Subject: Who's up for breaking IDL?
Posted by timrobishaw on Sat, 02 Aug 2003 21:10:42 GMT
View Forum Message <> Reply to Message

OK. This one is frustrating. Make a file called resolve_me.pro and make sure it's in a directory that will be in your !path (or add it to your !path once inside IDL). Make it a simple procedure, as simple as could be:

```
pro resolve_me
END
```

Now, open IDL.

```
IDL> resolve_routine, 'resolve_me'
% Compiled module: RESOLVE_ME.
```

Now, exit IDL.

Rename resolve_me.pro to Resolve_Me.pro.

```
>  mv resolve_me.pro Resolve_Me.pro
```

Now, open IDL.

```
IDL> resolve_routine, 'resolve_me'
% Attempt to call undefined procedure/function: 'RESOLVE_ME'.
% Execution halted at: $MAIN$
```

OK. Let's be case sensitive:

```
IDL> resolve_routine, 'Resolve_Me'
% Attempt to call undefined procedure/function: 'RESOLVE_ME'.
% Execution halted at: $MAIN$
```

WTF? Can we even find this routine now?

```
IDL> .run resolve_me
% Error opening file. File: resolve_me
  No such file or directory
```

Well, I guess not. Oh, wait. Let's think case sensitive again:

```
IDL> .run Resolve_Me
% Compiled module: RESOLVE_ME.
```

Huh? Ok. I'll bite. Then why the hell doesn't this work:

IDL> resolve_routine, 'Resolve_Me'

Oh, wait. Now it does. Yeah. I'm confused. What this suggests to me is that you cannot have a file that stores a routine and name it with capital letters. (I could call the actual routine name, inside the file, PRO ReSoLvE_Me... this isn't a case sensitive issue... it's all about how I choose to name the file.) Otherwise, RESOLVE_ROUTINE will choke on it if the routine has not already been compiled. That is dumb. Someone please tell me that I'm right???

-Tim.

Subject: Re: Who's up for breaking IDL?
Posted by David Fanning on Mon, 04 Aug 2003 13:52:11 GMT
View Forum Message <> Reply to Message

R.G. Stockwell writes:

> If the operating system allows case sensitive filenames, then IDL should
> be able to work with them.

Have you thought about how this would work? If your filename has 16 characters in it (and probably includes spaces, too, sigh...), then you would have to check !16 different variations on the filename to make sure you found it. The up side is that you would have time to go get a cup of coffee every time you performed an interaction with the operating system. :-)

Cheers,

David

--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Who's up for breaking IDL?
Posted by David Fanning on Mon, 04 Aug 2003 13:58:59 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> Have you thought about how this would work? If your
> filename has 16 characters in it (and probably
> includes spaces, too, sigh...), then you would
> have to check !16 different variations on the
> filename to make sure you found it.

The alternative, of course, is to forget about
automatic compilation of programs and let everyone
self-compile their programs. This wouldn't be
much of a burden, since about 75% of the programs
I see use this method already. :^)

Cheers,

David

P.S. Am I the only one here this morning!?

--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Who's up for breaking IDL?
Posted by R.G. Stockwell on Mon, 04 Aug 2003 14:12:04 GMT
View Forum Message <> Reply to Message

"David Fanning" <david@dfanning.com> wrote in message
news:MPG.199812f73a1727159896c4@news.frii.com...
> R.G. Stockwell writes:
>
>> If the operating system allows case sensitive filenames, then IDL should
>> be able to work with them.
>
> Have you thought about how this would work? If your
> filename has 16 characters in it (and probably
> includes spaces, too, sigh...), then you would
> have to check !16 different variations on the
> filename to make sure you found it. The up
> side is that you would have time to go get
> a cup of coffee every time you performed an
> interaction with the operating system. :-)
>

> Cheers,
>
> David

Yeah, it'd be a pain. What I'd do if I was IDL is strupcase all the filenames from
the OS then search, and report an error if you ended up with more than one filename.
You would have to do something like to maintain platform independence.

