TLS 暗号設定 暗号スイートの設定例

令和 2 年 7 月

独立行政法人 情報処理推進機構

目次

本書では、暗号スイートの設定を行う上での参考情報として、設定方法例を記載する。
なお､利用するバージョンやディストリビューションの違いにより､実装されている暗号スイートの種類や設定方法が異なる場合があることに留意すること。正式な取扱説明書やマニュアルを参照するとともに、一参考資料として利用されたい。

本書は以下のソフトウェアバージョンを対象として作成された。

OpenSSL 1.1.1d
GnuTLS 3.6.11.1
Apache httpd 2.4.41
lighttpd 1.4.54
nginx 1.16.1, 1.17.6
mod_gnutls 0.9.1

1. OpenSSL 系での設定例

1.1. OpenSSL の設定

1.1.1. OpenSSL 系での暗号スイートの設定例

〔TLS1.3 用暗号スイート設定文字列〕
TLS1.3 でサポートする暗号スイートは、コロン(:)で区切られた暗号スイート名を並べた文字列によって指定する。このとき、先頭に記載されたものがより高い優先順位を持つ。
以下にガイドラインに適合する TLS1.3 用の暗号スイートの設定例を示す。これはガイドラインに記載されたもののうち極力多くをサポートする設定例であるため、必要に応じて一部を削除することも可能である。

- 推奨セキュリティ型、セキュリティ例外型の設定例
  TLS_AES_128_GCM_SHA256:TLS_AES_128_CCM_SHA256:TLS_AES_128_CCM_8_SHA256:TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256

- 高セキュリティ型の設定例
  TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256:TLS_AES_128_CCM_SHA256:TLS_AES_128_CCM_8_SHA256

〔TLS1.2 以前用暗号スイート設定文字列〕
TLS1.2 以前でサポートする暗号スイートは、コロン(:)で区切られた以下の要素を並べた文字列によって暗号スイートのリストを設定する。この文字列は左から順に処理され、最終的にリストの先頭に存在する暗号スイートがより高い優先順位を持つ。

- OpenSSL 独自の暗号スイート名（1.1.3 節参照）による指定
  - 暗号スイートのリストに、順に特定の暗号スイートが追加される
- 個別の暗号スイート名に代えた「ECDHE」「ECDHE+AESGCM」といったパターン

- 暗号スイートのリストに該当する暗号スイートがすべて追加される
- 「+」によって複数のパターン名が連結された場合、それらの共通部分が対象となる
- パターン名のうち、特に混同に注意が必要なものを以下に示す
  - kRSA　RSA 鍵交換を使用する暗号スイート
    （TLS_RSA_WITH_AES_128_CBC_SHA など）
  - aRSA　RSA 認証を使用する暗号スイート
    （kRSA のものに加えて TLS_DHE_RSA_WITH_AES_128_CBC_SHA など）

- 前述の 2 つの記法に「+」「-」「!」を前置した暗号スイートのリストの操作
  - 「+」を前置した場合、リスト中の該当する暗号スイートがその時点での優先順位最下位に移動する
  - 「-」を前置した場合、リスト中の該当する暗号スイートがリストから削除される
  - 「!」を前置した場合、リスト中の該当する暗号スイートがリストから削除され、該当する暗号スイートはリストに追加することが不可能になる

以下にガイドラインの各型に適合する TLS1.2 以前用の暗号スイートの設定例を示す。これらはガイドラインに記載されたもののうち極力多くをサポートする設定例であるため、必要に応じて一部を削除することも可能である。

- パターン名による推奨セキュリティ型の設定例

```
ECDHE+AESGCM:DHE+aRSA+AESGCM:ECDHE+AESCCM:DHE+aRSA+AESCCM:+AES256:ECDHE+CHACHA20:DHE+aRSA+CHACHA20:+DHE:ECDHE+AES128:ECDHE+CAMELLIA128:ECDHE+AES:ECDHE+CAMELLIA:+ECDHE+SHA:DHE+aRSA+AES128:DHE+aRSA+CAMELLIA128:DHE+aRSA+AES:DHE+aRSA+CAMELLIA:+DHE+aRSA+SHA
```

