AbsArchive = filename:join(TempDir, "absolute.zip"),

```

1675 +     ok = file:write_file(AbsArchive, AbsZip),
1676 +
1677 +     {ok, ["tmp/absolute"]} = unzip(Config, AbsArchive, [verbose, {cwd,
TempDir}}]),
1678 +
1679 +     zipinfo_match(AbsArchive, "/tmp/absolute"),
1680 +
1681 +     case un_z64(get_value(unzip, Config)) /= unemzip of
1682 +         true ->
1683 +             {error, enoent} = file:read_file("/tmp/absolute"),
1684 +             {ok, <<>} = file:read_file(filename:join([TempDir, "tmp",
"absolute"]));
1685 +         false ->
1686 +             ok
1687 +     end,
1688 +
1689 +     RelArchive = filename:join(TempDir, "relative.zip"),
1690 +     Relative = filename:join(TempDir, "relative"),
1691 +     ok = file:write_file(Relative, <<>),
1692 +     ?assertMatch({ok, RelArchive}, zip(Config, RelArchive, "", [Relative],
[{:cwd, TempDir}])),
1693 +
1694 +     SanitizedRelative = filename:join(tl(filename:split(Relative))),
1695 +     case un_z64(get_value(unzip, Config)) := unemzip of
1696 +         true ->
1697 +             {ok, [SanitizedRelative]} = unzip(Config, RelArchive, [{:cwd,
TempDir}}]);
1698 +         false ->
1699 +             ok
1700 +     end,
1701 +
1702 +     zipinfo_match(RelArchive, SanitizedRelative),
1703 +
1704 +     ok.
1705 +

```

```
1657 1706 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
1658 1707 %% Generic zip interface
```

```
1659 1708 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```



Comments 0



Please [sign in](#) to comment.