The last beta for OpenSSL 1.1.0 (1.1.0-pre5) was released a week ago. According to OpenSSL's website[1], OpenSSL 1.1.0 final will be released on 2016-05-12.

OpenSSL 1.1.0 introduces many compatibility breaking changes[2][3] but they are almost done and I think we can start work on it.

Also, I think we can drop support for ancient (-0.9.7) versions of OpenSSL. This can save many compatibility macros (about 700 lines). Considering the last official release is over 9 years ago (2007-02-23) and even 0.9.8/1.0.0 series are no longer supported[1], there should be no reason to continue supporting them. In fact builds with OpenSSL 0.9.7 are already broken since r40461 (2013-04-25, Ruby 2.1.0) but it looks like nobody noticed it. (I personally think all unsupported version should be removed but 0.9.8 is unfortunately still widely used...)


I found some incompatibilities in OpenSSL 1.1.0 which affect Ruby-world API:

- SSL_CTX_set_tmp_ecdh_callback() is removed

  OpenSSL::SSL::SSLContext#tmp_ecdh_callback (exists in Ruby 2.3, [Feature #11356]) breaks. This is used to enable ECDH and specify the curve to be used.

  OpenSSL 1.0.2 and later provide SSL_CTX_set1_curves_list(). This sets the "supported (named) curves" and OpenSSL selects the most appropriate curve from the list. The automatic selection is enabled by SSL_CTX_set_ecdh_auto() on 1.0.2. It is always enabled on 1.1.0.

  Summarizing... ECDH is enabled:

  - OpenSSL 1.1.0: always.
  - OpenSSL 1.0.2, LibreSSL 2.3: if SSL_CTX_set_ecdh_auto() is called.
  - OpenSSL -1.0.2, LibreSSL 2.3: if SSL_CTX_set_tmp_ecdh[_callback,]() is called.

  The curve to be used is:

  - OpenSSL 1.0.2/1.1.0: automatically selected by OpenSSL, if enabled. The "supported curves" can be changed by SSL_CTX_set1_curves[_list,]().
  - LibreSSL 2.3: automatically selected if enabled. The list can't be changed.
  - OpenSSL -1.0.2, LibreSSL 2.3: set by SSL_CTX_set_tmp_ecdh[_callback,]().

  In my patch, I made this deprecated (rb_warn() on SSLContext#setup) it and added SSLContext#set_ecdh_curves:

  - OpenSSL 1.0.2/1.1.0: wraps SSL_CTX_set1_curves_list()
  - OpenSSL -1.0.1, LibreSSL 2.3: wraps SSL_CTX_set_tmp_ecdh() (so only 1 curve can be set)

    # assume the client supports P-224 and P-521
    ctx = OpenSSL::SSL::SSLContext.new
    ctx.set_ecdh_curves "P-384:P-224:P-521"
    svr = OpenSSL::SSL::SSLServer.new(sock, ctx)
    svr.accept # will use P-224

- Accessors for OpenSSL::PKey::{DH,DSA,RSA}'s paramters/keys
Since most structures of OpenSSL are made opaque, we can no longer set directly these values. OpenSSL 1.1.0 provides setter functions (e.g. DH_set0_key()) but they are not equivalent; they require setting all relevant values at the same time. For example, such code (4 usage in tests) is no longer possible:

```ruby
dh = OpenSSL::PKey::DH.new(File.read("dhparams.pem"))
dh.priv_key = OpenSSL::BN.new("000...")
```

In my patch I deprecated them and added XX#set_XXX methods (such as DH#set_key, which sets pub_key and priv_key).

```ruby
dh = OpenSSL::PKey::DH.new(File.read("dhparams.pem"))
dh.set_key(OpenSSL::BN.new("000..."), # pub_key
          OpenSSL::BN.new("000...")) # priv_key
```

This is not beautiful but it looks like there is no other way for this...

- **Security level**

Please see the manpage SSL_CTX_set_security_level(3):

[https://www.openssl.org/docs/manmaster/ssl/SSL_CTX_set_security_level.html](https://www.openssl.org/docs/manmaster/ssl/SSL_CTX_set_security_level.html)

OpenSSL 1.1.0 introduces "security level", which disables insecure settings, such as insecure ciphers, depending on the level. The default level is 1 and this means:

> The security level corresponds to a minimum of 80 bits of security. Any parameters offering below 80 bits of security are excluded. As a result RSA, DSA and DH keys shorter than 1024 bits and ECC keys shorter than 160 bits are prohibited. All export cipher suites are prohibited since they all offer less than 80 bits of security. SSL version 2 is prohibited. Any cipher suite using MD5 for the MAC is also prohibited.

(from 1.1.0-pre5 doc)

New methods would be essential. Actually this breaks many tests which use aNULL cipher suites.

I added SSLContext#security_level and SSLContext#security_level=. in my patch.

---

Here is my try. Because of the amount of OpenSSL changes, almost all .c files had to be modified. Roughly splitted into 38 commits:

[https://github.com/ruby/ruby/compare/trunk...rhenium:feature/openssl-110-v1](https://github.com/ruby/ruby/compare/trunk...rhenium:feature/openssl-110-v1)

(the attached gzip'ed is the same)

This includes:

- Add check for SSL_CTX_clear_options(), which doesn't exist in OpenSSL 0.9.8l and older.
- Drop support for 0.9.7.
- Remove OPENSSL_NO_HMAC support as it have never worked.
- Deprecate (rb_warn(jing) SSLContext#tmp_ecdh_callback and add SSL::SSLContext#set_ecdh_curves method.
- Deprecate parameters/keys setter for PKey::DH, PKey::RSA and PKey::DSA, and add PKey::XX#set_XXX methods.
- Add SSL::SSLContext#security_level and #security_level=.
- Test fixes for open-uri and rubygems (regenerate the test certificates; they are incorrectly created and OpenSSL 1.1.0 starts to complain about it).

This compiles and tests pass with the following versions (on my Linux/gcc environment):

- 0.9.8zh
- 1.0.0t
- 1.0.1s
- 1.0.2g
- OpenSSL master
- LibreSSL 2.3.3

Note that this doesn't compile with 1.1.0-pre5; OpenSSL GH-975 is required (already in master and the 1.1.0 final will include this).

[https://github.com/openssl/openssl/pull/975](https://github.com/openssl/openssl/pull/975)

The patch is far from perfect - I'm not sure about the new methods and also probably there are bugs.
Any comments are welcome, I'll update the patch.

Thanks,

Associated revisions

Revision 9199bec9 - 06/05/2016 12:38 PM - rhe
openssl: check existence of RAND_pseudo_bytes()

- ext/openssl/extconf.rb: Check if RAND_pseudo_bytes() is usable. It is marked as deprecated in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_rand.c: Disable Random.pseudo_bytes if RAND_pseudo_bytes() is unavailable.

- test/openssl/test_random.rb: Don't test Random.pseudo_bytes if not defined.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55282 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55282 - 06/05/2016 12:38 PM - rhenium (Kazuki Yamaguchi)
openssl: check existence of RAND_pseudo_bytes()

- ext/openssl/extconf.rb: Check if RAND_pseudo_bytes() is usable. It is marked as deprecated in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_rand.c: Disable Random.pseudo_bytes if RAND_pseudo_bytes() is unavailable.

- test/openssl/test_random.rb: Don't test Random.pseudo_bytes if not defined.

Revision 55282 - 06/05/2016 12:38 PM - rhe
openssl: check existence of RAND_pseudo_bytes()

- ext/openssl/extconf.rb: Check if RAND_pseudo_bytes() is usable. It is marked as deprecated in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_rand.c: Disable Random.pseudo_bytes if RAND_pseudo_bytes() is unavailable.

- test/openssl/test_random.rb: Don't test Random.pseudo_bytes if not defined.

Revision 55282 - 06/05/2016 12:38 PM - rhe
openssl: check existence of RAND_pseudo_bytes()

- ext/openssl/extconf.rb: Check if RAND_pseudo_bytes() is usable. It is marked as deprecated in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_rand.c: Disable Random.pseudo_bytes if RAND_pseudo_bytes() is unavailable.

- test/openssl/test_random.rb: Don't test Random.pseudo_bytes if not defined.

Revision 55282 - 06/05/2016 12:38 PM - rhe
openssl: check existence of RAND_pseudo_bytes()
• ext/openssl/extconf.rb: Check if RAND_pseudo_bytes() is usable. It is marked as deprecated in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

• ext/openssl/ossl_rand.c: Disable Random.pseudo_bytes if RAND_pseudo_bytes() is unavailable.

• test/openssl/test_random.rb: Don’t test Random.pseudo_bytes if not defined.

Revision e47bb7d - 06/05/2016 12:46 PM - rhe
openssl: support OpenSSL 1.1.0’s new multi-threading API

• ext/openssl/extconf.rb: Check absence of CRYPTO_lock() to see if the OpenSSL has the new threading API. In OpenSSL <= 1.0.2, an application had to set locking callbacks to use OpenSSL in a multi-threaded environment. OpenSSL 1.1.0 now finds pthreads or Windows threads so we don’t need to do something special. [ruby-core:75225] [Feature #12324]

Also check existence of *_up_ref(). Some structures in OpenSSL have a reference counter. We used to increment it with CRYPTO_add() which is a part of the old API.

• ext/openssl/openssl_missing.h: Implement *_up_ref() if missing.

• ext/openssl/openssl.c: Don’t set locking callbacks if unneeded.

• ext/openssl/ossl_pkey.c, ext/openssl/ossl_ssl.c, ext/openssl/ossl_x509cert.c, ext/openssl/ossl_x509crl.c, ext/openssl/ossl_x509store.c: Use *_up_ref() instead of CRYPTO_add().

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55283 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55283 - 06/05/2016 12:46 PM - rhenium (Kazuki Yamaguchi)
openssl: support OpenSSL 1.1.0’s new multi-threading API

• ext/openssl/extconf.rb: Check absence of CRYPTO_lock() to see if the OpenSSL has the new threading API. In OpenSSL <= 1.0.2, an application had to set locking callbacks to use OpenSSL in a multi-threaded environment. OpenSSL 1.1.0 now finds pthreads or Windows threads so we don’t need to do something special. [ruby-core:75225] [Feature #12324]

Also check existence of *_up_ref(). Some structures in OpenSSL have a reference counter. We used to increment it with CRYPTO_add() which is a part of the old API.

