Бесплатная электронная книга

УЧУСЬ

encryption

Free unaffiliated eBook created from
Stack Overflow contributors.

#encryption

Около

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: encryption

It is an unofficial and free encryption ebook created for educational purposes. All the content is extracted from Stack Overflow Documentation, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official encryption.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

глава 1: Начало работы с шифрованием

замечания

В этом разделе представлен обзор того, что такое шифрование, и почему разработчик может захотеть его использовать.

Следует также упомянуть о любых крупных предметах в рамках шифрования и ссылки на связанные темы. Поскольку документация для шифрования является новой, вам может потребоваться создать начальные версии этих связанных тем.

Examples

Что такое шифрование?

> В криптографии шифрование - это процесс кодирования сообщений или информации таким образом, что к ним могут обращаться только уполномоченные стороны.
>
> Источник: шифрование - Википедия

Прочитайте Начало работы с шифрованием онлайн:
https://riptutorial.com/ru/encryption/topic/4306/начало-работы-с-шифрованием

глава 2: Текстовое шифрование

Вступление

Современная криптография работает с байтами, а не с текстом, поэтому вывод криптографических алгоритмов - это байты. Иногда зашифрованные данные должны передаваться через текстовый носитель, и необходимо использовать двоично-безопасную кодировку.

параметры

параметр	подробности
`TE`	Текстовое кодирование. Преобразование из текста в байты. UTF-8 - общий выбор.
`BE`	Двоичное кодирование. Преобразование, которое способно обрабатывать любые произвольные данные и создавать допустимую строку. Base64 является наиболее часто используемым кодированием, а Base16 / hexadecimal - хорошим занявшим второе место. В Википедии есть список кандидатских кодировок (придерживайтесь букв «Произвольные»).

замечания

Общий алгоритм:

`Encrypt` :

- Преобразовать `InputText` в `InputBytes` через кодировку `TE` (текстовое кодирование).
- Шифровать `InputBytes` до `OutputBytes`
- Преобразование `OutputBytes` в `OutputText` через `BE` (двоичное кодирование).

`Decrypt` (назад BE и TE из `Encrypt`):

- Преобразование `InputText` в `InputBytes` помощью кодирования `BE` .
- Расшифровывать `InputBytes` в `OutputBytes`
- Преобразование `OutputBytes` в `OutputText` через `TE` .

Наиболее распространенной ошибкой является выбор «текстовой кодировки» вместо «двоичной кодировки» для `BE` , что является проблемой, если какой-либо зашифрованный байт (или любой байт IV) находится за пределами диапазона `0x20` - `0x7E` (для UTF-8 или ASCII). Поскольку «безопасный диапазон» составляет менее половины байтового

пространства, шансы на успешное кодирование текста исчезающе малы.

- Если строка после шифрования содержит `0x00` то программы на C / C ++, скорее всего, неверно интерпретируют это как конец строки.
- Если консольная программа увидит `0x08` она может стереть предыдущий символ (и управляющий код), в результате `InputText` значение `InputText` в `Decrypt` иметь неправильное значение (и неправильную длину).

