
Subject: IDL killer

Posted by [marc schellens\[1\]](#) on Wed, 19 Jan 2005 13:23:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

```
IDL> print,!version
{ x86 linux unix linux 6.0 Jun 27 2003    32    64}
IDL> b=bindgen(64)
IDL> byteorder,b,/sswap,/l64swap
Floating exception
```

Cheers,
marc

Subject: Re: IDL killer

Posted by [Ben Panter](#) on Wed, 19 Jan 2005 13:35:18 GMT

[View Forum Message](#) <> [Reply to Message](#)

Marc Schellens wrote:

```
> IDL> print,!version
> { x86 linux unix linux 6.0 Jun 27 2003    32    64}
> IDL> b=bindgen(64)
> IDL> byteorder,b,/sswap,/l64swap
> Floating exception
```

I get the same with

```
{ x86 linux unix linux 6.1 Jul 14 2004    32    64}
```

but if I don't use the /l64swap option everything is fine. Is it forcing IDL to use a 64bit command on a 32bit os?

Ben

--

```
-----
Ben Panter, Garching, Germany
email via www.benpanter.co.uk
```

Subject: Re: IDL killer

Posted by [Paolo Grigis](#) on Wed, 19 Jan 2005 14:54:29 GMT

[View Forum Message](#) <> [Reply to Message](#)

Well, to be fair, in the docs of byteorder it says:

[...] If the type of byte swapping is not specified via **one** of the

keywords below [...]
(my emphasis)

This could be understood as an implicit warning not to be so bold to actually try to use *two* keywords at the same time (but of course the penalty of death for your IDL session does indeed seem a bit hard;-)

Cheers,
Paolo

Marc Schellens wrote:

```
> IDL> print,!version
> { x86 linux unix linux 6.0 Jun 27 2003    32    64}
> IDL> b=bindgen(64)
> IDL> byteorder,b,/sswap,/l64swap
> Floating exception
>
>
> Cheers,
> marc
>
```

Subject: Re: IDL killer
Posted by [R.G.Stockwell](#) on Wed, 19 Jan 2005 17:51:27 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Marc Schellens" <m_schellens@hotmail.com> wrote in message
news:cslmtq\$2dho\$1@news.riken.go.jp...

```
> IDL> print,!version
> { x86 linux unix linux 6.0 Jun 27 2003    32    64}
> IDL> b=bindgen(64)
> IDL> byteorder,b,/sswap,/l64swap
> Floating exception
>
>
> Cheers,
> marc
```

Ditto on
IDL> print,!version
{ x86 Win32 Windows Microsoft Windows 6.1.1 Oct 11 2004 32 64}

Error message:
The exception Integer division by zero.

Cheers,

bob

Subject: Re: IDL killer

Posted by [Robert Barnett](#) on Wed, 19 Jan 2005 22:35:32 GMT

[View Forum Message](#) <> [Reply to Message](#)

Perhaps we could have a competition to find as many one line killers as possible.

Here's one, but it's only a 13 liner.

=====

I've managed to find an error specific to IDL on Solaris

"SunOS adac9 5.9 Generic sun4u sparc SUNW,Ultra-5_10"

"SunOS adac7 5.9 Generic sun4u sparc SUNW,Ultra-1"

The error occurs in both IDL 6.0 and IDL 6.1 but not in IDL 6.0 on Windows.

The error occurs when I run the DICOMTEST__TEST1 procedure from the attached code.

Running this program results in a core dump of the IDL process.

The problem is related to when I call IDLffDICOM::read as an inherited method.

If the filename points to a file which does not exist then this will cause a segmentation fault (Most likely in the dicom DLM)

This error can be easily avoided by running file_test before IDLffDICOM::read. Thus, I have managed to avoid this problem.

--

nrb@
Robbie Barnett
imag

Research Assistant
wsahs
Nuclear Medicine & Ultrasound
nsw
Westmead Hospital
gov
Sydney Australia
au
+61 2 9845 7223

```
; Define the dicomtest class which extends the IDLffDICOM class
pro dicomtest__define
struct = {dicomtest, image: ptr_new(), INHERITS IDLffDICOM }
end
```

```
; Define the dicomtest init method
function dicomtest::init
; Return 0 if the superclass did not init
if (NOT (self -> IDLffDICOM::init())) then return, 0
return, 1
end
```

```
; This procedure starts and runs the dicomtest wrapper object
pro dicomtest__test1
filename = 'dummyfilename'
help, file_test(filename)
dicom_obj = obj_new('dicomtest')
if (dicom_obj -> read(filename)) then print, 'ok'
end
```