• ext/openssl/openssl_missing.h: Implement *_up_ref() if missing.

• ext/openssl/openssl.c: Don’t set locking callbacks if unneeded.

• ext/openssl/ossl_pkey.c, ext/openssl/ossl_ssl.c, ext/openssl/ossl_x509cert.c, ext/openssl/ossl_x509crl.c, ext/openssl/ossl_x509store.c: Use *_up_ref() instead of CRYPTO_add().

Revision 55283 - 06/05/2016 12:46 PM - rhe
openssl: support OpenSSL 1.1.0’s new multi-threading API

• ext/openssl/extconf.rb: Check absence of CRYPTO_lock() to see if the OpenSSL has the new threading API. In OpenSSL <= 1.0.2, an application had to set locking callbacks to use OpenSSL in a multi-threaded environment. OpenSSL 1.1.0 now finds pthreads or Windows threads so we don’t need to do something special. [ruby-core:75225] [Feature #12324]

Also check existence of *_up_ref(). Some structures in OpenSSL have a reference counter. We used to increment it with CRYPTO_add() which is a part of the old API.

• ext/openssl/openssl_missing.h: Implement *_up_ref() if missing.

• ext/openssl/openssl.c: Don’t set locking callbacks if unneeded.
ext/openssl/ossl_x509store.c: Use _up_ref() instead of CRYPTO_add().

Revision 55283 - 06/05/2016 12:46 PM - rhe
openssl: support OpenSSL 1.1.0’s new multi-threading API

- ext/openssl/extconf.rb: Check absence of CRYPTO_lock() to see if the OpenSSL has the new threading API. In OpenSSL <= 1.0.2, an application had to set locking callbacks to use OpenSSL in a multi-threaded environment. OpenSSL 1.1.0 now finds pthreads or Windows threads so we don’t need to do something special. [ruby-core:75225] [Feature #12324]

Also check existence of _up_ref(). Some structures in OpenSSL have a reference counter. We used to increment it with CRYPTO_add() which is a part of the old API.

- ext/openssl/openssl_missing.h: Implement _up_ref() if missing.
- ext/openssl/ossl.c: Don’t set locking callbacks if unneeded.
- ext/openssl/oss1_pkey.c, ext/openssl/oss1_ssl.c,
  ext/openssl/ossl_x509cert.c, ext/openssl/ossl_x509crl.c,
  ext/openssl/ossl_x509store.c: Use _up_ref() instead of CRYPTO_add().

Revision 55283 - 06/05/2016 12:46 PM - rhe
openssl: support OpenSSL 1.1.0’s new multi-threading API

- ext/openssl/extconf.rb: Check absence of CRYPTO_lock() to see if the OpenSSL has the new threading API. In OpenSSL <= 1.0.2, an application had to set locking callbacks to use OpenSSL in a multi-threaded environment. OpenSSL 1.1.0 now finds pthreads or Windows threads so we don’t need to do something special. [ruby-core:75225] [Feature #12324]

Also check existence of _up_ref(). Some structures in OpenSSL have a reference counter. We used to increment it with CRYPTO_add() which is a part of the old API.

- ext/openssl/openssl_missing.h: Implement _up_ref() if missing.
- ext/openssl/ossl.c: Don’t set locking callbacks if unneeded.
- ext/openssl/oss1_pkey.c, ext/openssl/oss1_ssl.c,
  ext/openssl/ossl_x509cert.c, ext/openssl/ossl_x509crl.c,
  ext/openssl/ossl_x509store.c: Use _up_ref() instead of CRYPTO_add().

Revision 63abe007 - 06/05/2016 03:00 PM - rhe
openssl: adapt OpenSSL::PKey to OpenSSL 1.1.0 opaque structs

- ext/openssl/openssl_missing.[ch]: Implement EVP_PKEY_get0_() and {RSA,DSA,EC_KEY,DH}get0() functions.
  OpenSSL 1.1.0 makes EVP_PKEY/RSA/DSA/DH opaque. We used to provide setter methods for each parameter of each PKey type, for example PKey::RSA#e, but this is no longer possible because the new API RSA_set0_key() requires the ‘n’ at the same time. This commit adds deprecation warning to them and adds PKey::#set_ methods as direct wrapper for those new APIs. For example, ‘rsa.e = 3’ now needs to be rewritten as ‘rsa.set_key(rsa.n, 3, rsa.d)’.
  [ruby-core:75225] [Feature #12324]

- ext/openssl/oss1_pkey.*.[ch]: Use the new accessor functions. Implement RSAAsset_[key.factors.crt_params], DSAAsset_[key,pqg], DHAsset_[key,pqg]. Emit a warning with rb_warning() when old setter methods are used.

- test/drb/ut_array_drbssl.rb, test/drb/ut_drb_drbssl.rb,
  test/rubygems/test_gem_remote_fetcher.rb: Don’t set a priv_key for DH object that are used in tmp_dh_callback. Generating a new key pair every time should be fine - actually the private exponent is ignored in OpenSSL >= 1.0.2f/1.0.1r even if we explicitly set.
  https://www.openssl.org/news/secadv/20160128.txt

11/27/2020

5/26
Revision 55285 - 06/05/2016 03:00 PM - rhenium (Kazuki Yamaguchi)
opensearch: adapt OpenSSL::PKey to OpenSSL 1.1.0 opaque structs

- ext/openssl/openssl_missing.[ch]: Implement EVP_PKEY_get0() and 
  {RSA,DSA,EC_KEY,DH}.get0() functions.
  OpenSSL 1.1.0 makes EVP_PKEY/RSA/DSA/DH opaque. We used to provide
  setter methods for each parameter of each PKey type, for example
  PKey::RSA#:e=, but this is no longer possible because the new API
  RSA_set0_key() requires the 'n' at the same time. This commit adds
  deprecation warning to them and adds PKey:::set_ methods as direct
  wrapper for those new APIs. For example, 'rsa.e = 3' now needs to be
  rewritten as 'rsa.set_key(rsa.n, 3, rsa.d)'.
  [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_pkey*.ch]: Use the new accessor functions. Implement
  RSA#set_[key,factors,crt_params], DSA#set_[key,pqg], DH#set_[key,pqg].
  Emit a warning with rb_warning() when old setter methods are used.

- test/drb/ut_array_drbssl.rb, test/drb/ut_drb_drbssl.rb,
  test/rubygems/test_gem_remote_fetcher.rb: Don't set a priv_key for DH
  object that are used in tmp_dh_callback. Generating a new key pair
  every time should be fine - actually the private exponent is ignored
  in OpenSSL >= 1.0.2f/1.0.1r even if we explicitly set.
  https://www.openssl.org/news/secadv/20160128.txt

Revision 55285 - 06/05/2016 03:00 PM - rhe

openssl: adapt OpenSSL::PKey to OpenSSL 1.1.0 opaque structs

- ext/openssl/openssl_missing.[ch]: Implement EVP_PKEY_get0() and 
  {RSA,DSA,EC_KEY,DH}.get0() functions.
  OpenSSL 1.1.0 makes EVP_PKEY/RSA/DSA/DH opaque. We used to provide
  setter methods for each parameter of each PKey type, for example
  PKey::RSA#:e=, but this is no longer possible because the new API
  RSA_set0_key() requires the 'n' at the same time. This commit adds
  deprecation warning to them and adds PKey:::set_ methods as direct
  wrapper for those new APIs. For example, 'rsa.e = 3' now needs to be
  rewritten as 'rsa.set_key(rsa.n, 3, rsa.d)'.
  [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_pkey*.ch]: Use the new accessor functions. Implement
  RSA#set_[key,factors,crt_params], DSA#set_[key,pqg], DH#set_[key,pqg].
  Emit a warning with rb_warning() when old setter methods are used.

- test/drb/ut_array_drbssl.rb, test/drb/ut_drb_drbssl.rb,
  test/rubygems/test_gem_remote_fetcher.rb: Don't set a priv_key for DH
  object that are used in tmp_dh_callback. Generating a new key pair
  every time should be fine - actually the private exponent is ignored
  in OpenSSL >= 1.0.2f/1.0.1r even if we explicitly set.
  https://www.openssl.org/news/secadv/20160128.txt

Revision 55285 - 06/05/2016 03:00 PM - rhe

openssl: adapt OpenSSL::PKey to OpenSSL 1.1.0 opaque structs

- ext/openssl/openssl_missing.[ch]: Implement EVP_PKEY_get0() and 
  {RSA,DSA,EC_KEY,DH}.get0() functions.
  OpenSSL 1.1.0 makes EVP_PKEY/RSA/DSA/DH opaque. We used to provide
  setter methods for each parameter of each PKey type, for example
  PKey::RSA#:e=, but this is no longer possible because the new API
  RSA_set0_key() requires the 'n' at the same time. This commit adds
  deprecation warning to them and adds PKey:::set_ methods as direct
  wrapper for those new APIs. For example, 'rsa.e = 3' now needs to be
  rewritten as 'rsa.set_key(rsa.n, 3, rsa.d)'.
  [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_pkey*.ch]: Use the new accessor functions. Implement
  RSA#set_[key,factors,crt_params], DSA#set_[key,pqg], DH#set_[key,pqg].

11/27/2020 6/26
Emit a warning with rb_warning() when old setter methods are used.

