erlang / otp Public

Code Issues 366 Pull requests 151 Actions Projects Wiki Se

Commit 28c5d5a

Mikaka27 committed 19 hours ago Verified

```
Fix root escape vulnerability in SSH_FXP_FSETSTAT
```

master (#11027) · OTP-28.4.3 ••• OTP-26.2.5.20

1 parent 412bff5 commit 28c5d5a

2 files changed +139 -61 lines changed

Top

Filter files…

- lib/ssh
  - src
    - ssh_sftpd.erl
  - test
    - ssh_sftpd_SUITE.erl

2 files changed +139 -61 lines changed

Search within code

lib/ssh/src/ssh_sftpd.erl

```
@@ -54,8 +54,8 @@
54  54          max_files,                  % integer >= 0 max no files sent during READDIR
55  55          options,          % from the subsystem declaration
56  56          handles           % list of open handles
57      -       %% handle is either {<int>, directory, {Path, unread|eof}} or
58      -       %% {<int>, file, {Path, IoDevice}}
    57  +       %% handle is either {<int>, directory, {AbsPath, unread|eof}} or
    58  +       %% {<int>, file, {AbsPath, IoDevice}}
```

```
59  59          }).
60  60
61  61      %%================================================================
        @@ -246,18 +246,18 @@ handle_op(?SSH_FXP_OPENDIR, ReqId,
246 246                           "Not a directory"),
247 247              State1;
248 248          true ->
249     -            add_handle(State1, XF, ReqId, directory, {RelPath,unread})
    249 +            add_handle(State1, XF, ReqId, directory, {AbsPath,unread})
250 250          end;
251 251      handle_op(?SSH_FXP_READDIR, ReqId,
252 252            <<?UINT32(HLen), BinHandle:HLen/binary>>,
253 253            State) ->
254 254          XF = State#state.xf,
255 255          case get_handle(State#state.handles, BinHandle) of
256     -        {_Handle, directory, {_RelPath, eof}} ->
    256 +        {_Handle, directory, {_AbsPath, eof}} ->
257 257              ssh_xfer:xf_send_status(XF, ReqId, ?SSH_FX_EOF),
258 258              State;
259     -        {Handle, directory, {RelPath, Status}} ->
260     -            read_dir(State, XF, ReqId, Handle, RelPath, Status);
    259 +        {Handle, directory, {AbsPath, Status}} ->
    260 +            read_dir(State, XF, ReqId, Handle, AbsPath, Status);
261 261          _ ->
262 262              ssh_xfer:xf_send_status(XF, ReqId, ?SSH_FX_INVALID_HANDLE),
263 263              State
        @@ -293,7 +293,7 @@ handle_op(?SSH_FXP_READ, ReqId, <<?UINT32(HLen),
        BinHandle:HLen/binary,
293 293                            ?UINT64(Offset), ?UINT32(Len)>>,
294 294            State) ->
295 295          case get_handle(State#state.handles, BinHandle) of
296     -        {_Handle, file, {_Path, IoDevice}} ->
    296 +        {_Handle, file, {_AbsPath, IoDevice}} ->
297 297              read_file(ReqId, IoDevice, Offset, Len, State);
298 298          _ ->
299 299              ssh_xfer:xf_send_status(State#state.xf, ReqId,
        @@ -304,7 +304,7 @@ handle_op(?SSH_FXP_WRITE, ReqId,
304 304            <<?UINT32(HLen), BinHandle:HLen/binary, ?UINT64(Offset),
305 305             ?UINT32(Len), Data:Len/binary>>, State) ->
```

```
306  306         case get_handle(State#state.handles, BinHandle) of
307       -      {_Handle, file, {_Path, IoDevice}} ->
     307  +      {_Handle, file, {_AbsPath, IoDevice}} ->
308  308             write_file(ReqId, IoDevice, Offset, Data, State);
309  309         _ ->
310  310             ssh_xfer:xf_send_status(State#state.xf, ReqId,
     ↓ ↑    @@ -347,8 +347,8 @@ handle_op(?SSH_FXP_FSETSTAT, ReqId, <<?UINT32(HLen),
            BinHandle:HLen/binary,
347  347           State0 = #state{handles = Handles}) ->
348  348
349  349         case get_handle(Handles, BinHandle) of
350       -      {_Handle, _Type, {Path,_}} ->
351       -          {Status, State1} = set_stat(Attr, Path, State0),
     350  +      {_Handle, _Type, {AbsPath,_}} ->
     351  +          {Status, State1} = set_stat(Attr, AbsPath, State0),
352  352             send_status(Status, ReqId, State1);
353  353         _ ->
354  354             ssh_xfer:xf_send_status(State0#state.xf, ReqId,