- パターン名による高セキュリティ型の設定例

```
ECDHE+AESGCM:DHE+aRSA+AESGCM:ECDHE+AESCCM:DHE+aRSA+AESCCM:ECDHE+CHACHA20:DHE+aRSA+CHACHA20:+AES128:+DHE
```

- パターン名によるセキュリティ例外型の設定例

```
DHE+aRSA+AESGCM:ECDHE+AESGCM:DHE+aRSA+AESCCM:ECDHE+AESCCM:+AES256:DHE+aRSA+CHACHA20:ECDHE+CHACHA20:kRSA+AESGCM:kRSA+AESCCM:+kRSA+AES256:DHE+aRSA+AES128:DHE+aRSA+CAMELLIA128:+DHE+aRSA+SHA:ECDHE+AES128:ECDHE+CAMELLIA128:+ECDHE+SHA:DHE+aRSA+AES256:DHE+aRSA+CAMELLIA256:+DHE+aRSA+AES256+SHA:+DHE+aRSA+CAMELLIA256+SHA:ECDHE+AES256:ECDHE+CAMELLIA256:+ECDHE+AES256+SHA:kRSA+AES128:kRSA+CAMELLIA128:+kRSA+SHA:kRSA+AES:kRSA+CAMELLIA:+kRSA+AES256+SHA:+kRSA+CAMELLIA256+SHA
```

- 暗号スイート名による推奨セキュリティ型の設定例

```
ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-CCM:ECDHE-ECDSA-AES128-CCM8:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-CCM:ECDHE-ECDSA-AES256-CCM8:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES128-CCM:DHE-RSA-AE
```

S128-CCM8:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES256-CCM:DHE-RSA-AES256-CCM8:DHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-CAMELLIA128-SHA256:ECDHE-RSA-CAMELLIA128-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-CAMELLIA256-SHA384:ECDHE-RSA-CAMELLIA256-SHA384:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES256-SHA:ECDHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-CAMELLIA128-SHA256:DHE-RSA-AES256-SHA256:DHE-RSA-CAMELLIA256-SHA256:DHE-RSA-AES128-SHA:DHE-RSA-CAMELLIA128-SHA:DHE-RSA-AES256-SHA:DHE-RSA-CAMELLIA256-SHA

- 暗号スイート名による高セキュリティ型の設定例
  ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-CCM:ECDHE-ECDSA-AES256-CCM8:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-CCM:ECDHE-ECDSA-AES128-CCM8:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES256-CCM:DHE-RSA-AES256-CCM8:DHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES128-CCM:DHE-RSA-AES128-CCM8

- 暗号スイート名によるセキュリティ例外型の設定例
  DHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES128-CCM:ECDHE-ECDSA-AES128-CCM:DHE-RSA-AES128-CCM8:ECDHE-ECDSA-AES128-CCM8:DHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES256-CCM:ECDHE-ECDSA-AES256-CCM:DHE-RSA-AES256-CCM8:ECDHE-ECDSA-AES256-CCM8:DHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:AES128-GCM-SHA256:AES128-CCM:AES128-CCM8:AES256-GCM-SHA384:AES256-CCM:AES256-CCM8:DHE-RSA-AES128-SHA256:DHE-RSA-CAMELLIA128-SHA256:DHE-RSA-AES128-SHA:DHE-RSA-CAMELLIA128-SHA:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-CAMELLIA128-SHA256:ECDHE-RSA-CAMELLIA128-SHA256:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-CAMELLIA256-SHA256:DHE-RSA-AES256-SHA:DHE-RSA-CAMELLIA256-SHA:ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-CAMELLIA256-SHA384:ECDHE-RSA-CAMELLIA256-SHA384:ECDHE-ECDSA-AES256-SHA:ECDHE-RSA-AES256-SHA:AES128-SHA256:CAMELLIA128-SHA256:AES128-SHA:CAMELLIA128-SHA:AES256-SHA256:CAMELLIA256-SHA256:AES256-SHA:CAMELLIA256-SHA