- test/drb/ut_array_drbssl.rb, test/drb/ut_drb_drbssl.rb,
  test/rubygems/test_gem_remote_fetcher.rb: Don't set a priv_key for DH
  object that are used in tmp_dh_callback. Generating a new key pair
every time should be fine - actually the private exponent is ignored
in OpenSSL >= 1.0.2f/1.0.1r even if we explicitly set.
  https://www.openssl.org/news/secadv/20160128.txt

Revision 55285 - 06/05/2016 03:00 PM - rhe
openssl: adapt OpenSSL::PKey to OpenSSL 1.1.0 opaque structs

- ext/openssl/openssl_missing.[ch]: Implement EVP_PKEY_get0__() and
  {RSA,DSA,EC_KEY,DH}get0() functions.
  OpenSSL 1.1.0 makes EVP_PKEY/RSA/DSA/DH opaque. We used to provide
  setter methods for each parameter of each PKey type, for example
  PKey::RSA#e=, but this is no longer possible because the new API
  RSA_set0_key() requires the 'n' at the same time. This commit adds
  deprecation warning to them and adds PKey::#set_* methods as direct
  wrapper for those new APIs. For example, 'rsa.e = 3' now needs to be
  rewritten as 'rsa.set_key(rsa.n, 3, rsa.d)'.
  [ruby-core:75225] [Feature #12324]

- ext/openssl/openssl_missing.[ch]: Implement RSA#set{key,factors,crt_params}, DSA#set{key,pqg}, DH#set{key,pqg}.
  Emit a warning with rb_warning() when old setter methods are used.

- test/drb/ut_array_drbssl.rb, test/drb/ut_drb_drbssl.rb,
  test/rubygems/test_gem_remote_fetcher.rb: Don't set a priv_key for DH
  object that are used in tmp_dh_callback. Generating a new key pair
every time should be fine - actually the private exponent is ignored
in OpenSSL >= 1.0.2f/1.0.1r even if we explicitly set.
  https://www.openssl.org/news/secadv/20160128.txt

Revision 0a523ab2 - 06/05/2016 03:35 PM - rhe
openssl: adapt to OpenSSL 1.1.0 opaque structs

- ext/openssl/openssl_missing.[ch]: Implement EVP_PKEY_get0__() and
  {RSA,DSA,EC_KEY,DH}get0() functions.
  OpenSSL 1.1.0 made most of its
  structures opaque and requires use of these accessor functions.
  [ruby-core:75225] [Feature #12324]

- ext/openssl/openssl_missing.[ch]: Implement them if missing.

- ext/openssl/ossl*.c: Use these accessor functions.

- test/openssl/test_hmac.rb: Add missing test for HMAC#reset.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55287 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55287 - 06/05/2016 03:35 PM - rhenium (Kazuki Yamaguchi)
openssl: adapt to OpenSSL 1.1.0 opaque structs

- ext/openssl/openssl_missing.[ch]: Implement them if missing.

- ext/openssl/ossl*.c: Use these accessor functions.

- test/openssl/test_hmac.rb: Add missing test for HMAC#reset.

Revision 55287 - 06/05/2016 03:35 PM - rhe
openssl: adapt to OpenSSL 1.1.0 opaque structs

- ext/openssl/extconf.rb: Check existence of accessor functions that don't exist in OpenSSL 0.9.8. OpenSSL 1.1.0 made most of its structures opaque and requires use of these accessor functions. [ruby-core:75225] [Feature #12324]

- ext/openssl/openssl_missing.[ch]: Implement them if missing.

- ext/openssl/oss1*.c: Use these accessor functions.

- test/openssl/test_hmac.rb: Add missing test for HMAC#reset.

Revision 55287 - 06/05/2016 03:35 PM - rhe
openssl: adapt to OpenSSL 1.1.0 opaque structs

- ext/openssl/extconf.rb: Check existence of accessor functions that don't exist in OpenSSL 0.9.8. OpenSSL 1.1.0 made most of its structures opaque and requires use of these accessor functions. [ruby-core:75225] [Feature #12324]

- ext/openssl/openssl_missing.[ch]: Implement them if missing.

- ext/openssl/oss1*.c: Use these accessor functions.

- test/openssl/test_hmac.rb: Add missing test for HMAC#reset.

Revision 55287 - 06/05/2016 03:35 PM - rhe
openssl: adapt to OpenSSL 1.1.0 opaque structs

- ext/openssl/extconf.rb: Check existence of accessor functions that don't exist in OpenSSL 0.9.8. OpenSSL 1.1.0 made most of its structures opaque and requires use of these accessor functions. [ruby-core:75225] [Feature #12324]

- ext/openssl/openssl_missing.[ch]: Implement them if missing.

- ext/openssl/oss1*.c: Use these accessor functions.

- test/openssl/test_hmac.rb: Add missing test for HMAC#reset.

Revision c7b583a7 - 06/05/2016 04:18 PM - rhenium (Kazuki Yamaguchi)
openssl: avoid d2i_ASN1_BOOLEAN()

- ext/openssl/oss1_asn1.c (decode_bool): Do the same thing as d2i_ASN1_BOOLEAN() does by ourselves. This function is removed in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55288 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55288 - 06/05/2016 04:18 PM - rhenium (Kazuki Yamaguchi)
openssl: avoid d2i_ASN1_BOOLEAN()

- ext/openssl/oss1_asn1.c (decode_bool): Do the same thing as d2i_ASN1_BOOLEAN() does by ourselves. This function is removed in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55288 - 06/05/2016 04:18 PM - rhe
openssl: avoid d2i_ASN1_BOOLEAN()

- ext/openssl/oss1_asn1.c (decode_bool): Do the same thing as d2i_ASN1_BOOLEAN() does by ourselves. This function is removed in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55288 - 06/05/2016 04:18 PM - rhe
openssl: avoid d2i_ASN1_BOOLEAN()
• ext/openssl/ossl_asn1.c (decode_bool): Do the same thing as d2i_ASN1_BOOLEAN() does by ourselves. This function is removed in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55288 - 06/05/2016 04:18 PM - rhe
openssl: avoid d2i_ASN1_BOOLEAN()

• ext/openssl/ossl_asn1.c (decode_bool): Do the same thing as d2i_ASN1_BOOLEAN() does by ourselves. This function is removed in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 7451c146 - 06/05/2016 04:36 PM - rhe
openssl: use SSL_is_server()

• ext/openssl/extconf.rb: Check existence of SSL_is_server(). This function was introduced in OpenSSL 1.0.2. [ruby-core:75225] [Feature #12324]

• ext/openssl/osssl_missing.h: Implement SSL_is_server() if missing.

• ext/openssl/osssl_ssl.c (ssl_info_cb): Use SSL_is_server() to see if the SSL is server. The state machine in OpenSSL was rewritten and SSL_get_state() no longer returns SSL_ST_ACCEPT.

(osssl_ssl_cipher_to_ary, ossl_sslctx_session_get_cb): Add some consts to suppress warning.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55289 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55289 - 06/05/2016 04:36 PM - rhenium (Kazuki Yamaguchi)
openssl: use SSL_is_server()

• ext/openssl/extconf.rb: Check existence of SSL_is_server(). This function was introduced in OpenSSL 1.0.2. [ruby-core:75225] [Feature #12324]

• ext/openssl/osssl_missing.h: Implement SSL_is_server() if missing.

• ext/openssl/osssl_ssl.c (ssl_info_cb): Use SSL_is_server() to see if the SSL is server. The state machine in OpenSSL was rewritten and SSL_get_state() no longer returns SSL_ST_ACCEPT.

(osssl_ssl_cipher_to_ary, ossl_sslctx_session_get_cb): Add some consts to suppress warning.

Revision 55289 - 06/05/2016 04:36 PM - rhe
openssl: use SSL_is_server()

• ext/openssl/extconf.rb: Check existence of SSL_is_server(). This function was introduced in OpenSSL 1.0.2. [ruby-core:75225] [Feature #12324]

• ext/openssl/osssl_missing.h: Implement SSL_is_server() if missing.

• ext/openssl/osssl_ssl.c (ssl_info_cb): Use SSL_is_server() to see if the SSL is server. The state machine in OpenSSL was rewritten and SSL_get_state() no longer returns SSL_ST_ACCEPT.

(osssl_ssl_cipher_to_ary, ossl_sslctx_session_get_cb): Add some consts to suppress warning.

Revision 55289 - 06/05/2016 04:36 PM - rhe
openssl: use SSL_is_server()

• ext/openssl/extconf.rb: Check existence of SSL_is_server(). This function was introduced in OpenSSL 1.0.2. [ruby-core:75225] [Feature #12324]

• ext/openssl/osssl_missing.h: Implement SSL_is_server() if missing.

(osssl_ssl_cipher_to_ary, ossl_sslctx_session_get_cb): Add some consts to suppress warning.

Revision 55289 - 06/05/2016 04:36 PM - rhe
openssl: use SSL_is_server()
- ext/openssl/ossl_ssl.c (ssl_info_cb): Use SSL_is_server() to see if
  the SSL is server. The state machine in OpenSSL was rewritten and
  SSL_get_state() no longer returns SSL_ST_ACCEPT.

(osssl_ssl_cipher_to_ary, ossl_sslctx_session_get_cb): Add some
consts to suppress warning.

**Revision 55289 - 06/05/2016 04:36 PM - rhe**

openssl: use SSL_is_server()

- ext/openssl/extconf.rb: Check existence of SSL_is_server(). This
  function was introduced in OpenSSL 1.0.2. [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_missing.h: Implement SSL_is_server() if missing.

- ext/openssl/ossl_ssl.c (ssl_info_cb): Use SSL_is_server() to see if
  the SSL is server. The state machine in OpenSSL was rewritten and
  SSL_get_state() no longer returns SSL_ST_ACCEPT.

(osssl_ssl_cipher_to_ary, ossl_sslctx_session_get_cb): Add some
consts to suppress warning.

