Ruby master - Bug #11439
Win32 Registry corruption when writing REG_MULTI_SZ values
08/13/2015 07:37 AM - Iristyle (Ethan Brown)

Status: Closed
Priority: Normal
Assignee: 
Target version: 
Backport: 2.0.0: UNKNOWN, 2.1: REQUIRED, 2.2: DONE

Description:
A simple reproduction of the problem that breaks on first corrupted write:

```ruby
require 'win32/registry'

root = Win32::Registry::HKEY_CURRENT_USER
root.create('SOFTWARE\rubyfail') do |reg|
  1000.times do |i|
    value = ('a'..'z').to_a.shuffle[0,20].join
    reg.write('foo', Win32::Registry::REG_MULTI_SZ, [value])
    if reg['foo'] != [value]
      puts "iteration #{i} - expected value #{value}, got #{reg['foo']}"
      break
    end
  end
end
```

A message should be emitted like

```
iteration 4 - expected value notemdubgxlryjihcaf, got ["notemdubgxlryjihcaf", "\u6B00"]
```

Looking at the code, there are issues with how Ruby is handling strings during Registry writes. 

- For **REG_MULTI_SZ**, Ruby is encoding each string in UTF-16LE and joining them with a WCHAR_NUL, but is only appending a single WCHAR_NUL, when it should be appending two. The last 4 bytes of the buffer used for REG_MULTI_SZ should always be [0, 0, 0, 0]. Because the second NULL is missing, whatever data happens to be stored in the final 2 bytes is being written to the registry in the above example.

- Additionally, for **REG_SZ** or **REG_EXPAND_SZ**, Ruby is only encoding in UTF-16LE, but is not appending the appropriate NULL terminator prior to passing the string to SetValue. Simply calling data.encode(WCHAR) does not NULL terminate a string.

My suggestion is to rewrite the method like:

```ruby
def write(name, type, data)
  case type
  when REG_SZ, REG_EXPAND_SZ
    data = data.encode(WCHAR) + WCHAR_NUL
  when REG_MULTI_SZ
    data = data.to_a.map { |s| s.encode(WCHAR) }.join(WCHAR_NUL) << WCHAR_NUL << WCHAR_NUL
  when REG_BINARY
    data = data.to_s
  when REG_DWORD
    data = API.packdw(data.to_i)
  when REG_DWORD_BIG_ENDIAN
    data = [data.to_i].pack("N")
  when REG_QWORD
    data = API.packqw(data.to_i)
  else
```

05/17/2020
raise TypeError, "Unsupported type #{type}"
end
API.SetValue(@hkey, name, type, data, data.bytesize)
end

Note that termssize has been removed as it is unnecessary if data has the appropriate bytes already.

Furthermore, there is also a bug in the SetValue implementation for when the default size for a string is set, when no length is passed:

Since UTF-16LE strings are always an even number of bytes, it would only make sense to size |= data.size + 2 rather than size |=
data.size + 1. That of course assumes that the incoming string is already NULL terminated, which it may not be. My suggestion
would be to remove this line altogether unless there is intent to verify the last byte.

Please see the MSDN Documentation for SetRegValueEx for the specifics. The most important parts are:

lpData [in]
The data to be stored.
For string-based types, such as REG_SZ, the string must be null-terminated. With the REG_MULTI_SZ
data type, the string must be terminated with two null characters. String-literal values must be f
ormatted using a backslash preceded by another backslash as an escape character. For example, spec
ify "C:\\mydir\\myfile" to store the string "C:\mydir\myfile".
Note lpData indicating a null value is valid, however, if this is the case, cbData must be set to
'0'.

cbData [in]
The size of the information pointed to by the lpData parameter, in bytes. If the data is of type R
EG_SZ, REG_EXPAND_SZ, or REG_MULTI_SZ, cbData must include the size of the terminating null charac
ter or characters.

Associated revisions
Revision 51336eda - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: terminator size

  • ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]
git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@51575 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 51575 - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: terminator size

  • ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

Revision 51575 - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: terminator size

  • ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

Revision 51575 - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: terminator size

  • ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

Revision 51575 - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: terminator size

  • ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

Revision 51575 - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: terminator size

  • ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

Revision 51575 - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: terminator size

  • ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

Revision 51575 - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: terminator size

  • ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

Revision 51575 - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: terminator size

  • ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

05/17/2020
ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

Revision e5196e24 - 08/15/2015 01:51 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: size in bytes

ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]
git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@51584 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 51584 - 08/15/2015 01:51 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: size in bytes

ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]

Revision 51584 - 08/15/2015 01:51 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: size in bytes

ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]

Revision 51584 - 08/15/2015 01:51 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: size in bytes

ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]

Revision 51584 - 08/15/2015 01:51 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: size in bytes

ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]

