Ruby master - Feature #6980

OpenSSL support for AEAD additional authenticated data and tags

09/05/2012 04:11 AM - stouset (Stephen Touset)

Status:	Closed	
Priority:	Normal	
Assignee:	MartinBosslet (Martin Bosslet)	
Target version:	2.0.0	

Description

=begin
I've added support to OpenSSL::Cipher to support AEAD modes of operation. AEAD modes allow for plaintext additional authentication data to be combined with a ciphertext to generate a "tag" (e.g., a MAC). This tag can then be verified during decryption to ensure the secret key, nonce (IV), additional authentication data, ciphertext, and tag have not been changed or manipulated.

Usage can be inferred through documentation and tests.

```
cipher = OpenSSL::Cipher.new('aes-256-gcm')
cipher.encrypt
cipher.key = 'key'
cipher.iv = 'iv'
cipher.aad = 'aad'

ct = cipher.update('plain')
tag = cipher.gcm_tag

cipher.reset
cipher.decrypt
cipher.key = 'key'
cipher.iv = 'iv'
cipher.gcm_tag = 'tag'
cipher.aad = 'aad'

cipher.update(ct) + cipher.verify + cipher.final # => 'plain'

cipher.reset
cipher.decrypt
cipher.key = 'key'
cipher.iv = 'iv'
cipher.gcm_tag = 'tag'
cipher.aad = 'aad'

cipher.update(ct[0..-2] << ct[-1].succ) + cipher.verify + cipherfinal # => OpenSSL::Cipher::CipherError
```
=end

Associated revisions

Revision 215b5480 - 12/20/2012 06:03 AM - emboss

- ext/openssl/ossl_cipher.c: add support for Authenticated Encryption with Associated Data (AEAD) for OpenSSL versions that support the GCM encryption mode. It's the only mode supported for now by OpenSSL itself. Add Cipher#authenticated? to detect whether a chosen mode does support Authenticated Encryption.
- test/openssl/test_cipher.rb: add tests for Authenticated Encryption. [Feature #6980] [ruby-core:47426] Thank you, Stephen Touset for providing a patch!

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@38488 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 38488 - 12/20/2012 06:03 AM - emboss

- ext/openssl/ossl_cipher.c: add support for Authenticated Encryption with Associated Data (AEAD) for OpenSSL versions that support the GCM encryption mode. It's the only mode supported for now by OpenSSL itself. Add Cipher#authenticated? to detect whether a chosen mode does support Authenticated Encryption.
- test/openssl/test_cipher.rb: add tests for Authenticated Encryption. [Feature #6980] [ruby-core:47426] Thank you, Stephen Touset for providing a

patch!

Revision 38488 - 12/20/2012 06:03 AM - emboss

- ext/openssl/ossl_cipher.c: add support for Authenticated Encryption with Associated Data (AEAD) for OpenSSL versions that support the GCM encryption mode. It's the only mode supported for now by OpenSSL itself. Add Cipher#authenticated? to detect whether a chosen mode does support Authenticated Encryption.
- test/openssl/test_cipher.rb: add tests for Authenticated Encryption. [Feature #6980] [ruby-core:47426] Thank you, Stephen Touset for providing a patch!

Revision 38488 - 12/20/2012 06:03 AM - emboss

- ext/openssl/ossl_cipher.c: add support for Authenticated Encryption with Associated Data (AEAD) for OpenSSL versions that support the GCM encryption mode. It's the only mode supported for now by OpenSSL itself. Add Cipher#authenticated? to detect whether a chosen mode does support Authenticated Encryption.
- test/openssl/test_cipher.rb: add tests for Authenticated Encryption. [Feature #6980] [ruby-core:47426] Thank you, Stephen Touset for providing a patch!

Revision 38488 - 12/20/2012 06:03 AM - emboss

- ext/openssl/ossl_cipher.c: add support for Authenticated Encryption with Associated Data (AEAD) for OpenSSL versions that support the GCM encryption mode. It's the only mode supported for now by OpenSSL itself. Add Cipher#authenticated? to detect whether a chosen mode does support Authenticated Encryption.
- test/openssl/test_cipher.rb: add tests for Authenticated Encryption. [Feature #6980] [ruby-core:47426] Thank you, Stephen Touset for providing a patch!

Revision 38488 - 12/20/2012 06:03 AM - emboss

- ext/openssl/ossl_cipher.c: add support for Authenticated Encryption with Associated Data (AEAD) for OpenSSL versions that support the GCM encryption mode. It's the only mode supported for now by OpenSSL itself. Add Cipher#authenticated? to detect whether a chosen mode does support Authenticated Encryption.
- test/openssl/test_cipher.rb: add tests for Authenticated Encryption. [Feature #6980] [ruby-core:47426] Thank you, Stephen Touset for providing a patch!

Revision 38488 - 12/20/2012 06:03 AM - emboss

- ext/openssl/ossl_cipher.c: add support for Authenticated Encryption with Associated Data (AEAD) for OpenSSL versions that support the GCM encryption mode. It's the only mode supported for now by OpenSSL itself. Add Cipher#authenticated? to detect whether a chosen mode does support Authenticated Encryption.
- test/openssl/test_cipher.rb: add tests for Authenticated Encryption. [Feature #6980] [ruby-core:47426] Thank you, Stephen Touset for providing a patch!