**Revision c487224f - 06/06/2016 08:11 AM - rhe**

openssl: fix free function of OpenSSL::Cipher

- ext/openssl/ossl_cipher.c (osssl_cipher_free): Use EVP_CIPHER_CTX_free() to free EVP_CIPHER_CTX allocated by
  EVP_CIPHER_CTX_new(). [ruby-core:75225] [Feature #12324]

[git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55294 b2dd03c8-39d4-4d8f-98ff-823fe69b080e]

**Revision 55294 - 06/06/2016 08:11 AM - rhenium (Kazuki Yamaguchi)**

openssl: fix free function of OpenSSL::Cipher

- ext/openssl/ossl_cipher.c (osssl_cipher_free): Use EVP_CIPHER_CTX_free() to free EVP_CIPHER_CTX allocated by
  EVP_CIPHER_CTX_new(). [ruby-core:75225] [Feature #12324]

**Revision 55294 - 06/06/2016 08:11 AM - rhe**

openssl: fix free function of OpenSSL::Cipher

- ext/openssl/ossl_cipher.c (osssl_cipher_free): Use EVP_CIPHER_CTX_free() to free EVP_CIPHER_CTX allocated by
  EVP_CIPHER_CTX_new(). [ruby-core:75225] [Feature #12324]

**Revision 55294 - 06/06/2016 08:11 AM - rhe**

openssl: fix free function of OpenSSL::Cipher

- ext/openssl/ossl_cipher.c (osssl_cipher_free): Use EVP_CIPHER_CTX_free() to free EVP_CIPHER_CTX allocated by
  EVP_CIPHER_CTX_new(). [ruby-core:75225] [Feature #12324]

**Revision 55294 - 06/06/2016 08:11 AM - rhe**

openssl: fix free function of OpenSSL::Cipher

- ext/openssl/ossl_cipher.c (osssl_cipher_free): Use EVP_CIPHER_CTX_free() to free EVP_CIPHER_CTX allocated by
  EVP_CIPHER_CTX_new(). [ruby-core:75225] [Feature #12324]

**Revision 55294 - 06/06/2016 08:11 AM - rhe**

openssl: avoid deprecated version-specific ssl methods if necessary

- ext/openssl/extconf.rb: Check for SSL_CTX_set_min_proto_version() macro added in OpenSSL 1.1.0. Version-specific methods, such as
  TLSv1_method(), are deprecated in OpenSSL 1.1.0. We need to use
  version-flexible methods (TLS_"method() or SSLv23_"method()) and
disable other protocol versions as necessary. [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_ssl.c: Use SSL_CTX_set_[min,max]_proto_version() to
openssl: avoid deprecated version-specific ssl methods if necessary

- ext/openssl/extconf.rb: Check for SSL_CTX_set_min_proto_version() macro added in OpenSSL 1.1.0. Version-specific methods, such as TLSv1_method(), are deprecated in OpenSSL 1.1.0. We need to use version-flexible methods (TLS_*method() or SSLv23_*method()) and disable other protocol versions as necessary.
  
  [ruby-core:75225] [Feature #12324]

- ext/openssl/iossl_ssl.c: Use SSL_CTX_set_[min,max]_proto_version() to fix the protocol version.

Revision 55304 - 06/07/2016 05:57 AM - rhe

openssl: avoid deprecated version-specific ssl methods if necessary

- ext/openssl/extconf.rb: Check for SSL_CTX_set_min_proto_version() macro added in OpenSSL 1.1.0. Version-specific methods, such as TLSv1_method(), are deprecated in OpenSSL 1.1.0. We need to use version-flexible methods (TLS_*method() or SSLv23_*method()) and disable other protocol versions as necessary.
  
  [ruby-core:75225] [Feature #12324]

- ext/openssl/iossl_ssl.c: Use SSL_CTX_set_[min,max]_proto_version() to fix the protocol version.

Revision 55304 - 06/07/2016 05:57 AM - rhe

openssl: avoid deprecated version-specific ssl methods if necessary

- ext/openssl/extconf.rb: Check for SSL_CTX_set_min_proto_version() macro added in OpenSSL 1.1.0. Version-specific methods, such as TLSv1_method(), are deprecated in OpenSSL 1.1.0. We need to use version-flexible methods (TLS_*method() or SSLv23_*method()) and disable other protocol versions as necessary.
  
  [ruby-core:75225] [Feature #12324]

- ext/openssl/iossl_ssl.c: Use SSL_CTX_set_[min,max]_proto_version() to fix the protocol version.

Revision 55304 - 06/07/2016 05:57 AM - rhe

openssl: avoid deprecated version-specific ssl methods if necessary

- ext/openssl/extconf.rb: Check for SSL_CTX_set_min_proto_version() macro added in OpenSSL 1.1.0. Version-specific methods, such as TLSv1_method(), are deprecated in OpenSSL 1.1.0. We need to use version-flexible methods (TLS_*method() or SSLv23_*method()) and disable other protocol versions as necessary.
  
  [ruby-core:75225] [Feature #12324]

- ext/openssl/iossl_ssl.c: Use SSL_CTX_set_[min,max]_proto_version() to fix the protocol version.

Revision 55304 - 06/07/2016 05:57 AM - rhe

openssl: avoid deprecated version-specific ssl methods if necessary

- ext/openssl/extconf.rb: Check for SSL_CTX_set_min_proto_version() macro added in OpenSSL 1.1.0. Version-specific methods, such as TLSv1_method(), are deprecated in OpenSSL 1.1.0. We need to use version-flexible methods (TLS_*method() or SSLv23_*method()) and disable other protocol versions as necessary.
  
  [ruby-core:75225] [Feature #12324]

- ext/openssl/iossl_ssl.c: Use SSL_CTX_set_[min,max]_proto_version() to fix the protocol version.

Revision 15784ec - 06/07/2016 07:52 AM - rhe

openssl: add SSL::SSLContext#security_level=[,]

- ext/openssl/extconf.rb: Check for SSL_CTX_get_security_level(). OpenSSL 1.1.0 introduced "security level".
ext/openssl/ossl_ssl.c (ossl_sslctx_{get,set}_security_level): Add SSLContext#security_level and #security_level=.

*test/openssl/test_ssl.rb (test_security_level): Add test. ...but this doesn't actually test it. Because #security_level= is necessary in order to run other tests on OpenSSL 1.1.0, go without tests for now. Will fix after converting SSLContext#key= and #cert= to normal methods.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55309 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55309 - 06/07/2016 07:52 AM - rhenium (Kazuki Yamaguchi)
openssl: Add SSL::SSLContext#security_level{=,}

ext/openssl/extconf.rb: Check for SSL_CTX_get_security_level(). OpenSSL 1.1.0 introduced "security level".
[ruby-core:75225] [Feature #12324]

ext/openssl/ossl_ssl.c (ossl_sslctx_{get,set}_security_level): Add SSLContext#security_level and #security_level=.

*test/openssl/test_ssl.rb (test_security_level): Add test. ...but this doesn't actually test it. Because #security_level= is necessary in order to run other tests on OpenSSL 1.1.0, go without tests for now. Will fix after converting SSLContext#key= and #cert= to normal methods.

Revision 55309 - 06/07/2016 07:52 AM - rhe
openssl: Add SSL::SSLContext#security_level{=,}

ext/openssl/extconf.rb: Check for SSL_CTX_get_security_level(). OpenSSL 1.1.0 introduced "security level".
[ruby-core:75225] [Feature #12324]

ext/openssl/ossl_ssl.c (ossl_sslctx_{get,set}_security_level): Add SSLContext#security_level and #security_level=.

*test/openssl/test_ssl.rb (test_security_level): Add test. ...but this doesn't actually test it. Because #security_level= is necessary in order to run other tests on OpenSSL 1.1.0, go without tests for now. Will fix after converting SSLContext#key= and #cert= to normal methods.

Revision 55309 - 06/07/2016 07:52 AM - rhe
openssl: Add SSL::SSLContext#security_level{=,}

ext/openssl/extconf.rb: Check for SSL_CTX_get_security_level(). OpenSSL 1.1.0 introduced "security level".
[ruby-core:75225] [Feature #12324]

ext/openssl/ossl_ssl.c (ossl_sslctx_{get,set}_security_level): Add SSLContext#security_level and #security_level=.

*test/openssl/test_ssl.rb (test_security_level): Add test. ...but this doesn't actually test it. Because #security_level= is necessary in order to run other tests on OpenSSL 1.1.0, go without tests for now. Will fix after converting SSLContext#key= and #cert= to normal methods.
- ext/openssl/ossl_ssl.c (ossl_sslctx_[get,set]_security_level): Add
  SSLContext#security_level and #security_level=.

- test/openssl/test_ssl.rb (test_security_level): Add test. ...but this
doesn't actually test it. Because #security_level= is necessary in
order to run other tests on OpenSSL 1.1.0, go without tests for now.
Will fix after converting SSLContext#key= and #cert= to normal methods.

Revision b257af88 - 06/07/2016 12:20 PM - rhe
openssl: adjust tests for OpenSSL 1.1.0

This fixes make test-all TESTS=openssl with OpenSSL master.

- test/openssl/test_x509name.rb: Don't register OID for 'emailAddress'
  and 'serialNumber'. A recent change in OpenSSL made OBJ_create() reject
  an already existing OID. They were needed to run tests with
  OpenSSL 0.9.6 which is now unsupported.
  [https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=52832e470f5f68c222249ae5b539aeb3c74cdb25
  [ruby-core:75225] [Feature #12324]]

- test/openssl/test_ssl_session.rb (test_server_session): Duplicate
  SSL::Session before re-adding to the session store. OpenSSL 1.1.0
  starts rejecting SSL_SESSION once removed by SSL_CTX_remove_session().
  [https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=7c2d44e2547650102cd16d238125b76112a75
  [ruby-core:75225] [Feature #12324]]

- test/openssl/test_pkey_ec.rb (setup): Remove X25519 from @keys. X25519 is
  now in OpenSSL 1.1.0 but this is for key agreement and not for
  signing.

- test/openssl/test_pair.rb, test/openssl/test_ssl.rb,
  test/openssl/utils.rb: Set security level to 0 when using aNULL cipher
  suites.

- test/openssl/utils.rb: Use 1024 bits DSA key for client certificates.

- test/openssl/test_engine.rb: Run each test in separate process.
  We can no longer cleanup engines explicitly as ENGINE_cleanup() was
  removed.
  [https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=6d4fb1d59e61aacefa25edc4f5acef1ac93f743]

- ext/openssl/ossl_engine.c (ossl_engine_s_cleanup): Add a note to the
  RDoc for Engine.cleanup.

- ext/openssl/lib/osslssl/digest.rb: Don't define constants for DSS,
  DSS1 and SHA(-0) when using with OpenSSL 1.1.0. They are removed.