     ↓ ↑    @@ -454,44 +454,42 @@ get_handle(Handles, BinHandle) ->
454  454
455  455     %%% read_dir/5: read directory, send names, and return new state
456  456     read_dir(State0 = #state{file_handler = FileMod, max_files = MaxLength,
            file_state = FS0},
457       -       XF = #ssh_xfer{cm = _CM, channel = _Channel, vsn = Vsn}, ReqId, Handle,
            RelPath, {cache, Files}) ->
458       -      AbsPath = relate_file_name(RelPath, State0),
     457  +       XF = #ssh_xfer{cm = _CM, channel = _Channel, vsn = Vsn}, ReqId, Handle,
            AbsPath, {cache, Files}) ->
459  458         if
460  459         length(Files) > MaxLength ->
461  460             {ToSend, NewCache} = lists:split(MaxLength, Files),
462  461             {NamesAndAttrs, FS1} = get_attrs(AbsPath, ToSend, FileMod, FS0, Vsn),
463  462             ssh_xfer:xf_send_names(XF, ReqId, NamesAndAttrs),
464  463             Handles = lists:keyreplace(Handle, 1,
465  464                                        State0#state.handles,
466       -                                  {Handle, directory, {RelPath,{cache, NewCache}}}),
     465  +                                  {Handle, directory, {AbsPath,{cache, NewCache}}}),
467  466             State0#state{handles = Handles, file_state = FS1};
468  467         true ->
```

```
469  468            {NamesAndAttrs, FS1} = get_attrs(AbsPath, Files, FileMod, FS0, Vsn),
470  469            ssh_xfer:xf_send_names(XF, ReqId, NamesAndAttrs),
471  470            Handles = lists:keyreplace(Handle, 1,
472  471                                       State0#state.handles,
473       -                                     {Handle, directory, {RelPath,eof}}),
     472  +                                     {Handle, directory, {AbsPath,eof}}),
474  473            State0#state{handles = Handles, file_state = FS1}
475  474        end;
476  475    read_dir(State0 = #state{file_handler = FileMod, max_files = MaxLength,
           file_state = FS0},
477       -     XF = #ssh_xfer{cm = _CM, channel = _Channel, vsn = Vsn}, ReqId, Handle,
           RelPath, _Status) ->
478       -    AbsPath = relate_file_name(RelPath, State0),
     476  +     XF = #ssh_xfer{cm = _CM, channel = _Channel, vsn = Vsn}, ReqId, Handle,
           AbsPath, _Status) ->
479  477        {Res, FS1} = FileMod:list_dir(AbsPath, FS0),
480  478        case Res of
481  479        {ok, Files} when MaxLength == 0 orelse MaxLength > length(Files) ->
482  480            {NamesAndAttrs, FS2} = get_attrs(AbsPath, Files, FileMod, FS1, Vsn),
483  481            ssh_xfer:xf_send_names(XF, ReqId, NamesAndAttrs),
484  482            Handles = lists:keyreplace(Handle, 1,
485  483                                       State0#state.handles,
486       -                                     {Handle, directory, {RelPath,eof}}),
     484  +                                     {Handle, directory, {AbsPath,eof}}),
487  485            State0#state{handles = Handles, file_state = FS2};
488  486        {ok, Files} ->
489  487            {ToSend, Cache} = lists:split(MaxLength, Files),
490  488            {NamesAndAttrs, FS2} = get_attrs(AbsPath, ToSend, FileMod, FS1, Vsn),
491  489            ssh_xfer:xf_send_names(XF, ReqId, NamesAndAttrs),
492  490            Handles = lists:keyreplace(Handle, 1,
493  491                                       State0#state.handles,
494       -                                     {Handle, directory, {RelPath,{cache, Cache}}}),
     492  +                                     {Handle, directory, {AbsPath,{cache, Cache}}}),
495  493            State0#state{handles = Handles, file_state = FS2};
496  494        {error, Error} ->
497  495            State1 = State0#state{file_state = FS1},
           @@ -547,13 +545,13 @@ get_long_name(FileName, I) when is_record(I,
           file_info) ->
547  545            I#file_info.mode, I#file_info.uid, I#file_info.gid}).
548  546
```

```
549  547    %%% get_attrs: get stat of each file and return
550       - get_attrs(RelPath, Files, FileMod, FS, Vsn) ->
551       -     get_attrs(RelPath, Files, FileMod, FS, Vsn, []).