Or, just search for the name with the case given in IDL, I don't see why that
would make any difference. Of course, I would immediately change all my filenames to be capital variations of IDL_PROGRAM.
For instance, if you want my power spectrum program, you would type:
IDL> spectrum = Idl_PrOGRam(timeseries)
whereas my interpolation routine would be
IDL> interped = Idl_PROgRaM(timeseries)

That would work nicely with my policy of making all variable names from different
lengths of the underline character, eg
IDL> __ = 8*4
IDL> _____ = __^2

IDL> help, _____ , __
_____ INT = 1024
__ INT = 32

:)

Cheers,
bob

PS talk about job security

Subject: obfuscated code: was Re: Who's up for breaking IDL?
Posted by R.G. Stockwell on Mon, 04 Aug 2003 15:08:51 GMT
View Forum Message <> Reply to Message

"R.G. Stockwell" <noemail@please.com> wrote in message news:ARtXa.15$%h.12100@news.uswest.net...
>
...> That would work nicely with my policy of making all variable names from
> different
> lengths of the underline character, eg

```
> IDL>  __ = 8*4
> IDL>  _____ = __^2
>
> IDL> help, _____ , __
> _____          INT      =    1024
> __            INT      =      32
>
```

Weeeheee fun. Check out this S-Transform code (save as s_trans_lines.pro):
IDL> .run s_transf_lines

Cheers,
bob

```
; The S transform function
; written by bob stockwell

FUNCTION ________________, _, ___
 __ = 2
 if ___ ne 0.0 then ____ = _/(__*!dPI*___)
 _____ = dblarr(_)
 ______ = dindgen(_)
 _______ = (______ - _/__)^__/(__*____[0]^__)
 ________ = where(_______ lt 25)
 _____(________) = exp(-_______(_________))
 _____ = shift(_____,-_/__)
 return, dcomplex(_____,0)
end


;^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ ^^^^^^^^^^^^^
; the S Transform function
FUNCTION s_trans_lines, _, __

sz = size(_)
___ = n_elements(_)
____ = ___/2
_____ = dcomplexarr(___)
______ = dcomplexarr(___)
_______ = fft(_,-1)

  ________ = 0
  _________ = ____
  __________ = 1
  ___________ = floor((_________ - ________ )/__________)+1
```

```
if n_elements(__) eq 1 then begin
 __ = dblarr(____________)+__
endif

if n_elements(__) ne ____________ then begin
 __ = dblarr(____________)+__(0)
endif

____________________ = where(__ eq 0, ____________________)
if ____________________ gt 0 then begin
 Message,strcompress('Invalid __ Array. __ has a value of zero!')
endif




 _____________ = dcomplexarr(___,____________)
 ________ = shift(________,-_________)
 if _________ eq 0 then begin
  _______ = dblarr(___)
  _______(0) = 1
  _____________(*,0) = fft(________*_______,1)
 endif else begin
  _____________ = double(_________)
    _____________ = __(0) * ___/_____________
  _______ = ___________________(___,_____________)
  _____ = ________ *_______
    _____________(*,0) = fft(_____,1)
 endelse
  for _____________ = 1d,____________-1  do begin
   _____________ = float(_________) + _____________*__________
  _____________ = __(_____________) * ___/_____________
    _______ = ___________________(___,_____________)
    ________ = shift(________,-__________)
    _____ = ________ * _______
    _____________(*,_____________) = fft(_____,1)
   endfor

return, _____________



end ; end of function
```

```
;*****************************************************
; MAIN LEVEL EXAMPLE CODE BELOW
;*****************************************************

randm =randomn(seed,74)
ts = complex(randm,randomn(seed,74)+3*sin(findgen(74)))+1
ts  = fltarr(74)

ti = 2*!dpi*12./74*findgen(74)+0.7
ts = 10*cos(ti)+randomn(Seed,74)
ts(22) = 12

factor = (findgen(38)+1)/38*4
a = s_trans_lines(ts,factor)

!P.multi=[0,1,2]
plot,ts,xtitle='Time'
oplot,imaginary(ts),linestyle=1
 contour,abs(a),/fill,nlevels=14,xtitle='Time',ytitle='Freque ncy (pos)'

end
```

Subject: Re: Who's up for breaking IDL?
Posted by Paul Van Delst[1] on Mon, 04 Aug 2003 15:31:43 GMT
View Forum Message <> Reply to Message

Tim Robishaw wrote:
>
> Oh, wait. Now it does. Yeah. I'm confused. What this suggests to me
> is that you cannot have a file that stores a routine and name it with
> capital letters. (I could call the actual routine name, inside the
> file, PRO ReSoLvE_Me... this isn't a case sensitive issue... it's all
> about how I choose to name the file.) Otherwise, RESOLVE_ROUTINE will
> choke on it if the routine has not already been compiled. That is
> dumb. Someone please tell me that I'm right???