- test/openssl/test_digest.rb, test/openssl/test_pkey_dsa.rb,
  test/openssl/test_pkey_dsa.rb, test/openssl/test_ssl.rb,
  test/openssl/test_x509cert.rb, test/openssl/test_x509req.rb: Don't
  test unsupported hash functions.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55314 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55314 - 06/07/2016 12:20 PM - rhenium (Kazuki Yamaguchi)
openssl: adjust tests for OpenSSL 1.1.0

This fixes make test-all TESTS=openssl with OpenSSL master.

- test/openssl/test_x509name.rb: Don't register OID for 'emailAddress'
  and 'serialNumber'. A recent change in OpenSSL made OBJ_create() reject
  an already existing OID. They were needed to run tests with
  OpenSSL 0.9.6 which is now unsupported.
  [https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=52832e470f5f68c222249ae5b539aeb3c74cdb25
  [ruby-core:75225] [Feature #12324]]

- test/openssl/test_ssl_session.rb (test_server_session): Duplicate
  SSL::Session before re-adding to the session store. OpenSSL 1.1.0
  starts rejecting SSL_SESSION once removed by SSL_CTX_remove_session().
  [https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=7c2d44e2547650102cd16d238125b76112a75
  [ruby-core:75225] [Feature #12324]]
- test/openssl/test_pkey_ec.rb (setup): Remove X25519 from @keys. X25519 is new in OpenSSL 1.1.0 but this is for key agreement and not for signing.

- test/openssl/test_pair.rb, test/openssl/test_ssl.rb, test/openssl/utils.rb: Set security level to 0 when using aNULL cipher suites.

- test/openssl/utils.rb: Use 1024 bits DSA key for client certificates.

- test/openssl/test_engine.rb: Run each test in separate process. We can no longer cleanup engines explicitly as ENGINE_cleanups() was removed. https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=6d4fb1d59e61aacefa25edc4fe5acfe1ac93f743

- ext/openssl/ossl_engine.c (ossl_engine_s_cleanup): Add a note to the RDoc for Engine.cleanup.

- ext/openssl/lib/openssl/digest.rb: Don't define constants for DSS, DSS1 and SHA(-0) when using with OpenSSL 1.1.0. They are removed.


Revision 55314 - 06/07/2016 12:20 PM - rhe
openssl: adjust tests for OpenSSL 1.1.0

This fixes make test-all TESTS=openssl with OpenSSL master.

- test/openssl/test_x509name.rb: Don't register OID for 'emailAddress' and 'serialNumber'. A recent change in OpenSSL made OBJ_create() reject an already existing OID. They were needed to run tests with OpenSSL 0.9.6 which is now unsupported. https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=52832e470f5fe8c222249ae5b539aeb3c74cdb25 [ruby-core:75225] [Feature #12324]

- test/openssl/test_ssl_session.rb (test_server_session): Duplicate SSL::Session before re-adding to the session store. OpenSSL 1.1.0 starts rejecting SSL_SESSION once removed by SSL_CTX_remove_session(). https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=7c2d4f3ee2547650102cd16d238125b76112ae75

- test/openssl/test_pkey_ec.rb (setup): Remove X25519 from @keys. X25519 is new in OpenSSL 1.1.0 but this is for key agreement and not for signing.

- test/openssl/test_pair.rb, test/openssl/test_ssl.rb, test/openssl/utils.rb: Set security level to 0 when using aNULL cipher suites.

- test/openssl/utils.rb: Use 1024 bits DSA key for client certificates.

- test/openssl/test_engine.rb: Run each test in separate process. We can no longer cleanup engines explicitly as ENGINE_cleanups() was removed. https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=6d4fb1d59e61aacefa25edc4fe5acfe1ac93f743

- ext/openssl/ossl_engine.c (ossl_engine_s_cleanup): Add a note to the RDoc for Engine.cleanup.

- ext/openssl/lib/openssl/digest.rb: Don't define constants for DSS, DSS1 and SHA(-0) when using with OpenSSL 1.1.0. They are removed.


11/27/2020 14/26
This fixes make test-all TESTS=openssl with OpenSSL master.

- test/openssl/test_x509name.rb: Don't register OID for 'emailAddress' and 'serialNumber'. A recent change in OpenSSL made OBJ_create() reject an already existing OID. They were needed to run tests with OpenSSL 0.9.6 which is now unsupported. [ruby-core:75225] [Feature #12324]

- test/openssl/test_ssl_session.rb (test_server_session): Duplicate SSL::Session before re-adding to the session store. OpenSSL 1.1.0 starts rejecting SSL_SESSION once removed by SSL_CTX_remove_session(). [ruby-core:75225] [Feature #12324]

- test/openssl/test_pkey_ec.rb (setup): Remove X25519 from @keys. X25519 is new in OpenSSL 1.1.0 but this is for key agreement and not for signing.

- test/openssl/test_pair.rb, test/openssl/test_ssl.rb, test/openssl/utils.rb: Set security level to 0 when using aNULL cipher suites.

- test/openssl/utils.rb: Use 1024 bits DSA key for client certificates.

- test/openssl/test_engine.rb: Run each test in separate process. We can no longer cleanup engines explicitly as ENGINE_cleanup() was removed. [ruby-core:75225] [Feature #12324]

- ext/openssl/ossl_engine.c (ossl_engine_s_cleanup): Add a note to the RDoc for Engine.cleanup.

- ext/openssl/lib/openssl/digest.rb: Don't define constants for DSS, DSS1 and SHA(-0) when using with OpenSSL 1.1.0. They are removed.


Revision 55314 - 06/07/2016 12:20 PM - rhe
openssl: adjust tests for OpenSSL 1.1.0

This fixes make test-all TESTS=openssl with OpenSSL master.

- test/openssl/test_x509name.rb: Don't register OID for 'emailAddress' and 'serialNumber'. A recent change in OpenSSL made OBJ_create() reject an already existing OID. They were needed to run tests with OpenSSL 0.9.6 which is now unsupported. [ruby-core:75225] [Feature #12324]

- test/openssl/test_ssl_session.rb (test_server_session): Duplicate SSL::Session before re-adding to the session store. OpenSSL 1.1.0 starts rejecting SSL_SESSION once removed by SSL_CTX_remove_session(). [ruby-core:75225] [Feature #12324]

- test/openssl/test_pkey_ec.rb (setup): Remove X25519 from @keys. X25519 is new in OpenSSL 1.1.0 but this is for key agreement and not for signing.

- test/openssl/test_pair.rb, test/openssl/test_ssl.rb, test/openssl/utils.rb: Set security level to 0 when using aNULL cipher suites.

- test/openssl/utils.rb: Use 1024 bits DSA key for client certificates.

- test/openssl/test_engine.rb: Run each test in separate process.
We can no longer cleanup engines explicitly as ENGINE_cleanup() was removed. 
https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=6d4fb1d59e61aacefa25edc4fe5acfe1ac93f743

- ext/openssl/ossl_engine.c (ossl_engine_s_cleanup): Add a note to the RDoc for Engine.cleanup.
- ext/openssl/lib/openssl/digest.rb: Don't define constants for DSS, DSS1 and SHA-0) when using with OpenSSL 1.1.0. They are removed.

Revision 06552d56 - 06/07/2016 12:29 PM - rhe
open-uri: regenerate server certificates used in tests

- test/open-uri/test_ssl.rb: Regenerate test certificates. The test CA certificate was incorrectly generated. A CA certificate must have the basic constraints extension with CA bit set to TRUE. OpenSSL <= 1.0.2 allowed the error when the certificate is in the trusted store but OpenSSL 1.1.0 no longer does. [ruby-core:75225] [Feature #12324]
git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55315 b2dd03c6-39d4-4d8f-98ff-823fe69b080e

Revision 55315 - 06/07/2016 12:29 PM - rhenium (Kazuki Yamaguchi)
open-uri: regenerate server certificates used in tests

- test/open-uri/test_ssl.rb: Regenerate test certificates. The test CA certificate was incorrectly generated. A CA certificate must have the basic constraints extension with CA bit set to TRUE. OpenSSL <= 1.0.2 allowed the error when the certificate is in the trusted store but OpenSSL 1.1.0 no longer does. [ruby-core:75225] [Feature #12324]
Revision 55315 - 06/07/2016 12:29 PM - rhe
open-uri: regenerate server certificates used in tests

- test/open-uri/test_ssl.rb: Regenerate test certificates. The test CA certificate was incorrectly generated. A CA certificate must have the basic constraints extension with CA bit set to TRUE. OpenSSL <= 1.0.2 allowed the error when the certificate is in the trusted store but OpenSSL 1.1.0 no longer does. [ruby-core:75225] [Feature #12324]

Revision 55315 - 06/07/2016 12:29 PM - rhe
open-uri: regenerate server certificates used in tests

- test/open-uri/test_ssl.rb: Regenerate test certificates. The test CA certificate was incorrectly generated. A CA certificate must have the basic constraints extension with CA bit set to TRUE. OpenSSL <= 1.0.2 allowed the error when the certificate is in the trusted store but OpenSSL 1.1.0 no longer does. [ruby-core:75225] [Feature #12324]

Revision 55315 - 06/07/2016 12:29 PM - rhe
open-uri: regenerate server certificates used in tests

- test/open-uri/test_ssl.rb: Regenerate test certificates. The test CA certificate was incorrectly generated. A CA certificate must have the basic constraints extension with CA bit set to TRUE. OpenSSL <= 1.0.2 allowed the error when the certificate is in the trusted store but OpenSSL 1.1.0 no longer does. [ruby-core:75225] [Feature #12324]

Revision b9051665 - 06/07/2016 12:29 PM - rhe
rubylems: regenerate server certificates used in tests

- test/rubylems/*(.cert,.cert_32).pem: Regenerate test certificates for OpenSSL 1.1.0. This is already in upstream. 
https://github.com/rubylems/rubylems/commit/9be5c53939440a61c46ba73cfffbeb65fcfad72be [ruby-core:75225] [Feature #12324]
git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55317 b2dd03c6-39d4-4d8f-98ff-823fe69b080e

