```
From 918bfff86ca8d6d4e4ec5b30994451e0bd74aba9 Mon Sep 17 00:00:00 2001
From: Leon Timmermans <fawaka@gmail.com>
Date: Fri, 23 May 2025 15:40:41 +0200
Subject: [PATCH] CVE-2025-40909: Clone dirhandles without fchdir

This uses fdopendir and dup to dirhandles. This means it won't change
working directory during thread cloning, which prevents race conditions
that can happen if a third thread is active at the same time.
---
 Configure                       |   6 ++
 Cross/config.sh-arm-linux       |   1 +
 Cross/config.sh-arm-linux-n770 |   1 +
 Porting/Glossary                |   5 ++
 Porting/config.sh               |   1 +
 config_h.SH                     |   6 ++
 configure.com                   |   1 +
 plan9/config_sh.sample          |   1 +
 sv.c                            |  91 +++----------------------------
 t/op/threads-dirh.t             | 104 +++-------------------------------
 win32/config.gc                 |   1 +
 win32/config.vc                 |   1 +
 12 files changed, 28 insertions(+), 191 deletions(-)

diff --git a/Configure b/Configure
index 44c12ced4014..7a13249caa96 100755
--- a/Configure
+++ b/Configure
@@ -478,6 +478,7 @@ d_fd_set=''
 d_fds_bits=''
 d_fdclose=''
 d_fdim=''
+d_fdopendir=''
 d_fegetround=''
 d_ffs=''
 d_ffsl=''
@@ -13344,6 +13345,10 @@ esac
 set i_fcntl
 eval $setvar

+: see if fdopendir exists
+set fdopendir d_fdopendir
+eval $inlibc
+
 : see if fork exists
 set fork d_fork
 eval $inlibc
@@ -25052,6 +25057,7 @@ d_flockproto='$d_flockproto'
 d_fma='$d_fma'
 d_fmax='$d_fmax'
 d_fmin='$d_fmin'
+d_fdopendir='$d_fdopendir'
 d_fork='$d_fork'
 d_fp_class='$d_fp_class'
 d_fp_classify='$d_fp_classify'
diff --git a/Cross/config.sh-arm-linux b/Cross/config.sh-arm-linux
index bfa0b00d5f0f..9e056539198b 100644
--- a/Cross/config.sh-arm-linux
+++ b/Cross/config.sh-arm-linux
@@ -212,6 +212,7 @@ d_fd_macros='define'
 d_fd_set='define'
 d_fdclose='undef'
 d_fdim='undef'
+d_fdopendir=undef
 d_fds_bits='undef'
```

```
  d_fegetround='define'
  d_ffs='undef'
diff --git a/Cross/config.sh-arm-linux-n770 b/Cross/config.sh-arm-linux-n770
index 47ad5c37e3fd..365e4c4f9671 100644
--- a/Cross/config.sh-arm-linux-n770
+++ b/Cross/config.sh-arm-linux-n770
@@ -211,6 +211,7 @@ d_fd_macros='define'
  d_fd_set='define'
  d_fdclose='undef'
  d_fdim='undef'
+d_fdopendir=undef
  d_fds_bits='undef'
  d_fegetround='define'
  d_ffs='undef'
diff --git a/Porting/Glossary b/Porting/Glossary
index bb505c653b0b..8b2965ca99c6 100644
--- a/Porting/Glossary
+++ b/Porting/Glossary
@@ -947,6 +947,11 @@ d_fmin (d_fmin.U):
         This variable conditionally defines the HAS_FMIN symbol, which
         indicates to the C program that the fmin() routine is available.

+d_fdopendir (d_fdopendir.U):
+       This variable conditionally defines the HAS_FORK symbol, which
+       indicates that the fdopen routine is available to open a
+       directory descriptor.