1.1.2. 設定ファイルを用いた TLS1.3 暗号スイートの設定

OpenSSL を TLS ライブラリとして使用する一部のアプリケーションは、OpenSSL 1.1.1 より導入された TLS1.3 暗号スイートの設定を行う API に対応していない。このようなアプリケーションの TLS1.3 暗号スイートの設定を行うには、以下のような内容の openssl.cnf ファイルを作成する。

```
openssl_conf = openssl_init

[openssl_init]
ssl_conf = ssl_sect

[ssl_sect]
system_default = system_default_sect

[system_default_sect]
Ciphersuites = (TLS1.3 用暗号スイート設定文字列)[1]
```

設定を有効にしてアプリケーションを起動するためには、環境変数 OPENSSL_CONF に作成した openssl.cnf のパスを指定して実行する。例えば以下のようにして nginx を実行する。

```
OPENSSL_CONF=/etc/nginx/openssl.cnf nginx -c /etc/nginx/nginx.conf
```

1.1.3. 一般的な名称と OpenSSL での名称の対応表

ガイドラインに記載する暗号スイート名	OpenSSL での暗号スイート名表記
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	ECDHE-ECDSA-AES128-GCM-SHA256
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	ECDHE-RSA-AES128-GCM-SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	ECDHE-ECDSA-AES256-GCM-SHA384
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDHE-RSA-AES256-GCM-SHA384
TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	ECDHE-ECDSA-CHACHA20-POLY1305
TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	ECDHE-RSA-CHACHA20-POLY1305
TLS_ECDHE_ECDSA_WITH_AES_128_CCM	ECDHE-ECDSA-AES128-CCM
TLS_ECDHE_ECDSA_WITH_AES_256_CCM	ECDHE-ECDSA-AES256-CCM
TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8	ECDHE-ECDSA-AES128-CCM8
TLS_ECDHE_ECDSA_WITH_AES_256_CCM_8	ECDHE-ECDSA-AES256-CCM8
TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_GCM_SHA256	非対応
TLS_ECDHE_RSA_WITH_CAMELLIA_128_GCM_SHA256	非対応
TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_GCM_SHA384	非対応
TLS_ECDHE_RSA_WITH_CAMELLIA_256_GCM_SHA384	非対応
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	DHE-RSA-AES128-GCM-SHA256
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	DHE-RSA-AES256-GCM-SHA384
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	DHE-RSA-CHACHA20-POLY1305