Yep - you're right. Filenames should always be lowercase (on unix systems at
least...dunno about windows.)

It is a bit nutty, I agree, but that's the way it is. I think case sensitivity
in any language is silly. I don't know how C programmers can stand it.

paulv

--
Paul van Delst
CIMSS @ NOAA/NCEP/EMC

Ph: (301)763-8000 x7748
Fax:(301)763-8545

Subject: Re: Who's up for breaking IDL?
Posted by Paul Van Delst[1] on Mon, 04 Aug 2003 15:34:14 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
>
> R.G. Stockwell writes:
>
>> If the operating system allows case sensitive filenames, then IDL should
>> be able to work with them.
>
> Have you thought about how this would work? If your
> filename has 16 characters in it (and probably
> includes spaces, too, sigh...), then you would
> have to check !16 different variations on the
> filename to make sure you found it.

Huh? Why is that? If you allowed case sensitivity in filenames wouldn't you still have to check for just one file since banana.pro and Banana.pro are different files.

paulv

--
Paul van Delst
CIMSS @ NOAA/NCEP/EMC
Ph: (301)763-8000 x7748
Fax:(301)763-8545

Subject: Re: Who's up for breaking IDL?
Posted by Craig Markwardt on Mon, 04 Aug 2003 15:39:21 GMT
View Forum Message <> Reply to Message

"R.G. Stockwell" <noemail@please.com> writes:
> Yeah, it'd be a pain. What I'd do if I was IDL is strupcase all the
> filenames from
> the OS then search, and report an error if you ended up with more than one
> filename.
> You would have to do something like to maintain platform independence.

You are making an assumption of how the IDL path searching ability works. It is quite possible that it does *not* work by retrieving

"all the filenames" from the path. Also, your proposal would require IDL to have within it, filename comparison code, which presumably is exactly what the OS does well, not client programs. Windows is distinct in that it *stores* file names with case sensitivity, but *compares* them without case sensitivity. Unix perserves case for both storage and comparison.

> Or, just search for the name with the case given in IDL, I don't see why
> that
> would make any difference.

The IDL language disregards all upper and lower cases. The IDL astronomy library uses a lot of UPPER CASE, but I use "lower case," and then there are people who use MixedCase. It would be pretty much unacceptable at this date to change the requirement on case sensitivity for IDL programs.

Nit pickily yours,
Craig

--
---------- ----------
Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
---------- ----------

Subject: Re: Who's up for breaking IDL?
Posted by Paul Van Delst[1] on Mon, 04 Aug 2003 15:39:28 GMT
View Forum Message <> Reply to Message

Paul van Delst wrote:
>
> Tim Robishaw wrote:
>>
>> Oh, wait. Now it does. Yeah. I'm confused. What this suggests to me
>> is that you cannot have a file that stores a routine and name it with
>> capital letters. (I could call the actual routine name, inside the
>> file, PRO ReSoLvE_Me... this isn't a case sensitive issue... it's all
>> about how I choose to name the file.) Otherwise, RESOLVE_ROUTINE will
>> choke on it if the routine has not already been compiled. That is
>> dumb. Someone please tell me that I'm right???
>
> Yep - you're right. Filenames should always be lowercase (on unix systems at
> least...dunno about windows.)
>
> It is a bit nutty, I agree, but that's the way it is. I think case sensitivity

> in any language is silly. I don't know how C programmers can stand it.

I just realised after I posted this that it doesn't really make sense since I contradict myself.

Oh well. :o(

What I *meant* to say was I think use of case sensitive filenames is o.k. (IDL doesn't), but not case sensitive variables in actual IDL code.