Revision 55317 - 06/07/2016 12:29 PM - rhenium (Kazuki Yamaguchi)
rubylems: regenerate server certificates used in tests

- test/rubylems/*(.cert,.cert_32).pem: Regenerate test certificates for OpenSSL 1.1.0. This is already in upstream. 
https://github.com/rubylems/rubylems/commit/9be5c53939440a61c46ba73cfffbeb65fcfad72be [ruby-core:75225] [Feature #12324]
Revision 55317 - 06/07/2016 12:29 PM - rhe
rubygems: regenerate server certificates used in tests

- test/rubygems/*_{cert,cert_32}.pem: Regenerate test certificates for OpenSSL 1.1.0. This is already in upstream.  
  https://github.com/rubygems/rubygems/commit/9be5c53939440a61c4bba73cfffbeb5cfad72be [ruby-core:75225] [Feature #12324]

Revision 55317 - 06/07/2016 12:29 PM - rhe
rubygems: regenerate server certificates used in tests

- test/rubygems/*_{cert,cert_32}.pem: Regenerate test certificates for OpenSSL 1.1.0. This is already in upstream.  
  https://github.com/rubygems/rubygems/commit/9be5c53939440a61c4bba73cfffbeb5cfad72be [ruby-core:75225] [Feature #12324]

Revision 55317 - 06/07/2016 12:29 PM - rhe
rubygems: regenerate server certificates used in tests

- test/rubygems/*_{cert,cert_32}.pem: Regenerate test certificates for OpenSSL 1.1.0. This is already in upstream.  
  https://github.com/rubygems/rubygems/commit/9be5c53939440a61c4bba73cfffbeb5cfad72be [ruby-core:75225] [Feature #12324]

Revision accd3074 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55335 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55335 - 06/09/2016 06:03 AM - rhenium (Kazuki Yamaguchi)
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]

Revision 55335 - 06/09/2016 06:03 AM - rhe
openssl: fix build with OpenSSL 1.1.0 and no pkg-config

- ext/openssl/extconf.rb: Check for CRYPTO_malloc() and SSL_new(). OpenSSL_add_all_digests() and SSL_library_init() are deprecated and converted to macros in OpenSSL 1.1.0. [ruby-core:75225] [Feature #12324]
Revision 55344 - 06/09/2016 12:42 PM - rhenium (Kazuki Yamaguchi)
openssl: use ASN1_ENUMERATED_to_BN() if needed

- ext.openssl/ossl_asn1.c (asn1integer_to_num): Use
  ASN1_ENUMERATED_to_BN() to convert an ASN1_ENUMERATED to a BN.
  Starting from OpenSSL 1.1.0, ASN1_INTEGER_to_BN() rejects
  non-ASN1_INTEGER objects. The format of INTEGER and ENUMERATED are
  almost identical so they behaved in the same way in OpenSSL <= 1.0.2.
  
  [ruby-core:75225] [Feature #12324]
  
- test/openssl/test_asn1.rb (test_decode_enuminated): Test that it
  works.

Revision 55344 - 06/09/2016 12:42 PM - rhenium (Kazuki Yamaguchi)
openssl: use ASN1_ENUMERATED_to_BN() if needed

- ext.openssl/ossl_asn1.c (asn1integer_to_num): Use
  ASN1_ENUMERATED_to_BN() to convert an ASN1_ENUMERATED to a BN.
  Starting from OpenSSL 1.1.0, ASN1_INTEGER_to_BN() rejects
  non-ASN1_INTEGER objects. The format of INTEGER and ENUMERATED are
  almost identical so they behaved in the same way in OpenSSL <= 1.0.2.
  
  [ruby-core:75225] [Feature #12324]
  
- test/openssl/test_asn1.rb (test_decode_enuminated): Test that it
  works.

Revision 55344 - 06/09/2016 12:42 PM - rhenium (Kazuki Yamaguchi)
openssl: use ASN1_ENUMERATED_to_BN() if needed

- ext.openssl/ossl_asn1.c (asn1integer_to_num): Use
  ASN1_ENUMERATED_to_BN() to convert an ASN1_ENUMERATED to a BN.
  Starting from OpenSSL 1.1.0, ASN1_INTEGER_to_BN() rejects
  non-ASN1_INTEGER objects. The format of INTEGER and ENUMERATED are
  almost identical so they behaved in the same way in OpenSSL <= 1.0.2.
  
  [ruby-core:75225] [Feature #12324]
  
- test/openssl/test_asn1.rb (test_decode_enuminated): Test that it
  works.

Revision 55344 - 06/09/2016 12:42 PM - rhenium (Kazuki Yamaguchi)
openssl: use ASN1_ENUMERATED_to_BN() if needed

- ext.openssl/ossl_asn1.c (asn1integer_to_num): Use
  ASN1_ENUMERATED_to_BN() to convert an ASN1_ENUMERATED to a BN.
  Starting from OpenSSL 1.1.0, ASN1_INTEGER_to_BN() rejects
  non-ASN1_INTEGER objects. The format of INTEGER and ENUMERATED are
  almost identical so they behaved in the same way in OpenSSL <= 1.0.2.
  
  [ruby-core:75225] [Feature #12324]
  
- test/openssl/test_asn1.rb (test_decode_enuminated): Test that it
  works.

Revision aefb79c2 - 06/12/2016 04:48 AM - rhenium (Kazuki Yamaguchi)
openssl: avoid test failure in test_engine.rb

- test/openssl/test_engine.rb (test_openssl_engine_builtin, test_openssl_engine_by_id_string): Skip test if 'openssl' engine is already loaded. And
  test the number increased by Engine.load [_by_id,], not the total count of loaded engines. Previously, we called OpenSSL::Engine.cleanup every
time running a test case, but we no longer can do it. [ruby-core:75225] [Feature #12324]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@55387 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 55387 - 06/12/2016 04:48 AM - rhenium (Kazuki Yamaguchi)
openssl: avoid test failure in test_engine.rb

- test/openssl/test_engine.rb (test_openssl_engine_builtin, test_openssl_engine_by_id_string): Skip test if 'openssl' engine is already loaded. And test the number increased by Engine.load[by_id,], not the total count of loaded engines. Previously, we called OpenSSL::Engine.cleanup every time running a test case, but we no longer can do it. [ruby-core:75225] [Feature #12324]

Revision 55387 - 06/12/2016 04:48 AM - rhe
openssl: avoid test failure in test_engine.rb

- test/openssl/test_engine.rb (test_openssl_engine_builtin, test_openssl_engine_by_id_string): Skip test if 'openssl' engine is already loaded. And test the number increased by Engine.load[by_id,], not the total count of loaded engines. Previously, we called OpenSSL::Engine.cleanup every time running a test case, but we no longer can do it. [ruby-core:75225] [Feature #12324]

Revision 55387 - 06/12/2016 04:48 AM - rhe
openssl: avoid test failure in test_engine.rb

- test/openssl/test_engine.rb (test_openssl_engine_builtin, test_openssl_engine_by_id_string): Skip test if 'openssl' engine is already loaded. And test the number increased by Engine.load[by_id,], not the total count of loaded engines. Previously, we called OpenSSL::Engine.cleanup every time running a test case, but we no longer can do it. [ruby-core:75225] [Feature #12324]

Revision 55387 - 06/12/2016 04:48 AM - rhe
openssl: avoid test failure in test_engine.rb

- test/openssl/test_engine.rb (test_openssl_engine_builtin, test_openssl_engine_by_id_string): Skip test if 'openssl' engine is already loaded. And test the number increased by Engine.load[by_id,], not the total count of loaded engines. Previously, we called OpenSSL::Engine.cleanup every time running a test case, but we no longer can do it. [ruby-core:75225] [Feature #12324]

Revision 8dd0a046 - 06/14/2016 12:35 PM - rhe
openssl: fix accessor functions for RSA and DH in openssl_missing.h

- ext/openssl/openssl_missing.h (DH_set0_pqg, RSA_set0_key): DH_set0_pqg() allows 'q' to be NULL. Fix a typo in RSA_set0_key(). Fixes r55285. [ruby-core:75225] [Feature #12324]

Revision 55408 - 06/14/2016 12:35 PM - rhenium (Kazuki Yamaguchi)
openssl: fix accessor functions for RSA and DH in openssl_missing.h

- ext/openssl/openssl_missing.h (DH_set0_pqg, RSA_set0_key): DH_set0_pqg() allows 'q' to be NULL. Fix a typo in RSA_set0_key(). Fixes r55285. [ruby-core:75225] [Feature #12324]

Revision 55408 - 06/14/2016 12:35 PM - rhe
openssl: fix accessor functions for RSA and DH in openssl_missing.h

- ext/openssl/openssl_missing.h (DH_set0_pqg, RSA_set0_key): DH_set0_pqg() allows 'q' to be NULL. Fix a typo in RSA_set0_key(). Fixes r55285. [ruby-core:75225] [Feature #12324]

Revision 55408 - 06/14/2016 12:35 PM - rhe
openssl: fix accessor functions for RSA and DH in openssl_missing.h

- ext/openssl/openssl_missing.h (DH_set0_pqg, RSA_set0_key): DH_set0_pqg() allows 'q' to be NULL. Fix a typo in RSA_set0_key(). Fixes r55285. [ruby-core:75225] [Feature #12324]

Revision 55408 - 06/14/2016 12:35 PM - rhe
openssl: fix accessor functions for RSA and DH in openssl_missing.h

- ext/openssl/openssl_missing.h (DH_set0_pqg, RSA_set0_key): DH_set0_pqg() allows 'q' to be NULL. Fix a typo in RSA_set0_key(). Fixes r55285. [ruby-core:75225] [Feature #12324]

Revision 31388c4a - 06/19/2016 05:31 AM - rhe
openssl: add 'const's required in OpenSSL master