[1] 1.1.1 節にて例示した TLS1.3 用の暗号スイートの設定例を記述する

TLS_DHE_RSA_WITH_AES_128_CCM	DHE-RSA-AES128-CCM
TLS_DHE_RSA_WITH_AES_256_CCM	DHE-RSA-AES256-CCM
TLS_DHE_RSA_WITH_AES_128_CCM_8	DHE-RSA-AES128-CCM8
TLS_DHE_RSA_WITH_AES_256_CCM_8	DHE-RSA-AES256-CCM8
TLS_DHE_RSA_WITH_CAMELLIA_128_GCM_SHA256	非対応
TLS_DHE_RSA_WITH_CAMELLIA_256_GCM_SHA384	非対応
TLS_RSA_WITH_AES_256_GCM_SHA384	AES256-GCM-SHA384
TLS_RSA_WITH_CAMELLIA_256_GCM_SHA384	非対応
TLS_RSA_WITH_AES_128_GCM_SHA256	AES128-GCM-SHA256
TLS_RSA_WITH_CAMELLIA_128_GCM_SHA256	非対応
TLS_RSA_WITH_AES_128_CCM	AES128-CCM
TLS_RSA_WITH_AES_256_CCM	AES256-CCM
TLS_RSA_WITH_AES_128_CCM_8	AES128-CCM8
TLS_RSA_WITH_AES_256_CCM_8	AES256-CCM8
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	ECDHE-ECDSA-AES128-SHA256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	ECDHE-RSA-AES128-SHA256
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	ECDHE-ECDSA-AES256-SHA384
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	ECDHE-RSA-AES256-SHA384
TLS_ECDHE_ECDSA_WITH_CAMELLIA_128_CBC_SHA256	ECDHE-ECDSA-CAMELLIA128-SHA256
TLS_ECDHE_RSA_WITH_CAMELLIA_128_CBC_SHA256	ECDHE-RSA-CAMELLIA128-SHA256
TLS_ECDHE_ECDSA_WITH_CAMELLIA_256_CBC_SHA384	ECDHE-ECDSA-CAMELLIA256-SHA384
TLS_ECDHE_RSA_WITH_CAMELLIA_256_CBC_SHA384	ECDHE-RSA-CAMELLIA256-SHA384
TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	DHE-RSA-AES128-SHA256
TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	DHE-RSA-AES256-SHA256
TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA256	DHE-RSA-CAMELLIA128-SHA256
TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA256	DHE-RSA-CAMELLIA256-SHA256
TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	ECDHE-ECDSA-AES128-SHA
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	ECDHE-RSA-AES128-SHA
TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	ECDHE-ECDSA-AES256-SHA
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	ECDHE-RSA-AES256-SHA
TLS_DHE_RSA_WITH_AES_128_CBC_SHA	DHE-RSA-AES128-SHA
TLS_DHE_RSA_WITH_AES_256_CBC_SHA	DHE-RSA-AES256-SHA
TLS_DHE_RSA_WITH_CAMELLIA_128_CBC_SHA	DHE-RSA-CAMELLIA128-SHA
TLS_DHE_RSA_WITH_CAMELLIA_256_CBC_SHA	DHE-RSA-CAMELLIA256-SHA
TLS_RSA_WITH_AES_128_CBC_SHA256	AES128-SHA256
TLS_RSA_WITH_AES_256_CBC_SHA256	AES256-SHA256
TLS_RSA_WITH_CAMELLIA_128_CBC_SHA256	CAMELLIA128-SHA256
TLS_RSA_WITH_CAMELLIA_256_CBC_SHA256	CAMELLIA256-SHA256
TLS_RSA_WITH_AES_128_CBC_SHA	AES128-SHA

TLS_RSA_WITH_AES_256_CBC_SHA	AES256-SHA
TLS_RSA_WITH_CAMELLIA_128_CBC_SHA	CAMELLIA128-SHA
TLS_RSA_WITH_CAMELLIA_256_CBC_SHA	CAMELLIA256-SHA

1.2. アプリケーションの設定

1.2.1. Apache+mod_ssl の設定

1.1.1 節に従い、各<VirtualHost>中の SSLCipherSuite を以下のように設定する。

SSLCipherSuite "TLS1.2 以前用暗号スイート設定文字列"
SSLCipherSuite TLSv1.3 "TLS1.3 用暗号スイート設定文字列"

1.2.2. lighttpd の設定

1.1.1 節に従い、各$SERVER["socket"]によるポート設定中の ssl.cipher-list および ssl.openssl.ssl-conf-cmd を以下のように設定する。

ssl.cipher-list = "TLS1.2 以前用暗号スイート設定文字列"
ssl.openssl.ssl-conf-cmd = ("Ciphersuites" => "TLS1.3 用暗号スイート設定文字列")

1.2.3. nginx の設定

1.1.1 節に従い、各 server 中の ssl_ciphers を以下のように設定する。nginx 1.16.1 および 1.17.6 は TLS1.3 暗号スイートの設定に対応していないため、TLS1.3 暗号スイートの設定を行う必要がある場合は、1.1.2 節の方法を用いる。

ssl_ciphers "TLS1.2 以前用暗号スイート設定文字列";

2. GnuTLS 系での設定例

2.1. GnuTLS の設定

2.1.1. プロトコルバージョンの設定

GnuTLS ではプロトコルバージョンの設定と暗号スイートの設定が一体化している。次節で示す各型の設定文字列例について、サポートする TLS プロトコルバージョンを変更する必要がある