     548  + get_attrs(AbsBase, Files, FileMod, FS, Vsn) ->
     549  +     get_attrs(AbsBase, Files, FileMod, FS, Vsn, []).
552  550
553       - get_attrs(_RelPath, [], _FileMod, FS, _Vsn, Acc) ->
     551  + get_attrs(_AbsBase, [], _FileMod, FS, _Vsn, Acc) ->
554  552        {lists:reverse(Acc), FS};
555       - get_attrs(RelPath, [F | Rest], FileMod, FS0, Vsn, Acc) ->
556       -     Path = filename:absname(F, RelPath),
     553  + get_attrs(AbsBase, [F | Rest], FileMod, FS0, Vsn, Acc) ->
     554  +     Path = filename:absname(F, AbsBase),
557  555        case FileMod:read_link_info(Path, FS0) of
558  556        {{ok, Info}, FS1} ->
559  557            Name = if Vsn =< 3 ->
     @@ -563,12 +561,12 @@ get_attrs(RelPath, [F | Rest], FileMod, FS0, Vsn, Acc)
     ->
563  561                  F
564  562             end,
565  563            Attrs = ssh_sftp:info_to_attr(Info),
566       -          get_attrs(RelPath, Rest, FileMod, FS1, Vsn, [{Name, Attrs} | Acc]);
     564  +          get_attrs(AbsBase, Rest, FileMod, FS1, Vsn, [{Name, Attrs} | Acc]);
567  565        {{error, Msg}, FS1} when
568  566                  Msg == enoent ;   % The item has disappeared after reading the
    list of items to check
569  567                  Msg == eacces ->  % You are not allowed to read this
570  568                %% Skip this F and check the remaining Rest
571       -          get_attrs(RelPath, Rest, FileMod, FS1, Vsn, Acc);
     569  +          get_attrs(AbsBase, Rest, FileMod, FS1, Vsn, Acc);
572  570        {Error, FS1} ->
573  571            {Error, FS1}
574  572        end.
     @@ -591,23 +589,25 @@ fstat(Vsn, ReqId, Data, State) when Vsn >= 4->
591  589
592  590    fstat(ReqId, BinHandle, State) ->
593  591        case get_handle(State#state.handles, BinHandle) of
594       -      {_Handle, _Type, {Path, _}} ->
595       -          stat(ReqId, Path, State, read_file_info);
     592  +      {_Handle, _Type, {AbsPath, _}} ->
```

```
                593 +          do_stat(ReqId, AbsPath, State, read_file_info);
596             594         _ ->
597             595             ssh_xfer:xf_send_status(State#state.xf, ReqId,
598             596                         ?SSH_FX_INVALID_HANDLE),
599             597             State
600             598         end.
601             599
602             -  stat(ReqId, RelPath, State0=#state{file_handler=FileMod,
603             -                     file_state=FS0}, F) ->
                600 +  stat(ReqId, RelPath, State0, F) ->
604             601         AbsPath = relate_file_name(RelPath, State0),
                602 +       do_stat(ReqId, AbsPath, State0, F).
                603 +
                604 +  do_stat(ReqId, AbsPath, State0=#state{file_handler=FileMod, file_state=FS0}, F)
                    ->
605             605         XF = State0#state.xf,
606             606         {Res, FS1} = FileMod:F(AbsPath, FS0),
607             607         State1 = State0#state{file_state = FS1},
608             608         case Res of
609             609         {ok, FileInfo} ->
610             -           ssh_xfer:xf_send_attr(XF, ReqId,
                610 +           ssh_xfer:xf_send_attr(XF, ReqId,
611             611                       ssh_sftp:info_to_attr(FileInfo)),
612             612             State1;
613             613         {error, E} ->
                    @@ -715,7 +715,7 @@ do_open(ReqId, State0, Path, Flags) ->
715             715             State1 = State0#state{file_state = FS1},
716             716             case Res of
717             717             {ok, IoDevice} ->
718             -               add_handle(State1, State0#state.xf, ReqId, file, {Path,IoDevice});
                718 +               add_handle(State1, State0#state.xf, ReqId, file,
                    {AbsPath,IoDevice});
719             719             {error, Error} ->
720             720                 ssh_xfer:xf_send_status(State1#state.xf, ReqId,
721             721                             ssh_xfer:encode_erlang_status(Error)),
```