File Attachments

1) [dicomtest__define.pro](#), downloaded 155 times

Subject: Re: IDL killer

Posted by [JD Smith](#) on Wed, 19 Jan 2005 22:54:48 GMT

[View Forum Message](#) <> [Reply to Message](#)

On Thu, 20 Jan 2005 09:35:32 +1100, Robert Barnett wrote:

```
>
> Perhaps we could have a competition to find as many one line killers as
> possible.
>
```

This one's a real puzzler:

```
r=execute(string(reverse([5b,1b,6b,0b]*3b+(bindgen(4) mod 3b gt 0b)+101b)))
```

JD

Subject: Re: IDL killer

Posted by [David Fanning](#) on Wed, 19 Jan 2005 23:28:23 GMT

[View Forum Message](#) <> [Reply to Message](#)

JD Smith writes:

> This one's a real puzzler:

>

> r=execute(string(reverse([5b,1b,6b,0b]*3b+(bindgen(4) mod 3b gt 0b)+101b)))

J.D, do you get enough sleep at night!?

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: IDL killer

Posted by [R.G.Stockwell](#) on Thu, 20 Jan 2005 00:10:20 GMT

[View Forum Message](#) <> [Reply to Message](#)

"JD Smith" <jdsmith@as.arizona.edu> wrote in message
news:pan.2005.01.19.22.54.48.408558@as.arizona.edu...

> This one's a real puzzler:

>

> r=execute(string(reverse([5b,1b,6b,0b]*3b+(bindgen(4) mod 3b gt
> 0b)+101b)))

>

> JD

lol

Subject: Re: IDL killer

Posted by [R.G.Stockwell](#) on Thu, 20 Jan 2005 00:11:05 GMT

[View Forum Message](#) <> [Reply to Message](#)

"JD Smith" <jdsmith@as.arizona.edu> wrote in message
news:pan.2005.01.19.22.54.48.408558@as.arizona.edu...

> This one's a real puzzler:

>

> r=execute(string(reverse([5b,1b,6b,0b]*3b+(bindgen(4) mod 3b gt
> 0b)+101b)))

>

> JD

lol

Subject: Re: IDL killer

Posted by [R.G.Stockwell](#) on Thu, 20 Jan 2005 00:25:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

"R.G. Stockwell" <noemail@please.com> wrote in message
news:vGCHd.44\$es.5980@news.uswest.net...

> "JD Smith" <jdsmith@as.arizona.edu> wrote in message

> news:pan.2005.01.19.22.54.48.408558@as.arizona.edu...

>> This one's a real puzzler:

>>

>> r=execute(string(reverse([5b,1b,6b,0b]*3b+(bindgen(4) mod 3b gt
>> 0b)+101b)))

>>

>> JD

I mean

string(reverse([0b,1b,0b]*2b+(bindgen(3) mod 2b gt 0b)+108b))

Subject: Re: IDL killer

Posted by [wmconnolley](#) on Fri, 21 Jan 2005 19:22:55 GMT

[View Forum Message](#) <> [Reply to Message](#)

Ben Panter <me@privacy.net> wrote:

> Marc Schellens wrote:

>> IDL> print,!version

>> { x86 linux unix linux 6.0 Jun 27 2003 32 64}

>> IDL> b=bindgen(64)

>> IDL> byteorder,b,/sswap,/l64swap

>> Floating exception

> I get the same with

```
> { x86 linux unix linux 6.1 Jul 14 2004 32 64}
```

On a Sun (ditto alpha) it survives but doesn't like the op:

```
wmc> b=bindgen(64)
```

```
wmc> byteorder,b,/sswap,/l64swap
```

```
% BYTEORDER: Operand's size must be a multiple of swap datum size: B.
```

```
% Execution halted at: $MAIN$
```

```
wmc> print,!version
```

```
{ sparc sunos unix Solaris 6.0.3 Feb 26 2004 64 64}
```

-W.

--

William M Connolley | wmc@bas.ac.uk | <http://www.antarctica.ac.uk/met/wmc/>
Climate Modeller, British Antarctic Survey | Disclaimer: I speak for myself
I'm a .signature virus! copy me into your .signature file & help me spread!