+
  d_fork (d_fork.U):
         This variable conditionally defines the HAS_FORK symbol, which
         indicates to the C program that the fork() routine is available.
diff --git a/Porting/config.sh b/Porting/config.sh
index a921f7e1c79a..6231ea0f31ea 100644
--- a/Porting/config.sh
+++ b/Porting/config.sh
@@ -223,6 +223,7 @@ d_fd_macros='define'
  d_fd_set='define'
  d_fdclose='undef'
  d_fdim='define'
+d_fdopendir='define'
  d_fds_bits='define'
  d_fegetround='define'
  d_ffs='define'
diff --git a/config_h.SH b/config_h.SH
index da0f2dbcd7b7..5a0f81cf2011 100755
--- a/config_h.SH
+++ b/config_h.SH
@@ -142,6 +142,12 @@ sed <<!GROK!THIS! >$CONFIG_H -e 's!^#undef\(.*/\)\*!/\*#define\1 \*!' -
e 's!^#un
   */
  #$d_fcntl HAS_FCNTL            /**/

+/* HAS_FDOPENDIR:
+ * This symbol, if defined, indicates that the fdopen routine is
+ * available to open a directory descriptor.
+ */
+#$d_fdopendir HAS_FDOPENDIR            /**/
+
  /* HAS_FGETPOS:
   *     This symbol, if defined, indicates that the fgetpos routine is
   *     available to get the file position indicator, similar to ftell().
diff --git a/configure.com b/configure.com
index 99527c180bfc..7c38711bb85d 100644
--- a/configure.com
+++ b/configure.com
@@ -6010,6 +6010,7 @@ $ WC "d_fd_set='" + d_fd_set + "'"
```

```
   $ WC "d_fd_macros='define'"
   $ WC "d_fdclose='undef'"
   $ WC "d_fdim='" + d_fdim + "'"
  +$ WC "d_fdopendir='undef'"
   $ WC "d_fds_bits='define'"
   $ WC "d_fegetround='undef'"
   $ WC "d_ffs='undef'"
  diff --git a/plan9/config_sh.sample b/plan9/config_sh.sample
  index 636acbdf6db3..246bad954424 100644
  --- a/plan9/config_sh.sample
  +++ b/plan9/config_sh.sample
  @@ -212,6 +212,7 @@ d_fd_macros='undef'
   d_fd_set='undef'
   d_fdclose='undef'
   d_fdim='undef'
  +d_fdopendir=undef
   d_fds_bits='undef'
   d_fegetround='undef'
   d_ffs='undef'
  diff --git a/sv.c b/sv.c
  index ae6d09dea28a..8a005b2d165b 100644
  --- a/sv.c
  +++ b/sv.c
  @@ -14096,15 +14096,6 @@ Perl_dirp_dup(pTHX_ DIR *const dp, CLONE_PARAMS *const param)
   {
       DIR *ret;

  -#if defined(HAS_FCHDIR) && defined(HAS_TELLDIR) && defined(HAS_SEEKDIR)
  -    DIR *pwd;
  -    const Direntry_t *dirent;
  -    char smallbuf[256]; /* XXX MAXPATHLEN, surely? */
  -    char *name = NULL;
  -    STRLEN len = 0;
  -    long pos;
  -#endif
  -
       PERL_UNUSED_CONTEXT;
       PERL_ARGS_ASSERT_DIRP_DUP;

  @@ -14116,89 +14107,13 @@ Perl_dirp_dup(pTHX_ DIR *const dp, CLONE_PARAMS *const param)
       if (ret)
           return ret;

  -#if defined(HAS_FCHDIR) && defined(HAS_TELLDIR) && defined(HAS_SEEKDIR)
  +#ifdef HAS_FDOPENDIR

       PERL_UNUSED_ARG(param);

  -    /* create anew */
  -
  -    /* open the current directory (so we can switch back) */
  -    if (!(pwd = PerlDir_open("."))) return (DIR *)NULL;
  -
  -    /* chdir to our dir handle and open the present working directory */
  -    if (fchdir(my_dirfd(dp)) < 0 || !(ret = PerlDir_open("."))) {
  -        PerlDir_close(pwd);
  -        return (DIR *)NULL;
  -    }
  -    /* Now we should have two dir handles pointing to the same dir. */
  -
  -    /* Be nice to the calling code and chdir back to where we were. */
  -    /* XXX If this fails, then what? */
  -    PERL_UNUSED_RESULT(fchdir(my_dirfd(pwd)));
  +    ret = fdopendir(dup(my_dirfd(dp)));
```

```
-    /* We have no need of the pwd handle any more. */
-    PerlDir_close(pwd);
-
-#ifdef DIRNAMLEN
-# define d_namlen(d) (d)->d_namlen
-#else
-# define d_namlen(d) strlen((d)->d_name)
-#endif
-    /* Iterate once through dp, to get the file name at the current posi-
-       tion. Then step back. */
-    pos = PerlDir_tell(dp);
-    if ((dirent = PerlDir_read(dp))) {
-        len = d_namlen(dirent);