lib/ssh/test/ssh_sftpd_SUITE.erl

```
                @@ -55,11 +55,13 @@
```

```
55  55              ver3_open_flags/1,
56  56              ver3_rename/1,
57  57              ver6_basic/1,
58      -           write_file/1
    58  +           write_file/1,
    59  +           access_attributes_outside_root/1
59  60             ]).
60  61
61  62    -include_lib("common_test/include/ct.hrl").
62  63    -include_lib("kernel/include/file.hrl").
    64  + -include_lib("stdlib/include/assert.hrl").
63  65    -include("ssh_xfer.hrl").
64  66    -include("ssh.hrl").
65  67    -include("ssh_test_lib.hrl").
      ↓↑    @@ -104,7 +106,8 @@ all() ->
104 106         root_with_cwd,
105 107         relative_path,
106 108         open_file_dir_v5,
107     -       open_file_dir_v6].
    109 +       open_file_dir_v6,
    110 +       access_attributes_outside_root].
108 111
109 112   groups() ->
110 113       [].
      ↓↑    @@ -140,6 +143,7 @@ end_per_group(_GroupName, Config) ->
140 143   %%--------------------------------------------------------------------
141 144
142 145   init_per_testcase(TestCase, Config) ->
    146 +     {OsFamily, _} = os:type(),
143 147       ssh:start(),
144 148       prep(Config),
145 149       PrivDir = proplists:get_value(priv_dir, Config),
      ↕     @@ -150,7 +154,7 @@ init_per_testcase(TestCase, Config) ->
150 154              {user_dir, PrivDir},
151 155              {user_passwords,[{?USER, ?PASSWD}]},
152 156              {pwdfun, fun(_,_) -> true end}],
153     -     {ok, Sftpd} = case TestCase of
    157 +     Result = case TestCase of
```

```
154  158                  ver6_basic ->
155  159                  SubSystems = [ssh_sftpd:subsystem_spec([{sftpd_vsn, 6}])],
156  160                  ssh:daemon(0, [{subsystems, SubSystems}|Options]);
     @@ -164,6 +168,14 @@ init_per_testcase(TestCase, Config) ->
164  168                              SubSystems = [ssh_sftpd:subsystem_spec([{root,
     RootDir},
165  169                                                                      {cwd,
     CWD}])],
166  170                              ssh:daemon(0, [{subsystems, SubSystems}|Options]);
     171  +                       access_attributes_outside_root when OsFamily =:= win32 ->
     172  +                           {skip, "Not implemented on windows"};
     173  +                       access_attributes_outside_root ->
     174  +                           Rand = integer_to_list(rand:uniform(1000000)),
     175  +                           RootDir = filename:join("/tmp", Rand),
     176  +                           ok = file:make_dir(RootDir),
     177  +                           SubSystems = [ssh_sftpd:subsystem_spec([{root,
     RootDir}])],
     178  +                           ssh:daemon(0, [{subsystems, SubSystems}|Options]);
167  179                  root_with_cwd ->
168  180                  RootDir = filename:join(PrivDir, root_with_cwd),
169  181                  CWD     = filename:join(RootDir, home),
     @@ -183,42 +195,59 @@ init_per_testcase(TestCase, Config) ->
183  195                  SubSystems = [ssh_sftpd:subsystem_spec([])],
184  196                  ssh:daemon(0, [{subsystems, SubSystems}|Options])
185  197              end,
186       -
187       -      Port = ssh_test_lib:daemon_port(Sftpd),
188  198
189       -      Cm = ssh_test_lib:connect(Port,
190       -                                [{user_dir, ClientUserDir},
191       -                                 {user, ?USER}, {password, ?PASSWD},
192       -                                 {user_interaction, false},
193       -                                 {silently_accept_hosts, true}]),
194       -      {ok, Channel} =
195       -          ssh_connection:session_channel(Cm, ?XFER_WINDOW_SIZE,
196       -                                         ?XFER_PACKET_SIZE, ?SSH_TIMEOUT),
197       -
198       -      success = ssh_connection:subsystem(Cm, Channel, "sftp", ?SSH_TIMEOUT),