Examples

C

```
internal sealed class TextToTextCryptography : IDisposable
{
    // This type is not thread-safe because it repeatedly mutates the IV property.
    private SymmetricAlgorithm _cipher;

    // The input to Encrypt and the output from Decrypt need to use the same Encoding
    // so text -> bytes -> text produces the same text.
    private Encoding _textEncoding;

    // The output text ("the wire format") needs to be the same encoding for To-The-Wire
    // and From-The-Wire.
    private Encoding _binaryEncoding;

    /// <summary>
    /// Construct a Text-to-Text encryption/decryption object.
    /// </summary>
    /// <param name="key">
    ///   The cipher key to use
    /// </param>
    /// <param name="textEncoding">
    ///   The text encoding to use, or <c>null</c> for UTF-8.
    /// </param>
    /// <param name="binaryEncoding">
    ///   The binary/wire encoding to use, or <c>null</c> for Base64.
    /// </param>
    internal TextToTextCryptography(
        byte[] key,
        Encoding textEncoding,
        Encoding binaryEncoding)
    {
        // The rest of this class can operate on any SymmetricAlgorithm, but
        // at some point you either need to pick one, or accept an input choice.
        SymmetricAlgorithm cipher = Aes.Create();

        // If the key isn't valid for the algorithm this will throw.
        // Since cipher is an Aes instance the key must be 128, 192, or 256 bits
        // (16, 24, or 32 bytes).
        cipher.Key = key;

        // These are the defaults, expressed here for clarity
        cipher.Padding = PaddingMode.PKCS7;
        cipher.Mode = CipherMode.CBC;

        _cipher = cipher;
```

```
        _textEncoding = textEncoding ?? Encoding.UTF8;

        // Allow null to mean Base64 since there's not an Encoding class for Base64.
        _binaryEncoding = binaryEncoding;
    }

    internal string Encrypt(string text)
    {
        // Because we are encrypting with CBC we need an Initialization Vector (IV).
        // Just let the platform make one up.
        _cipher.GenerateIV();
        byte[] output;

        using (ICryptoTransform encryptor = _cipher.CreateEncryptor())
        {
            if (!encryptor.CanTransformMultipleBlocks)
                throw new InvalidOperationException("Rewrite this code with CryptoStream");

            byte[] input = _textEncoding.GetBytes(text);
            byte[] encryptedOutput = encryptor.TransformFinalBlock(input, 0, input.Length);

            byte[] iv = _cipher.IV;

            // Build output as iv.Concat(encryptedOutput).ToArray();
            output = new byte[iv.Length + encryptedOutput.Length];
            Buffer.BlockCopy(iv, 0, output, 0, iv.Length);
            Buffer.BlockCopy(encryptedOutput, 0, output, iv.Length, encryptedOutput.Length);
        }

        return BytesToWire(output);
    }

    internal string Decrypt(string text)
    {
        byte[] inputBytes = WireToBytes(text);

        // Rehydrate the IV
        byte[] iv = new byte[_cipher.BlockSize / 8];
        Buffer.BlockCopy(inputBytes, 0, iv, 0, iv.Length);

        _cipher.IV = iv;

        byte[] output;

        using (ICryptoTransform decryptor = _cipher.CreateDecryptor())
        {
            if (!decryptor.CanTransformMultipleBlocks)
                throw new InvalidOperationException("Rewrite this code with CryptoStream");

            // Decrypt everything after the IV.
            output = decryptor.TransformFinalBlock(
                inputBytes,
                iv.Length,
                inputBytes.Length - iv.Length);
        }

        return _textEncoding.GetString(output);
    }

    private string BytesToWire(byte[] bytes)
    {
```

```
        if (_binaryEncoding != null)
        {
            return _binaryEncoding.GetString(bytes);
        }

        // Let null _binaryEncoding be Base64.
        return Convert.ToBase64String(bytes);
    }

    private byte[] WireToBytes(string wireText)
    {
        if (_binaryEncoding != null)
        {
            return _binaryEncoding.GetBytes(wireText);
        }

        // Let null _binaryEncoding be Base64.
        return Convert.FromBase64String(wireText);
    }

    public void Dispose()
    {
        _cipher.Dispose();
        _cipher = null;
    }
}
```

Прочитайте Текстовое шифрование онлайн: https://riptutorial.com/ru/encryption/topic/10179/текстовое-шифрование

глава 3: Цезарский шифр

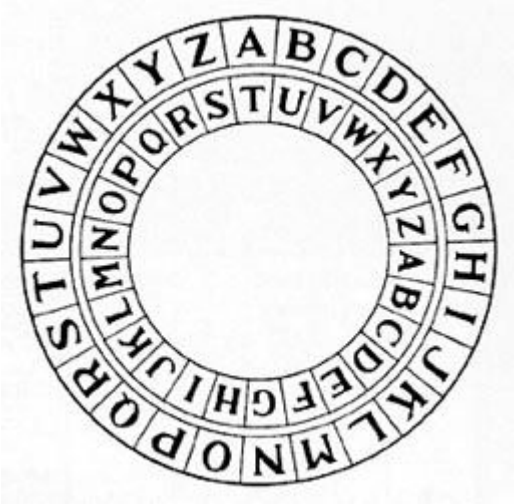
Вступление

Цезарский шифр является одним из самых простых и широко известных методов шифрования. Имена происходят от Юлиуса Цезаря, который, по словам Светония, использовал его со сдвигом втрое для защиты посланий военного значения

Examples

шифрование

Захват происходит, заменяя каждую букву альфабетом другой буквой. Клавиши могут быть показаны с помощью этого круга. Здесь используется сдвиг в 8 символов.



Здесь A заменяется на T, B на U, C на V и т. Д. Таким образом, следующая строка будет зашифрована:

оригинал	Зашифрованные
ПРИВЕТ, МИР	AXEEH PHKEW

Дешифрирование

Описания происходит по тому же самому, что и в примере шифрования. Пример:

Зашифрованные	оригинал
AXEEH PHKEW	ПРИВЕТ, МИР

Взлом

Цезарные шифры легко взломать. Если вы знаете, что зашифрованный A равен H, а P равен I, все с одним и тем же ключом шифрования могут быть зашифрованы.

Прочитайте Цезарский шифр онлайн: https://riptutorial.com/ru/encryption/topic/8823/цезарский-шифр

кредиты

S. No	Главы	Contributors
1	Начало работы с шифрованием	Community, H. Pauwelyn
2	Текстовое шифрование	bartonjs
3	Цезарский шифр	H. Pauwelyn