The downside of allowing case-sensitive filenames is that a simple rule (always lowercase) is gone and users have to rack the brain matter a bit more. In my experience, this can be disasterous (i.e. requiring people to think.... :o) in the short term.

paulv

--
Paul van Delst
CIMSS @ NOAA/NCEP/EMC
Ph: (301)763-8000 x7748
Fax:(301)763-8545

Subject: Re: Who's up for breaking IDL?
Posted by David Fanning on Mon, 04 Aug 2003 15:58:41 GMT
View Forum Message <> Reply to Message

Paul van Delst writes:

> Huh? Why is that? If you allowed case sensitivity in filenames wouldn't you
> still have to check for just one file since banana.pro and Banana.pro are
> different files.

I was talking about *automatic* compilation, not the standard practice of compiling everything you plan to use ahead of time so there are absolutely no surprises and so you have something to do that impresses the boss. :-)

Cheers,

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/

Subject: Re: Who's up for breaking IDL?
Posted by R.G. Stockwell on Mon, 04 Aug 2003 16:00:46 GMT
View Forum Message <> Reply to Message

"Craig Markwardt" <craigmnet@cow.physics.wisc.edu> wrote in message
news:onk79tph3q.fsf@cow.physics.wisc.edu...
>
> "R.G. Stockwell" <noemail@please.com> writes:
>> Yeah, it'd be a pain. What I'd do if I was IDL is strupcase all the
>> filenames from
>> the OS then search, and report an error if you ended up with more than
one
>> filename.
>> You would have to do something like to maintain platform independence.
>
> You are making an assumption of how the IDL path searching ability
> works. It is quite possible that it does *not* work by retrieving
> "all the filenames" from the path. Also, your proposal would require
> IDL to have within it, filename comparison code, which presumably is
> exactly what the OS does well, not client programs.

Yes, you are right. IDL does not store names, only paths. IDL does
build up a cache though. According to the help
"After trying to open the file in the user's current working
directory, IDL will attempt to open the file in each of the
directories given by the !PATH system variable, in the
order specified by !PATH. The search stops with the
first directory containing a file with the desired name.
IDL now maintains an in-memory cache of the .pro and .sav files
located in directories referenced via the !PATH system variable.
This path cache is built automatically during normal operation, as
IDL searches the directories specified by !PATH to locate the
code for IDL routines required at runtime. "

Having read this, I don't see why the IDL code of the OP does not work.
It should try to open the file given in the filename string, and that fact
that
it fails (for the Resolve_Me.pro example) means that somewhere IDL
has changed that string. Sounds like a bug.

Windows is
> distinct in that it *stores* file names with case sensitivity, but
> *compares* them without case sensitivity. Unix perserves case for
> both storage and comparison.
>

>> Or, just search for the name with the case given in IDL, I don't see why
>> that
>> would make any difference.
>
> The IDL language disregards all upper and lower cases. The IDL
> astronomy library uses a lot of UPPER CASE, but I use "lower case,"
> and then there are people who use MixedCase. It would be pretty much
> unacceptable at this date to change the requirement on case
> sensitivity for IDL programs.

I concur, certainly for all idl code. However, I think literal string, or
string
values should be case sensitive (and they are). So in the original posters
code, the
IDL> resolve_routine, 'Resolve_Me'
% Attempt to call undefined procedure/function: 'RESOLVE_ME'.
% Execution halted at: $MAIN$

should have worked, since it should have tried to open the correct file.
Of couse, this leads to the problem of having two differently cased
filenames in
the same folder, but I don't think that problem is any different that having
two
files of the same name in different folders. That is just a programers
error
if the programmer lets it happen.

Cheers,
bob

Subject: Re: Who's up for breaking IDL?
Posted by fburton on Mon, 04 Aug 2003 16:05:17 GMT
View Forum Message <> Reply to Message

In article <3F2E7C5F.731392A1@noaa.gov>,
Paul van Delst <paul.vandelst@ssec.wisc.edu> wrote:
> Yep - you're right. Filenames should always be lowercase (on unix systems at
> least...dunno about windows.)

Surely the most "humane" default is to preserve case in filenames
(and program identifiers), but have case-insensitive matching. The
arguments for case sensitivity in dealing with natural language
text don't apply to programming languages or interactions with the
filesystem, do they?