     199  +      case Result of
     200  +          {ok, Sftpd} ->
```

```
                201 +             Port = ssh_test_lib:daemon_port(Sftpd),
199             202
200             -     ProtocolVer = case atom_to_list(TestCase) of
201             -               "ver3_" ++ _ ->
202             -               3;
203             -               _ ->
204             -               ?SSH_SFTP_PROTOCOL_VERSION
205             -           end,
                203 +             Cm = ssh_test_lib:connect(Port,
                204 +                                       [{user_dir, ClientUserDir},
                205 +                                        {user, ?USER}, {password, ?PASSWD},
                206 +                                        {user_interaction, false},
                207 +                                        {silently_accept_hosts, true}]),
                208 +             {ok, Channel} =
                209 +                 ssh_connection:session_channel(Cm, ?XFER_WINDOW_SIZE,
                210 +                                                ?XFER_PACKET_SIZE, ?
                    SSH_TIMEOUT),
206             211
207             -     Data = <<?UINT32(ProtocolVer)>> ,
                212 +             success = ssh_connection:subsystem(Cm, Channel, "sftp", ?
                    SSH_TIMEOUT),
208             213
209             -     Size = 1 + size(Data),
                214 +             ProtocolVer = case atom_to_list(TestCase) of
                215 +                               "ver3_" ++ _ ->
                216 +                                   3;
                217 +                               _ ->
                218 +                                   ?SSH_SFTP_PROTOCOL_VERSION
                219 +                           end,
                220 +
                221 +             Data = <<?UINT32(ProtocolVer)>> ,
210             222
211             -     ssh_connection:send(Cm, Channel, << ?UINT32(Size),
212             -                         ?SSH_FXP_INIT, Data/binary >>),
                223 +             Size = 1 + size(Data),
213             224
214             -     {ok, <<?SSH_FXP_VERSION, ?UINT32(Version), _Ext/binary>>, _}
215             -         = reply(Cm, Channel),
                225 +             ssh_connection:send(Cm, Channel, << ?UINT32(Size),
                226 +                                                 ?SSH_FXP_INIT, Data/binary >>),
```

```
216  227
217       -      ct:log("Client: ~p Server ~p~n", [ProtocolVer, Version]),
     228  +              {ok, <<?SSH_FXP_VERSION, ?UINT32(Version), _Ext/binary>>, _}
     229  +                  = reply(Cm, Channel),
218  230
219       -      [{sftp, {Cm, Channel}}, {sftpd, Sftpd }| Config].
     231  +              ct:log("Client: ~p Server ~p~n", [ProtocolVer, Version]),
220  232
     233  +              [{sftp, {Cm, Channel}}, {sftpd, Sftpd }| Config];
     234  +          Other ->
     235  +              Other
     236  +      end.
     237  +
     238  + end_per_testcase(access_attributes_outside_root, Config) ->
     239  +      Sftpd = proplists:get_value(sftpd, Config),
     240  +      {ok, DaemonInfo} = ssh:daemon_info(Sftpd),
     241  +      ssh_cleanup(Config),
     242  +      DaemonOpts = proplists:get_value(options, DaemonInfo),
     243  +      Subsystems = proplists:get_value(subsystems, DaemonOpts),
     244  +      {_, {_, SftpdOpts}} = lists:keyfind("sftp", 1, Subsystems),
     245  +      RootDir = proplists:get_value(root, SftpdOpts),
     246  +      file:del_dir_r(RootDir);
221  247    end_per_testcase(_TestCase, Config) ->
     248  +      ssh_cleanup(Config).
     249  +
     250  + ssh_cleanup(Config) ->
222  251        catch ssh:stop_daemon(proplists:get_value(sftpd, Config)),
223  252        {Cm, Channel} = proplists:get_value(sftp, Config),