- ext/openssl/ssl_pkey.h, ext/openssl/ssl_pkey_dh.c, ext/openssl/ssl_pkey_dsa.c, ext/openssl/ssl_pkey_rsa.c: A few days ago, OpenSSL changed [DH,DSA,RSA]get0() to take const BIGNUM *. [ruby-core:75225] [Feature #12324]
Revision 55450 - 06/19/2016 05:31 AM - rhenium (Kazuki Yamaguchi)

openssl: add 'const's required in OpenSSL master

- ext/openssl/ossl_pkey.h, ext/openssl/ossl_pkey_dh.c, ext/openssl/ossl_pkey_dsa.c, ext/openssl/ossl_pkey_rsa.c: A few days ago, OpenSSL changed {DH,DSA,RSA}get0() to take const BIGNUM *. https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=fd809cddbd6e32b6b67b68c59f6d55fbed7a9327 [ruby-core:75225] [Feature #12324]

Revision 55450 - 06/19/2016 05:31 AM - rhe

openssl: add 'const's required in OpenSSL master

- ext/openssl/ossl_pkey.h, ext/openssl/ossl_pkey_dh.c, ext/openssl/ossl_pkey_dsa.c, ext/openssl/ossl_pkey_rsa.c: A few days ago, OpenSSL changed {DH,DSA,RSA}get0() to take const BIGNUM *. https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=fd809cddbd6e32b6b67b68c59f6d55fbed7a9327 [ruby-core:75225] [Feature #12324]

Revision 55450 - 06/19/2016 05:31 AM - rhe

openssl: add 'const's required in OpenSSL master

- ext/openssl/ossl_pkey.h, ext/openssl/ossl_pkey_dh.c, ext/openssl/ossl_pkey_dsa.c, ext/openssl/ossl_pkey_rsa.c: A few days ago, OpenSSL changed {DH,DSA,RSA}get0() to take const BIGNUM *. https://git.openssl.org/gitweb/?p=openssl.git;a=commit;h=fd809cddbd6e32b6b67b68c59f6d55fbed7a9327 [ruby-core:75225] [Feature #12324]

Revision 55460 - 06/20/2016 03:54 AM - rhe

rubygems: fix test for OpenSSL 1.1.0

- test/rubygems/test_gem_remote_fetcher.rb: OpenSSL::PKey::DH#priv_key= is not defined when ext/openssl is built with OpenSSL 1.1.0. https://github.com/rubygems/rubygems/pull/1648 [ruby-core:75225] [Feature #12324]
I did have openssl-pre5 installed, which failed, and I installed from openssl master and rebuilt ruby again, did an rvm mount on the results, and I got:

```
ext-ruby-2.4-ssl :004 > OpenSSL::PKey::EC.builtin_curves.include?(['X25519','X25519'])
=> true
```

so that totally looks right, and I now can play with the new IRTF CFRG curve...

**#2 - 05/04/2016 10:00 PM - mcr (Michael Richardson)**

Please see gist: [https://gist.github.com/mcr/73e10792abd529c5f53562ed275b7c1b](https://gist.github.com/mcr/73e10792abd529c5f53562ed275b7c1b)

(I could remove the rake/rails stuff around it if you like)

with X25519 selected as the curve, I get:

```
% rake highway:selfsigned
rake aborted!
OpenSSL::X509::CertificateError: EVP lib
/corp/projects/pandora/highway/lib/tasks/cert.rake:53:inblock (2 levels) in ''
/home/mcr/.rvm/gems/ext-ruby-2.4-ssl/bin/ruby_executable_hooks:15:in eval'
/home/mcr/.rvm/gems/ext-ruby-2.4-ssl/bin/ruby_executable_hooks:15:in'
Tasks: TOP => highway:selfsigned
```

With the secp112r1 curve, I get a good certificate generated:

```
% openssl x509 -in db/cert/vendor_secp112r1.crt -noout -text
Certificate:
Data:
Version: 3 (0x2)
Serial Number: 1 (0x1)
Signature Algorithm: ecdsa-with-SHA256
...
ASN1 OID: secp112r1
```

I'm investigating generating the certificate with the openssl ca driver, to help isolate it as an openssl vs ruby integration issue.

**#3 - 05/05/2016 03:04 AM - rhenium (Kazuki Yamaguchi)**

It's because X25519 is an algorithm for key exchange. Ed25519, the corresponding (sharing the elliptic curve) signature scheme, is not yet implemented in OpenSSL. There are tickets but it looks like nobody is working on it at the moment.

[https://rt.openssl.org/Ticket/Display.html?id=4077](https://rt.openssl.org/Ticket/Display.html?id=4077)
[https://github.com/openssl/openssl/issues/487](https://github.com/openssl/openssl/issues/487)

**#4 - 05/05/2016 10:04 AM - rhenium (Kazuki Yamaguchi)**

- File patches-v2.tar.gz added
- File interdiff-v1-v2.patch added

I made v2 of the patch:

[https://github.com/ruby/ruby/compare/trunk...rhenium:feature/openssl-110-v2](https://github.com/ruby/ruby/compare/trunk...rhenium:feature/openssl-110-v2)

Changes:

- Re-ordered commits in more meaningful order (and squashed some small commits).
  
  I'm so sorry if you already started reviewing. I also attach the interdiff (this is small).

- When using with an older OpenSSL, make OpenSSL::SSL::SSLContext#security_level= raise NotImplementedError if the argument is not 0.

The patches 01-05 are small cleanups/fixes.

- [01/35] ext/openssl: check if SSL_CTX_clear_options() is available
- [02/35] ext/openssl: fix ex_data handling for X509_STORE
- [03/35] ext/openssl: always use our implementation of SSL_SESSION_cmp()
- [04/35] ext/openssl: no need to check OPENSSL_FIPS in extconf.rb
- [05/35] ext/openssl: remove 'extern "C" { }' blocks

06 drops OpenSSL -0.9.7 support.

- [06/35] ext/openssl: drop support for OpenSSL 0.9.6/0.9.7
Then 07 removes OPENSSL_NO_HMAC support which has been broken for years.

- [07/35] ext/openssl: drop support for OPENSSL_NO_HMAC

08-14 handles removed / deprecated functions.

- [08/35] ext/openssl: include openssl/asn1.h instead of openssl/asn1_mac.h
- [09/35] ext/openssl: avoid deprecated BN primes functions
- [10/35] ext/openssl: disable OpenSSL::Random_pseudo_bytes if deprecated
- [11/35] ext/openssl: d2i_ASN1_BOOLEAN is removed
- [12/35] ext/openssl: avoid using deprecated protocol version specific methods
- [13/35] ext/openssl: SSL_state() is removed
- [14/35] ext/openssl: setting locking callbacks is no longer needed

15-24 fix compilation errors by replacing direct struct accesses with accessor functions.

- [15/35] ext/openssl: use EVP_CIPHER_CTX_[new,free] to allocate EVP_CIPHER_CTX
- [16/35] ext/openssl: use HMAC_CTX_[new,free,reset] to allocate HMAC_CTX
- [17/35] ext/openssl: use EVP_MD_CTX_new() to allocate EVP_MD_CTX
- [18/35] ext/openssl: OCSP_SINGLERESP and OCSP_CERTID are also made opaque
- [19/35] ext/openssl: BIGNUM and BN_GENCB is made opaque
- [20/35] ext/openssl: SSL_SESSION is made opaque
- [21/35] ext/openssl: use _up_ref() functions
- [22/35] ext/openssl: X509* are made opaque
- [25/35] ext/openssl: use SSL_CTX_get_cert()s()
- [24/35] ext/openssl: EVP_PKEY, DH, RSA, EC_KEY are made opaque

Now it compiles. The patch 24 (?) and 25-28 contain Ruby-land changes.

- [25/35] ext/openssl: add SSLContext#security_level, #security_level=
- [26/35] ext/openssl: add SSLContext#set_ecdh_curves
- [27/35] ext/openssl: remove SHA, DSS, DSS1 if using OpenSSL 1.1.0
- [28/35] ext/openssl: ENGINE.cleanup is no-op in OpenSSL 1.1.0

The next is TODO, I think this is a bug in OpenSSL.

- [29/35] ext/openssl: avoid SEGV on Cipher.new("ChaCha20-Poly1305")

30-35 fixes some test cases that are affected by OpenSSL's changes.

- [30/35] test/openssl: DSA256 is prohibited with security_level=1
- [31/35] test/openssl: X25519 doesn't support signing
- [32/35] test/openssl: fix test_server_session to dup the session
- [33/35] test/openssl: don't test default session timeout
- [34/35] test/open-uri: regenerate test certificates
- [35/35] test/rubygems: regenerate certificates

#5 - 05/09/2016 04:24 AM - naruse (Yui NARUSE)
- Status changed from Open to Assigned
- Assignee set to matz (Yukihiro Matsumoto)

I think you become a committer and commit it by yourself. (and maintain it)

#6 - 05/09/2016 07:32 AM - zzak (Zachary Scott)
I'm happy for your help with Ruby's OpenSSL!

This is a rather large change, and would like to see it land in the (To Be Released) openssl gem:
https://github.com/ruby/openssl

If you're interested, I can give you commit to the openssl gem repo as well.

#7 - 05/09/2016 07:39 AM - naruse (Yui NARUSE)

Zachary Scott wrote:

I'm happy for your help with Ruby's OpenSSL!

This is a rather large change, and would like to see it land in the (To Be Released) openssl gem:
https://github.com/ruby/openssl

If you're interested, I can give you commit to the openssl gem repo as well.
zzak (Zachary Scott) Could you explain him (or add explanation to README.rd of ruby/openssl) the relation and status between openssl gem and Ruby repo.

#8 - 05/09/2016 01:39 PM - rhenium (Kazuki Yamaguchi)
Thanks! I'd be happy to maintain ext/openssl.
I have read #5481 and #9612 before, so I know that gemifying ext/openssl is in progress.

#9 - 05/10/2016 05:29 AM - vo.x (Vit Ondruch)

even 0.9.8/1.0.0 series are no longer supported[1]

If I am not mistaken, 0.9.8 is still part of RHEL5/CentOS5 and they are still supported, although not by upstream.

#10 - 05/10/2016 06:45 AM - rhenium (Kazuki Yamaguchi)
Vit Ondruch wrote:

even 0.9.8/1.0.0 series are no longer supported[1]

If I am not mistaken, 0.9.8 is still part of RHEL5/CentOS5 and they are still supported, although not by upstream.