Revision 51584 - 08/15/2015 01:51 AM - nobu (Nobuyoshi Nakada)
win32/registry.rb: size in bytes

ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]

Revision 776c8182 - 08/17/2015 02:02 PM - nagachika (Tomoyuki Chikanaga)
merge revision(s) 51575,51584: [Backport #11439]

* ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

* ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_2@51618 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 51618 - 08/17/2015 02:02 PM - nagachika (Tomoyuki Chikanaga)
merge revision(s) 51575,51584: [Backport #11439]

* ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

* ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]

Revision 911ade36 - 08/17/2015 02:27 PM - usa (Usaku NAKAMURA)
merge revision(s) 51575,51584: [Backport #11439]

* ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

* ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/branches/ruby_2_1@51620 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 51620 - 08/17/2015 02:27 PM - usa (Usaku NAKAMURA)
merge revision(s) 51575,51584: [Backport #11439]

- ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

- ext/win32/lib/win32/registry.rb (API#SetValue): data size should be in bytes, not in chars. [ruby-core:70365] [Bug #11439]

History

#1 - 08/13/2015 03:31 PM - nobu (Nobuyoshi Nakada)

- Status changed from Open to Feedback

It seems working with the trunk.

#2 - 08/13/2015 04:02 PM - Iristyle (Ethan Brown)

Since this is a corruption that depends on the memory state of the environment (whether or not it's zeroed), you might want to increase the number of iterations, or change the random string length in my reproduction case.

I can reproduce this very easily on Windows 2008R2 x64 (running inside VMWare fusion) on the stated Ruby version.

The errant code is still present in trunk at https://github.com/ruby/ruby/blob/trunk/ext/win32/lib/win32/registry.rb#L738-L759

#3 - 08/13/2015 04:05 PM - cremno (cremno phobia)

This code relies on the fact that Ruby strings are implicitly null-terminated. Otherwise you'd always have to either create a copy or modify the original when calling functions that expect a null-terminated string.

Since 2.1 that's even the case for non-8-bit-encodings. E.g. for UTF-16 the implicit null terminator should be two null bytes (or one null character). But here it isn't: only one null byte instead of two are written in 2.1 (see the emitted message). So at least r49408 needs to be backported (Array#join and lot of other methods internally call attrbuf_cat()). The 2.2 backport is r49644. It contains other related fixes that likely need to be backported too.

#4 - 08/13/2015 04:54 PM - Iristyle (Ethan Brown)

Ah I did not realize that strings all implicitly null-terminated. Thanks for that explanation cremno phobia - that makes sense. That would also explain why I could repro only against REG_MULTI_SZ but not against REG_SZ.

I feel like the implementation is a little sneaky given it relies on that internal implementation detail (so from the outside looking in, it feels a bit magical to increase the buffer size passed to the Windows API they way it's being done). But that's probably well understood by Ruby maintainers.

I still believe the logic in SetValue should be size ||= data.size + WCHAR_SIZE given the data has been encoded as WCHAR there... but I'm not sure that any callers are calling SetValue without passing a size anyhow?

#5 - 08/14/2015 09:42 AM - nobu (Nobuyoshi Nakada)

- Status changed from Feedback to Closed

Applied in changeset r51575.

win32/registry.rb: terminator size

  - ext/win32/lib/win32/registry.rb (API#SetValue): add terminator size, not 1 byte. [ruby-core:70365] [Bug #11439]

#6 - 08/15/2015 01:03 AM - cremno (cremno phobia)

Ethan Brown wrote:

> I still believe the logic in SetValue should be size ||= data.size + WCHAR_SIZE given the data has been encoded as WCHAR there...

That's only half correct. It actually has to be size ||= data.bytesize + WCHAR_SIZE. The cbData argument must be the size in bytes but String#size returns the length in characters.

#7 - 08/17/2015 02:03 PM - nagachika (Tomoyuki Chikanaga)

- Backport changed from 2.0.0: UNKNOWN, 2.1: REQUIRED, 2.2: REQUIRED to 2.0.0: UNKNOWN, 2.1: REQUIRED, 2.2: DONE

Backported into ruby_2_2 branch at r51618.

#8 - 08/17/2015 02:28 PM - usa (Usaku NAKAMURA)
ruby_2_1 r51620 merged revision(s) 51575,51584.

#9 - 08/17/2015 04:08 PM - cremno (cremno phobia)
Usaku NAKAMURA wrote:

ruby_2_1 r51620 merged revision(s) 51575,51584.

This issue is about two different bugs. r49405 to r49408 need to be backported too.

#10 - 08/17/2015 05:56 PM - usa (Usaku NAKAMURA)

- Backport changed from 2.0.0: UNKNOWN, 2.1: DONE, 2.2: DONE to 2.0.0: UNKNOWN, 2.1: REQUIRED, 2.2: DONE

Oh, thanks for telling!