Ruby - Bug #4325

[ext/openssl] Encoding of subclasses fails when it shouldn't

01/26/2011 09:46 AM - MartinBosslet (Martin Bosslet)

Status:	Closed		
Priority:	Normal		
Assignee:	MartinBosslet (Martin Bosslet)		
Target version:	1.9.3		
ruby -v:	trunk	**Backport:**	

Description

=begin
While skimming through ossl_asn1.c I noticed that my patch for infinite length encoding
causes problems when encoding subclasses of OpenSSL::ASN1::Sequence or OpenSSL::ASN1::Set
with infinite length.

E.g. the following fails in trunk:

```
require 'openssl'

sub = Class.new(OpenSSL::ASN1::Sequence)
instance = sub.new([OpenSSL::ASN1::EndOfContent.new])
instance.infinite_length = true
puts instance.to_der
```

=> test.rb:10:in to_der': invalid constructed encoding (OpenSSL::ASN1::ASN1Error) from test.rb:10:in '

This can be fixed with the appended code that checks for subclass relationship instead
of comparing the class directly with Set or Sequence.

Regards,
Martin
=end

History

#1 - 01/26/2011 05:18 PM - nahi (Hiroshi Nakamura)

=begin
It would be good if you show us usecases of subclassing Sequence and Set. Can you?
=end

#2 - 01/26/2011 11:52 PM - MartinBosslet (Martin Bosslet)

=begin
The first thought that came into my mind for changing this was to provide consistency. There are several
similar operations that also use rb_is_kind_of instead of checking the class directly, mainly in
ossl_asn1_default_tag, which is used throughout the entire encoding process.

But there may also be good use cases when subclassing would make sense. For example if one would like
to have some kind of default value set before encoding the value.

A probable scenario where this would make sense could e.g. be an ASN.1 structure that comes with a SET of
certificates. Depending on the application environment, this SET might be predetermined, and to simplify
things one might want to add the predetermined certificates in case someone else forgot to do so. To
achieve this the developer could subclass OpenSSL::ASN1::Set and overwrite #to_der by setting the default
first (if needed) and then delegate to Set's implementation.

I don't know, I couldn't find a more convincing example right now, but I think the consistency argument is
more convincing anyway :)

Regards,
Martin

=end

#3 - 01/29/2011 02:57 AM - MartinBosslet (Martin Bosslet)

=begin
Hi,
I found a imo quite reasonable use case for sublassing Sequence or Set.
Imagine you have an instance of a rather large sequence or set permanently
stored in a variable somewhere - and this instance gets encoded to DER very
often (e.g. for building a digest of the DER bytes). To improve performance
of this operation, one could subclass sequence and cache the DER-encoded
form by lazily setting an instance variable of the subclass, so that the
actual encoding has to be done only once.
=end

#4 - 05/12/2011 08:14 AM - MartinBosslet (Martin Bosslet)

- Assignee set to MartinBosslet (Martin Bosslet)

#5 - 06/26/2011 10:43 PM - MartinBosslet (Martin Bosslet)

- Status changed from Open to Closed

I agree to wait until this causes problems for anyone. The current solution seems fine for now.

Files

fix_cons_encode_inf.diff	2.25 KB	01/26/2011	MartinBosslet (Martin Bosslet)