-        if (len > sizeof(dirent->d_name) && sizeof(dirent->d_name) > PTRSIZE) {
-            /* If the len is somehow magically longer than the
-             * maximum length of the directory entry, even though
-             * we could fit it in a buffer, we could not copy it
-             * from the dirent.  Bail out. */
-            PerlDir_close(ret);
-            return (DIR*)NULL;
-        }
-        if (len <= sizeof smallbuf) name = smallbuf;
-        else Newx(name, len, char);
-        Move(dirent->d_name, name, len, char);
-    }
-    PerlDir_seek(dp, pos);
-
-    /* Iterate through the new dir handle, till we find a file with the
-       right name. */
-    if (!dirent) /* just before the end */
-        for(;;) {
-            pos = PerlDir_tell(ret);
-            if (PerlDir_read(ret)) continue; /* not there yet */
-            PerlDir_seek(ret, pos); /* step back */
-            break;
-        }
-    else {
-        const long pos0 = PerlDir_tell(ret);
-        for(;;) {
-            pos = PerlDir_tell(ret);
-            if ((dirent = PerlDir_read(ret))) {
-                if (len == (STRLEN)d_namlen(dirent)
-                    && memEQ(name, dirent->d_name, len)) {
-                    /* found it */
-                    PerlDir_seek(ret, pos); /* step back */
-                    break;
-                }
-                /* else we are not there yet; keep iterating */
-            }
-            else { /* This is not meant to happen. The best we can do is
-                      reset the iterator to the beginning. */
-                PerlDir_seek(ret, pos0);
-                break;
-            }
-        }
-    }
-#undef d_namlen
-
-    if (name && name != smallbuf)
-        Safefree(name);
-#endif
-
-#ifdef WIN32
+#elif defined(WIN32)
     ret = win32_dirp_dup(dp, param);
```

```
 #endif

diff --git a/t/op/threads-dirh.t b/t/op/threads-dirh.t
index bb4bcfc14184..14c399ca19cd 100644
--- a/t/op/threads-dirh.t
+++ b/t/op/threads-dirh.t
@@ -13,16 +13,12 @@ BEGIN {
      skip_all_if_miniperl("no dynamic loading on miniperl, no threads");
      skip_all("runs out of memory on some EBCDIC") if $ENV{PERL_SKIP_BIG_MEM_TESTS};

-     plan(6);
+     plan(1);
 }

 use strict;
 use warnings;
 use threads;
-use threads::shared;
-use File::Path;
-use File::Spec::Functions qw 'updir catdir';
-use Cwd 'getcwd';

 # Basic sanity check: make sure this does not crash
 fresh_perl_is <<'# this is no comment', 'ok', {}, 'crash when duping dirh';
@@ -31,101 +27,3 @@ fresh_perl_is <<'# this is no comment', 'ok', {}, 'crash when duping
dirh';
    async{}->join for 1..2;
    print "ok";
 # this is no comment
-
-my $dir;
-SKIP: {
- skip "telldir or seekdir not defined on this platform", 5
-    if !$Config::Config{d_telldir} || !$Config::Config{d_seekdir};
- my $skip = sub {
-   chdir($dir);
-   chdir updir;
-   skip $_[0], 5
- };
-
- if(!$Config::Config{d_fchdir} && $^O ne "MSWin32") {
-  $::TODO = 'dir handle cloning currently requires fchdir on non-Windows platforms';
- }
-
- my @w :shared; # warnings accumulator
- local $SIG{__WARN__} = sub { push @w, $_[0] };
-
- $dir = catdir getcwd(), "thrext$$" . int rand() * 100000;
-
- rmtree($dir) if -d $dir;
- mkdir($dir);
-
- # Create a dir structure like this:
- #   $dir
- #     |
- #     `- toberead
- #            |
- #            +---- thrit
- #            |
- #            +---- rile
- #            |
- #            `---- zor
-
- chdir($dir);
- mkdir 'toberead';
```

```
 - chdir 'toberead';
 - {open my $fh, ">thrit" or &$skip("Cannot create file thrit")}
 - {open my $fh, ">rile" or &$skip("Cannot create file rile")}
 - {open my $fh, ">zor" or &$skip("Cannot create file zor")}
 - chdir updir;
 -
 - # Then test that dir iterators are cloned correctly.