場合はそれぞれ以下のように変更を加える。

推奨セキュリティ型および高セキュリティ型において TLS1.3 を無効にする場合、以下に示すように該当部分を変更する。

変更箇所）+VERS-TLS1.2:+VERS-TLS1.3
変更後）+VERS-TLS1.2

セキュリティ例外型において TLS1.3 を無効にする場合、以下に示すように該当部分を変更する。

変更箇所）+VERS-TLS1.0:+VERS-TLS1.1:+VERS-TLS1.2:+VERS-TLS1.3
変更後）+VERS-TLS1.0:+VERS-TLS1.1:+VERS-TLS1.2

セキュリティ例外型において TLS1.0 を無効にする場合、以下に示すように該当部分を変更する。

変更箇所）+VERS-TLS1.0:+VERS-TLS1.1:+VERS-TLS1.2:+VERS-TLS1.3
変更後）+VERS-TLS1.1:+VERS-TLS1.2:+VERS-TLS1.3

セキュリティ例外型において TLS1.0 および TLS1.1 を無効にする場合、以下に示すように該当部分を変更する。

変更箇所）+VERS-TLS1.0:+VERS-TLS1.1:+VERS-TLS1.2:+VERS-TLS1.3
変更後）+VERS-TLS1.2:+VERS-TLS1.3

2.1.2. GnuTLS の設定例

GnuTLS では個別の暗号スイートを指定することはできず、鍵交換・暗号・MAC(メッセージ認証符号)で使用するアルゴリズムをそれぞれ指定することにより、それらを組み合わせた暗号スイートが自動的に有効になる。このとき、優先順位が常に鍵交換＞暗号＞MAC の順に強く反映されるため、細かな暗号スイートの優先順位制御は不可能となっている。

以下の例は特に註釈が無い場合はガイドラインに記載された暗号スイートのうち極力多くをサポートする設定例であるため、必要に応じて一部を削除することも可能である。

- 推奨セキュリティ型の設定例

  NONE:%SERVER_PRECEDENCE:+VERS-TLS1.2:+VERS-TLS1.3:+SIGN-ALL:+COMP-NULL:+GROUP-ALL:+ECDHE-ECDSA:+ECDHE-RSA:+DHE-RSA:+AES-128-GCM:+CAMELLIA-128-GCM:+AES-128-CCM:+AES-128-CCM-8:+AES-256-GCM:+CAMELLIA-256-GCM:+AES-256-CCM:+AES-256-CCM-8:+CHACHA20-POLY1305:+AES-128-CBC:+CAMELLIA-128-CBC:+AES-256-CBC:+CAMELLIA-256-CBC:+AEAD:+SHA384:+SHA256:+SHA1

  （注）ガイドラインに示されたグループ毎の優先順位とは異なり、以下の優先順位となる。

  1. グループ A の一部(ECDHE-ECDSA)
  2. グループ C の一部(ECDHE-ECDSA)とグループ D の一部(ECDHE-ECDSA) が混在

3. グループ A の一部(ECDHE-RSA)
4. グループ C の一部(ECDHE-RSA)とグループ D の一部(ECDHE-RSA) が混在
5. グループ B
6. グループ E とグループ F が混在

- 推奨セキュリティ型の設定例（グループ A・B のみ）
  ガイドラインに示されたグループ毎の優先順位に準拠している。
  NONE:%SERVER_PRECEDENCE:+VERS-TLS1.2:+VERS-TLS1.3:+SIGN-ALL:+COMP-NULL:+GROUP-ALL:+ECDHE-ECDSA:+ECDHE-RSA:+DHE-RSA:+AES-128-GCM:+CAMELLIA-128-GCM:+AES-128-CCM:+AES-128-CCM-8:+AES-256-GCM:+CAMELLIA-256-GCM:+AES-256-CCM:+AES-256-CCM-8:+CHACHA20-POLY1305:+AEAD:+SHA384:+SHA256