Yes, that's exactly why I didn't drop them in this series, and it didn't reduce code greatly compared to dropping 0.9.6/0.9.7.

#11 - 05/15/2016 01:31 AM - rhenium (Kazuki Yamaguchi)
- File patches-v3.tar.gz added
- File interdiff-v2-v3.patch added

Patch v3. It looks like OpenSSL 1.1.0 is delaying...

https://github.com/ruby/ruby/compare/trunk...rhenium:feature/openssl-110-v3

Changes:
- Rebased to apply atop trunk.
- [03/36] is new: I noticed OpenSSL::X509::Attribute#value= segfaults if the passed argument is bad.
- Fixed my broken [v2 22/35]. It was not covered by test.

#12 - 05/15/2016 02:58 AM - rhenium (Kazuki Yamaguchi)
matz (Yukihiro Matsumoto)

Can I have SVN access?
I'd love to help improving/maintaining Ruby and ext/openssl. Several of my patches including openssl-related and non-openssl-related, have landed with the help of nobu (Nobuyoshi Nakada) and @nurse. I was planning to send more patches, but it would be nice if I can commit without bothering other committers.

#13 - 05/15/2016 03:24 AM - hsbt (Hiroshi SHIBATA)
I added this issue to agenda of developer meeting at next week.

https://bugs.ruby-lang.org/projects/ruby/wiki/DevelopersMeeting20160517Japan

Please wait a few days.

#14 - 05/15/2016 03:51 AM - rhenium (Kazuki Yamaguchi)
Hiroshi SHIBATA wrote:
I added this issue to agenda of developer meeting at next week.

https://bugs.ruby-lang.org/projects/ruby/wiki/DevelopersMeeting20160517Japan

Please wait a few days.
Thanks!

#15 - 05/17/2016 05:39 AM - matz (Yukihiro Matsumoto)
OK, you can be a committer. Follow the instruction in:


Matz.

#16 - 05/17/2016 10:38 AM - rhenium (Kazuki Yamaguchi)

Thanks, I will.

#17 - 05/28/2016 09:43 AM - hsbt (Hiroshi SHIBATA)

- Assignee changed from matz (Yukihiro Matsumoto) to rhenium (Kazuki Yamaguchi)

I added commit-bit to Kazuki Yamaguchi-san few weeks ago.
Please do apply your proposal by yourself.

#18 - 06/05/2016 12:38 PM - Anonymous

- Status changed from Assigned to Closed

Applied in changeset r55282.

openssl: check existence of RAND_pseudo_bytes()

- ext/openssl/extconf.rb: Check if RAND_pseudo_bytes() is usable. It is marked as deprecated in OpenSSL 1.1.0.
  [ruby-core:75225] [Feature #12324]
- ext/openssl/oss1_rand.c: Disable Random.pseudo_bytes if RAND_pseudo_bytes() is unavailable.
- test/openssl/test_random.rb: Don't test Random.pseudo_bytes if not defined.

#19 - 06/11/2016 01:04 PM - terceiro (Antonio Terceiro)

- Backport set to 2.1: UNKNOWN, 2.2: UNKNOWN, 2.3: UNKNOWN
- Tracker changed from Feature to Bug

Hi, would it be possible to have a backport of this for Ruby 2.3? Debian 9 will be released with Ruby 2.3, and we will probably need to build against OpenSSL 1.1.0 between now and then.

#20 - 06/15/2016 01:51 PM - rhenium (Kazuki Yamaguchi)

This is based on r55162 (openssl: drop OpenSSL 0.9.6/0.9.7 support) so backporting 1.1.0 support is not so simple.

Perhaps Debian should disable ext/openssl and bundle openssl gem (though not yet released)?

#21 - 06/28/2016 07:37 PM - terceiro (Antonio Terceiro)

Kazuki Yamaguchi wrote:

This is based on r55162 (openssl: drop OpenSSL 0.9.6/0.9.7 support) so backporting 1.1.0 support is not so simple.

...well, openssl 0.9.x is long gone from all supported Debian releases, so including that would not be a problem for Debian, but I understand why it could be an issue for Ruby 2.3 upstream. So if I take that, would I be able to also take the other patches and form a Debian-specific OpenSSL 1.1.0 support patch?

Perhaps Debian should disable ext/openssl and bundle openssl gem (though not yet released)?

That would be even better, but it's not clear to me what's the status of that standalone openssl package, e.g. is it a safe drop-in replacement from the...
openssl bundled with the interpreter? Do you have any pointers?

**#22 - 12/17/2016 01:32 AM - normalperson (Eric Wong)**

k@rhe.jp wrote:

Feature #12324: Support OpenSSL 1.1.0 (and drop support for 0.9.6/0.9.7)
https://bugs.ruby-lang.org/issues/12324

rhe: did you commit all of these?

I notice SSL_CTX_clear_options is still missing on an ancient CentOS 5.x machine with 0.9.8e. Thanks.

**#23 - 12/17/2016 07:42 AM - rhenium (Kazuki Yamaguchi)**

On Sat, Dec 17, 2016 at 01:31:12AM +0000, Eric Wong wrote:

k@rhe.jp wrote:

Feature #12324: Support OpenSSL 1.1.0 (and drop support for 0.9.6/0.9.7)
https://bugs.ruby-lang.org/issues/12324

rhe: did you commit all of these?

I notice SSL_CTX_clear_options is still missing on an ancient CentOS 5.x machine with 0.9.8e. Thanks.

I dropped that patch. I found the OpenSSL 0.9.8e package in RHEL/CentOS 5 has backported SSL_CTX_clear_options() as part of CVE-2009-3555 fix, and no one except them would use < 0.9.8m anymore.

**#24 - 12/17/2016 09:11 AM - normalperson (Eric Wong)**

Kazuki Yamaguchi k@rhe.jp wrote:

On Sat, Dec 17, 2016 at 01:31:12AM +0000, Eric Wong wrote:

k@rhe.jp wrote:

Feature #12324: Support OpenSSL 1.1.0 (and drop support for 0.9.6/0.9.7)
https://bugs.ruby-lang.org/issues/12324

rhe: did you commit all of these?

I notice SSL_CTX_clear_options is still missing on an ancient CentOS 5.x machine with 0.9.8e. Thanks.

I dropped that patch. I found the OpenSSL 0.9.8e package in RHEL/CentOS 5 has backported SSL_CTX_clear_options() as part of CVE-2009-3555 fix, and no one except them would use < 0.9.8m anymore.

Thanks for the response. I'll work with my sysadmins and see if they're willing to upgrade OpenSSL; but they may not :<

If they are not, would you be willing to add SSL_CTX_clear_options? I could carry the patch myself, but maybe there are other ancient places who don't patch :<

Fwiw, I recommend using functions with custom name to emulate missing functions, and #define to the official name:

```c
static unsigned long
rb_SSL_CTX_clear_options(SSL_CTX *ctx, unsigned long op)
{
    return ctx->options &= ~op;
}
#define SSL_CTX_clear_options(ctx,op) rb_SSL_CTX_clear_options((ctx),(op))
```
It makes debugging easier since the internal symbol name won't conflict with the official version.

Thanks.

#25 - 12/17/2016 04:21 PM - rhenium (Kazuki Yamaguchi)
On Sat, Dec 17, 2016 at 09:09:49AM +0000, Eric Wong wrote:

Kazuki Yamaguchi k@rhe.jp wrote:

On Sat, Dec 17, 2016 at 01:31:12AM +0000, Eric Wong wrote:

k@rhe.jp wrote:

Feature #12324: Support OpenSSL 1.1.0 (and drop support for 0.9.6/0.9.7)
https://bugs.ruby-lang.org/issues/12324
rhe: did you commit all of these?

I notice SSL_CTX_clear_options is still missing on an ancient CentOS 5.x machine with 0.9.8e. Thanks.

I dropped that patch. I found the OpenSSL 0.9.8e package in RHEL/CentOS 5 has backported SSL_CTX_clear_options() as part of CVE-2009-3555 fix, and no one except them would use < 0.9.8m anymore.

Thanks for the response. I'll work with my sysadmins and see if they're willing to upgrade OpenSSL; but they may not :<

If they are not, would you be willing to add SSL_CTX_clear_options? I could carry the patch myself, but maybe there are other ancient places who don't patch :<

I'm fine with adding if it helps. I don't want to imagine such systems with 6 years unpatched OpenSSL, though.

I've added to the GitHub repository now:

https://github.com/ruby/openssl/commit/fd2ea0c41d55f3d0b34edb1d25c673bc004f6913

Will import to trunk in a few days together with a few more minor bug fixes.

#26 - 12/19/2016 10:21 PM - normalperson (Eric Wong)
Kazuki Yamaguchi k@rhe.jp wrote:

I'm fine with adding if it helps. I don't want to imagine such systems with 6 years unpatched OpenSSL, though.

I've added to the GitHub repository now:

https://github.com/ruby/openssl/commit/fd2ea0c41d55f3d0b34edb1d25c673bc004f6913

Thanks! Yeah, security is an afterthought, for me :<

Files

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>patches-v1.tar.gz</td>
<td>81.4 KB</td>
<td>04/27/2016</td>
<td>rhenium (Kazuki Yamaguchi)</td>
</tr>
<tr>
<td>patches-v2.tar.gz</td>
<td>79.9 KB</td>
<td>05/05/2016</td>
<td>rhenium (Kazuki Yamaguchi)</td>
</tr>
<tr>
<td>interdiff-v1-v2.patch</td>
<td>13.4 KB</td>
<td>05/05/2016</td>
<td>rhenium (Kazuki Yamaguchi)</td>
</tr>
<tr>
<td>patches-v3.tar.gz</td>
<td>80.1 KB</td>
<td>05/14/2016</td>
<td>rhenium (Kazuki Yamaguchi)</td>
</tr>
<tr>
<td>interdiff-v2-v3.patch</td>
<td>15.7 KB</td>
<td>05/14/2016</td>
<td>rhenium (Kazuki Yamaguchi)</td>
</tr>
</tbody>
</table>