Francis

Subject: Re: Who's up for breaking IDL?
Posted by R.G. Stockwell on Mon, 04 Aug 2003 16:06:14 GMT
View Forum Message <> Reply to Message

"David Fanning" <david@dfanning.com> wrote in message
news:MPG.199830a08e6606d89896c6@news.frii.com...
> Paul van Delst writes:
>
>> Huh? Why is that? If you allowed case sensitivity in filenames wouldn't
you
>> still have to check for just one file since banana.pro and Banana.pro
are
>> different files.
>
> I was talking about *automatic* compilation, not
> the standard practice of compiling everything you
> plan to use ahead of time so there are absolutely
> no surprises and so you have something to do that
> impresses the boss. :-)

I agree completely, automatic compliation should only look
for (the arbitrarily decided) lower case. There is no way for IDL to
know what the case "should be" based on the case insensitive code in
IDL.
However, I think the example shown by the original poster, which
used a literal string of the correct case, should have worked, and
the fact that it didn't is a bug.

IDL> resolve_routine, 'Resolve_Me'
% Attempt to call undefined procedure/function: 'RESOLVE_ME'.
% Execution halted at: $MAIN$

It seems like IDL took a string constant 'Resolve_Me' and changed
it to a different string constant 'RESOLVE_ME'.

Cheers,
bob

Subject: Re: Who's up for breaking IDL?
Posted by David Fanning on Mon, 04 Aug 2003 16:38:47 GMT
View Forum Message <> Reply to Message

R.G. Stockwell writes:

> However, I think the example shown by the original poster, which
> used a literal string of the correct case, should have worked, and

> the fact that it didn't is a bug.
>
> IDL> resolve_routine, 'Resolve_Me'
> % Attempt to call undefined procedure/function: 'RESOLVE_ME'.
> % Execution halted at: $MAIN$
>
> It seems like IDL took a string constant 'Resolve_Me' and changed
> it to a different string constant 'RESOLVE_ME'.

Bob, I think maybe you are missing the intention
of the RESOLVE_ALL and it companion RESOLVE_ROUTINE
(which appears to do all the work. I'm reminded of
Don Quiote and his faithful servant Sancho). These
were designed so the user could *automatically* resolve
all the unresolved references in a piece of code
so you could create a save file.

I think the designers figured that anybody who
simply wanted to *compile* a routine would probably
use the aptly named COMPILE command to do so. (Or,
the completely misnamed RUN command, but that's
another story.) I am 100% sure (although I have
no case sensitive operating system here to check)
that the .Compile command would work in the manner
the user expected from the RESOVE_ROUTINE command. :-)

Cheers,

David
--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Who's up for breaking IDL?
Posted by James Kuyper on Mon, 04 Aug 2003 16:46:20 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
...
> another story.) I am 100% sure (although I have
> no case sensitive operating system here to check)
> that the .Compile command would work in the manner
> the user expected from the RESOVE_ROUTINE command. :-)

Regardless of what the user expects? That's impressive! ;-)

Subject: Re: Who's up for breaking IDL?
Posted by R.G. Stockwell on Mon, 04 Aug 2003 17:06:04 GMT
View Forum Message <> Reply to Message

"David Fanning" <david@dfanning.com> wrote in message
news:MPG.19983a015914c4269896c7@news.frii.com...
> R.G. Stockwell writes:
>
>> However, I think the example shown by the original poster, which
>> used a literal string of the correct case, should have worked, and
>> the fact that it didn't is a bug.
>>
>> IDL> resolve_routine, 'Resolve_Me'
>> % Attempt to call undefined procedure/function: 'RESOLVE_ME'.
>> % Execution halted at: $MAIN$
>>
>> It seems like IDL took a string constant 'Resolve_Me' and changed
>> it to a different string constant 'RESOLVE_ME'.
>
> Bob, I think maybe you are missing the intention
> of the RESOLVE_ALL and it companion RESOLVE_ROUTINE

Yes, I am missing the intention.

> I think the designers figured that anybody who
> simply wanted to *compile* a routine would probably
> use the aptly named COMPILE command to do so. (Or,
> the completely misnamed RUN command, but that's
> another story.) I am 100% sure (although I have
> no case sensitive operating system here to check)
> that the .Compile command would work in the manner
> the user expected from the RESOVE_ROUTINE command. :-)

Yes, you are right. The following fails
IDL> .comp resolve_me
while the following does indeed work
IDL> .comp Resolve_Me

I hereby withdraw to the comfortable confines of under my rock.

Cheers,
bob