- 推奨セキュリティ型の設定例（グループ A・B・C・E のみ）
  NONE:%SERVER_PRECEDENCE:+VERS-TLS1.2:+VERS-TLS1.3:+SIGN-ALL:+COMP-NULL:+GROUP-ALL:+ECDHE-ECDSA:+ECDHE-RSA:+DHE-RSA:+AES-128-GCM:+CAMELLIA-128-GCM:+AES-128-CCM:+AES-128-CCM-8:+AES-256-GCM:+CAMELLIA-256-GCM:+AES-256-CCM:+AES-256-CCM-8:+CHACHA20-POLY1305:+AES-128-CBC:+CAMELLIA-128-CBC:+AES-256-CBC:+CAMELLIA-256-CBC:+AEAD:+SHA384:+SHA256
  （注）ガイドラインに示されたグループ毎の優先順位とは異なり、以下の優先順位となる。
  1. グループ A の一部(ECDHE-ECDSA)
  2. グループ C の一部(ECDHE-ECDSA)
  3. グループ A の一部(ECDHE-RSA)
  4. グループ C の一部(ECDHE-RSA)
  5. グループ B
  6. グループ E

- 高セキュリティ型の設定例
  ガイドラインに示されたグループ毎の優先順位に準拠している。
  NONE:%SERVER_PRECEDENCE:+VERS-TLS1.2:+VERS-TLS1.3:+SIGN-ALL:+COMP-NULL:+GROUP-ALL:+ECDHE-ECDSA:+ECDHE-RSA:+DHE-RSA:+AES-256-GCM:+CAMELLIA-256-GCM:+AES-256-CCM:+AES-256-CCM-8:+CHACHA20-POLY1305:+AES-128-GCM:+CAMELLIA-128-GCM:+AES-128-CCM:+AES-128-CCM-8:+AEAD:+SHA384:+SHA256

- セキュリティ例外型の設定例
  NONE:%SERVER_PRECEDENCE:+VERS-TLS1.0:+VERS-TLS1.1:+VERS-TLS1.2:+VERS-TLS1.3:+SIGN-ALL:+COMP-NULL:+GROUP-ALL:+DHE-RSA:+ECDHE-ECDSA:+ECDHE-RSA:+RSA:+AES-128-GCM:+CAMELLIA-128-GCM:+AES-128-CCM:+AES-128-CCM-8:+AES-256-GCM:+CAMELLIA-256-GCM:+AES-256-CCM:+AES-256-CCM-8:+CHACHA20-POLY1305:+AES-128-CBC:+CAMELLIA-128-CBC:+AES-256-CBC:+CAMELLIA-256-CBC:+AEAD:+SHA384:+SHA256:+SHA1
  （注）ガイドラインに示されたグループ毎の優先順位とは異なり、以下の優先順位となる。
  1. グループ X の一部(ECDHE-ECDSA)

2. グループ Z の一部(ECDHE-ECDSA)
3. グループ X の一部(ECDHE-RSA)
4. グループ Z の一部(ECDHE-RSA)
5. グループ X の一部(DHE-RSA)
6. グループ Z の一部(DHE-RSA)
7. グループ Y の一部(RSA)
8. グループ Z の一部(RSA)

- セキュリティ例外型の設定例（グループ X・グループ Y のみ）
  ガイドラインに示されたグループ毎の優先順位に準拠している。
  NONE:%SERVER_PRECEDENCE:+VERS-TLS1.0:+VERS-TLS1.1:+VERS-TLS1.2:+VERS-TLS1.3:+SIGN-ALL:+COMP-NULL:+GROUP-ALL:+DHE-RSA:+ECDHE-ECDSA:+ECDHE-RSA:+RSA:+AES-128-GCM:+CAMELLIA-128-GCM:+AES-128-CCM:+AES-128-CCM-8:+AES-256-GCM:+CAMELLIA-256-GCM:+AES-256-CCM:+AES-256-CCM-8:+CHACHA20-POLY1305:+AEAD:+SHA384:+SHA256

2.2. アプリケーションの設定

2.2.1. Apache+mod_gnutls の設定

2.1 節に従い、各<VirtualHost>中の GnuTLSPriorities を以下のように設定する。

mod_gnutls では<VirtualHost>中に複数の GnuTLSCertificateFile を指定したとしても意味を持たないため、証明書の種類（RSA, ECDSA）によって鍵交換アルゴリズムが限定されることがある。これにより、2.1 節に示した優先順位の不一致が部分的に解消するケースもある。

GnuTLSPriorities "暗号スイート設定文字列 [2]"

[2] 2.1 節で例示した暗号スイート設定文字列を記述する