Subject: Re: IDL killer

Posted by [Haje Korth](#) on Mon, 14 Mar 2005 13:06:43 GMT

[View Forum Message](#) <> [Reply to Message](#)

Marc,

ah, time for Easter eggs. I just checked under Windows and indeed IDL blows up!

Haje

<m_schellens@hotmail.com> wrote in message

news:1110793473.736235.57300@f14g2000cwb.googlegroups.com...

```
> ++(((a=1))=3)
```

```
>
```

```
> Cheers,
```

```
> marc
```

```
>
```

Subject: Re: IDL killer

Posted by [David Fanning](#) on Mon, 14 Mar 2005 13:14:08 GMT

[View Forum Message](#) <> [Reply to Message](#)

m_schellens@hotmail.com writes:

```
> ++(((a=1))=3)
```

I think it must be a general rule by now that any time

you fail to have a LHS of an expression in IDL (by, for example, putting the RHS in parentheses and thereby making it a temporary variable) there is an excellent chance you will crash IDL. Just on the face of it, I would guess there must be an almost unlimited number of these expressions.

I'm not so sure this is a bug as much as it is a peek into the underlying structure of the IDL code. In any case, I'm not surprised by it any longer.

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: IDL killer

Posted by [marc schellens\[1\]](#) on Mon, 14 Mar 2005 15:56:28 GMT

[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

> m_schellens@hotmail.com writes:

>

>> ++(((a=1))=3)

>

> I think it must be a general rule by now that any time
> you fail to have a LHS of an expression in IDL (by, for
> example, putting the RHS in parentheses and thereby
> making it a temporary variable) there is an excellent
> chance you will crash IDL. Just on the face of it,
> I would guess there must be an almost unlimited
> number of these expressions.

Well, I am sure it also crashes with other values,
what makes it almost infinite :-)

But here we DON'T fail to have a LHS!

```
IDL> help,(((a=1))=3)
```

```
A      INT      =      3
```

> I'm not so sure this is a bug as much as it is a peek
> into the underlying structure of the IDL code. In any

> case, I'm not surprised by it any longer.

Even though I would call it a bug, there is hardly a case where such a construct would make sense (please post whoever can think of one).

Generally assignment expressions shouldn't be allowed on LHS of expressions.

Ah, did I mention that GDL (<http://gnudatalanguage.sf.net>) refuses them, but does not crash :-)

Cheers,
marc

Subject: Re: IDL killer
Posted by [David Fanning](#) on Mon, 14 Mar 2005 16:15:58 GMT
[View Forum Message](#) <> [Reply to Message](#)

m_schellens@hotmail.com writes:

> Ah, did I mention that GDL (<http://gnudatalanguage.sf.net>)
> refuses them, but does not crash :-)

Perhaps the gentlemen building it have the advantage of hindsight. :-)

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Subject: Re: IDL killer
Posted by [Y.T.](#) on Mon, 21 Mar 2005 21:33:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

Ah, the benefits of living under a rock:

[12:37] YT [suttungr:~] > idl
IDL Version 5.5a (linux x86). (c) 2001, Research Systems, Inc.
Installation number: ****-.*.

Licensed for use by: MAD SCIENTISTS LTD.

```
IDL> ++(((a=1))=3)
```

```
++(((a=1))=3)
```

```
^
```

```
% Syntax error.
```

```
IDL> help,(((a=1))=3)
```

```
% Expression must be named variable in this context: <INT (3)>.
```

```
% Execution halted at: $MAIN$
```

Sooo... what exactly are the reasons to "up"grade to V6?

:)

cordially

Y.T.

--

Remove YourClothes before you email me.

Subject: Re: IDL killer

Posted by [David Fanning](#) on Mon, 21 Mar 2005 21:38:36 GMT

[View Forum Message](#) <> [Reply to Message](#)

Y.T. writes:

> Ah, the benefits of living under a rock:

>

>

> [12:37] YT [suttungr:~] > idl

> IDL Version 5.5a (linux x86). (c) 2001, Research Systems, Inc.

> Installation number: ****-*

> Licensed for use by: MAD SCIENTISTS LTD.

>

> IDL> ++(((a=1))=3)

>

> ++(((a=1))=3)

> ^

> % Syntax error.

> IDL> help,(((a=1))=3)

> % Expression must be named variable in this context: <INT ((3)>.
> % Execution halted at: \$MAIN\$
>
>
> Sooo... what exactly are the reasons to "up"grade to V6?

Incomprehensible object tools?

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