Revision b9bd8eaf - 12/20/2012 07:42 AM - emboss

- ext/openssl/ossl_cipher.c: fix errors for installations that do not feature Authenticated Encryption.
- ext/openssl/extconf.rb: detect presence of EVP_CTRL_GCM_GET_TAG to determine whether Authenticated Encryption can be used. [Feature #6980] [ruby-core:47426]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@38492 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 38492 - 12/20/2012 07:42 AM - emboss

- ext/openssl/ossl_cipher.c: fix errors for installations that do not feature Authenticated Encryption.
- ext/openssl/extconf.rb: detect presence of EVP_CTRL_GCM_GET_TAG to determine whether Authenticated Encryption can be used. [Feature #6980] [ruby-core:47426]

Revision 38492 - 12/20/2012 07:42 AM - emboss

- ext/openssl/ossl_cipher.c: fix errors for installations that do not feature Authenticated Encryption.
- ext/openssl/extconf.rb: detect presence of EVP_CTRL_GCM_GET_TAG to determine whether Authenticated Encryption can be used. [Feature #6980] [ruby-core:47426]

Revision 38492 - 12/20/2012 07:42 AM - emboss

- ext/openssl/ossl_cipher.c: fix errors for installations that do not feature Authenticated Encryption.
- ext/openssl/extconf.rb: detect presence of EVP_CTRL_GCM_GET_TAG to determine whether Authenticated Encryption can be used. [Feature

#6980] [ruby-core:47426]

Revision 38492 - 12/20/2012 07:42 AM - emboss

- ext/openssl/ossl_cipher.c: fix errors for installations that do not feature Authenticated Encryption.
- ext/openssl/extconf.rb: detect presence of EVP_CTRL_GCM_GET_TAG to determine whether Authenticated Encryption can be used. [Feature #6980] [ruby-core:47426]

Revision 38492 - 12/20/2012 07:42 AM - emboss

- ext/openssl/ossl_cipher.c: fix errors for installations that do not feature Authenticated Encryption.
- ext/openssl/extconf.rb: detect presence of EVP_CTRL_GCM_GET_TAG to determine whether Authenticated Encryption can be used. [Feature #6980] [ruby-core:47426]

Revision 38492 - 12/20/2012 07:42 AM - emboss

- ext/openssl/ossl_cipher.c: fix errors for installations that do not feature Authenticated Encryption.
- ext/openssl/extconf.rb: detect presence of EVP_CTRL_GCM_GET_TAG to determine whether Authenticated Encryption can be used. [Feature #6980] [ruby-core:47426]

History

#1 - 09/05/2012 04:14 AM - stouset (Stephen Touset)

- File openssl_aead_ciphers.patch added

Sorry, patch included unintentional whitespace changes. Reuploaded without whitespace changes.

#2 - 09/05/2012 05:46 AM - MartinBosslet (Martin Bosslet)

- Status changed from Open to Assigned

- Assignee set to MartinBosslet (Martin Bosslet)

- Target version changed from 1.9.3 to 2.0.0

#3 - 09/05/2012 07:39 AM - stouset (Stephen Touset)

=begin
I'm not necessarily happy with a GCM-specific (({gcm_tag})), and an (unimplemented but hypothetical) (({ccm_tag})) et al. But having a single (({tag})) method that probed for which mode it was currently in seemed too magical. I'm open to ideas.
=end

#4 - 10/25/2012 08:27 AM - stouset (Stephen Touset)

I take it given the recent feature freeze that this will *not* make it into 2.0?

#5 - 10/27/2012 07:35 AM - ko1 (Koichi Sasada)

Marin, how about this ticket?

#6 - 11/14/2012 11:07 AM - MartinBosslet (Martin Bosslet)

This would definitely be on my list for 2.0. Sorry for not having been more responsive. I talked with nahi at RubyConf about the tickets that are still open at the moment. I will ask if it is possible to extend the feature freeze for some of the items, there might be a chance. I, too, would like to see this make it into 2.0!

#7 - 11/24/2012 10:39 AM - mame (Yusuke Endoh)

- Priority changed from Normal to 5

Please commit it before preview2, i.e., in this month, and make sure that it causes no problem.

--
Yusuke Endoh mame@tsg.ne.jp

#8 - 12/20/2012 03:03 PM - Anonymous

- Status changed from Assigned to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r38488.

Stephen, thank you for reporting this issue.
Your contribution to Ruby is greatly appreciated.
May Ruby be with you.

- ext/openssl/ossl_cipher.c: add support for Authenticated Encryption with Associated Data (AEAD) for OpenSSL versions that support the GCM encryption mode. It's the only mode supported for now by OpenSSL itself. Add Cipher#authenticated? to detect whether a chosen mode does support Authenticated Encryption.
- test/openssl/test_cipher.rb: add tests for Authenticated Encryption. [Feature #6980] [ruby-core:47426] Thank you, Stephen Touset for providing a patch!

#9 - 12/20/2012 03:10 PM - MartinBosslet (Martin Bosslet)

Thanks again, Stephen! I changed the interface a bit to make it possible to support CCM mode as well once it will be available through the EVP interface. Instead of Cipher#gcm_tag, it is now called Cipher#auth_tag. Because of this change, I also made it Cipher#auth_data=, to indicate that both belong together conceptually.

I also omitted the additional Cipher#verify method, since tag verification will be performed during the call to Cipher#final. I didn't want to introduce an additional method - this way the overall Cipher interface stays coherent.

Files

openssl_aead_ciphers.patch	13.2 KB	09/05/2012	stouset (Stephen Touset)
openssl_aead_ciphers.patch	5.83 KB	09/05/2012	stouset (Stephen Touset)