224  253        ssh_connection:close(Cm, Channel),
          ↓
          ↑    @@ -792,6 +821,53 @@ open_file_dir_v6(Config) when is_list(Config) ->
792  821                      ?ACE4_READ_DATA  bor ?ACE4_READ_ATTRIBUTES,
793  822                      ?SSH_FXF_OPEN_EXISTING).
794  823
     824  + %%--------------------------------------------------------------------
     825  + access_attributes_outside_root(Config) when is_list(Config) ->
     826  +      Sftpd = proplists:get_value(sftpd, Config),
     827  +      {ok, DaemonInfo} = ssh:daemon_info(Sftpd),
     828  +      DaemonOpts = proplists:get_value(options, DaemonInfo),
     829  +      Subsystems = proplists:get_value(subsystems, DaemonOpts),
```

```
830 +     {_, {_, SftpdOpts}} = lists:keyfind("sftp", 1, Subsystems),
831 +     RootDir = proplists:get_value(root, SftpdOpts),
832 +
833 +     TargetName = "target-" ++ filename:basename(RootDir) ++ ".txt",
834 +     InsideRootFile = filename:join([RootDir, "tmp", TargetName]),
835 +     ok = file:make_dir(filename:dirname(InsideRootFile)),
836 +     ok = file:write_file(InsideRootFile, <<"inside root">>),
837 +     {ok, InsideRootFileInfo} = file:read_file_info(InsideRootFile),
838 +     InsideRootFileMode = InsideRootFileInfo#file_info.mode,
839 +
840 +     OutsideRootFile = filename:join("/tmp", TargetName),
841 +     ok = file:write_file(OutsideRootFile, <<"outside root">>),
842 +     try
843 +         {ok, OutsideRootFileInfo} = file:read_file_info(OutsideRootFile),
844 +         OutsideRootFileMode = OutsideRootFileInfo#file_info.mode,
845 +
846 +         {Cm, Channel} = proplists:get_value(sftp, Config),
847 +         ReqId0 = 0,
848 +         {ok, <<?SSH_FXP_HANDLE, ?UINT32(ReqId0), Handle/binary>>, _} =
849 +             open_file(OutsideRootFile, Cm, Channel, ReqId0,
850 +                       ?ACE4_READ_DATA bor ?ACE4_WRITE_ATTRIBUTES,
851 +                       ?SSH_FXF_OPEN_EXISTING),
852 +
853 +         Attrs = [?uint32(?SSH_FILEXFER_ATTR_PERMISSIONS), ?byte(?
    SSH_FILEXFER_TYPE_REGULAR),
854 +                  ?uint32(not_default_permissions())],
855 +
856 +         ReqId1 = 1,
857 +         {ok, <<?SSH_FXP_STATUS, ?UINT32(ReqId1), ?UINT32(?SSH_FX_OK),
    _/binary>>, _} =
858 +             set_attributes_open_file(Handle, Attrs, Cm, Channel, ReqId1),
859 +
860 +         {ok, NewOutsideRootFileInfo} = file:read_file_info(OutsideRootFile),
861 +         NewOutsideRootFileMode = NewOutsideRootFileInfo#file_info.mode,
862 +         ?assertEqual(OutsideRootFileMode, NewOutsideRootFileMode),
863 +
864 +         {ok, NewInsideRootFileInfo} = file:read_file_info(InsideRootFile),
865 +         NewInsideRootFileMode = NewInsideRootFileInfo#file_info.mode,
866 +         ?assertNotEqual(InsideRootFileMode, NewInsideRootFileMode)
867 +     after
```

```
        868 +          file:delete(OutsideRootFile)
        869 +      end.
        870 +
795     871    %%--------------------------------------------------------------------
796     872    %% Internal functions ------------------------------------------------
797     873    %%--------------------------------------------------------------------
        ...    @@ -806,7 +882,9 @@ prep(Config) ->
806     882        %% Initial config
807     883        DataDir = proplists:get_value(data_dir, Config),
808     884        FileName = filename:join(DataDir, "test.txt"),
809             -      file:copy(FileName, TestFile),
        885 +      {ok, Data0} = file:read_file(FileName),
        886 +      Data = ssh_test_lib:remove_comment(Data0),
        887 +      ok = file:write_file(TestFile, string:chomp(Data)),
810     888        Mode = 8#00400 bor 8#00200 bor 8#00040, % read & write owner, read group
811     889        {ok, FileInfo} = file:read_file_info(TestFile),
812     890        ok = file:write_file_info(TestFile,
        ...
```

Comments 0

Please sign in to comment.