 -
 - opendir my $toberead, 'toberead';
 - my $start_pos = telldir $toberead;
 - my @first_2 = (scalar readdir $toberead, scalar readdir $toberead);
 - my @from_thread = @{; async { [readdir $toberead ] } ->join };
 - my @from_main = readdir $toberead;
 - is join('-', sort @from_thread), join('-', sort @from_main),
 -     'dir iterator is copied from one thread to another';
 - like
 -   join('-', "", sort(@first_2, @from_thread), ""),
 -   qr/(?<!-rile)-rile-thrit-zor-(?!zor-)/i,
 -  'cloned iterator iterates exactly once over everything not already seen';
 -
 - seekdir $toberead, $start_pos;
 - readdir $toberead for 1 .. @first_2+@from_thread;
 - {
 -  local $::TODO; # This always passes when dir handles are not cloned.
 -  is
 -    async { readdir $toberead // 'undef' } ->join, 'undef',
 -   'cloned dir iterator that points to the end of the directory'
 -  ;
 - }
 -
 - # Make sure the cloning code can handle file names longer than 255 chars
 - SKIP: {
 -  chdir 'toberead';
 -  open my $fh,
 -    ">floccipaucinihilopilification-"
 -   . "pneumonoultramicroscopicsilicovolcanoconiosis-"
 -   . "lopadotemachoselachogaleokranioleipsanodrimypotrimmatosilphiokarabo"
 -   . "melitokatakechymenokichlepikossyphophattoperisteralektryonoptokephal"
 -   . "liokinklopeleiolagoiosiraiobaphetraganopterygon"
 -    or
 -     chdir updir,
 -     skip("OS does not support long file names (and I mean *long*)", 1);
 -  chdir updir;
 -  opendir my $dirh, "toberead";
 -  my $test_name
 -    = "dir iterators can be cloned when the next fn > 255 chars";
 -  while() {
 -   my $pos = telldir $dirh;
 -   my $fn = readdir($dirh);
 -   if(!defined $fn) { fail($test_name); last SKIP; }
 -   if($fn =~ 'lagoio') {
 -    seekdir $dirh, $pos;
 -    last;
 -   }
 -  }
 -  is length async { scalar readdir $dirh } ->join, 258, $test_name;
 - }
 -
 - is scalar @w, 0, 'no warnings during all that' or diag @w;
 - chdir updir;
 -}
 -rmtree($dir);
diff --git a/win32/config.gc b/win32/config.gc
index f8776188c09c..34aa8de6ed75 100644
--- a/win32/config.gc
```

```
+++ b/win32/config.gc
@@ -199,6 +199,7 @@ d_fd_macros='define'
 d_fd_set='define'
 d_fdclose='undef'
 d_fdim='undef'
+d_fdopendir='undef'
 d_fds_bits='define'
 d_fegetround='undef'
 d_ffs='undef'
diff --git a/win32/config.vc b/win32/config.vc
index 619979e22b53..536085fe94e0 100644
--- a/win32/config.vc
+++ b/win32/config.vc
@@ -199,6 +199,7 @@ d_fd_macros='define'
 d_fd_set='define'
 d_fdclose='undef'
 d_fdim='undef'
+d_fdopendir='undef'
 d_fds_bits='define'
 d_fegetround='undef'
 d_ffs='undef'
```