УДК 004.75

Лісіцина Катерина Володимирівна,
Старший викладач кафедри економічної кібернетики та інженерії програмного забезпечення, Запорізький інститут економіки та інформаційних технологій
e.lisicina@econom.zp.ua

КРИПТОГРАФІЧНА СИСТЕМА З ПІДТРИМКОЮ БАЗОВИХ АЛГОРИТМІВ ЗМІНИ ВХІДНИХ ДАНИХ

***Анотація**: Стаття присвячена розгляду існуючих моделей безпеки даних. Запропонований алгоритм дозволяє реалізувати модель шифрування з підтримкою базових алгоритмів зміни вхідних даних. На основі алгоритму шифрування інформації без заданої фіксації довжини, необхідно отримати вхідний ключ шифрування, який визначає характер даних. Також реалізована можливість отримати розмір даних, який перетворюється довільним чином після визначення розміру наступного блоку шифрування. Отримані дані, що визначають розмір блоку шифрування, так само шифруються. Це має дозволити швидко зашифровувати великий об'єм (розмір) даних без необхідності розбивати рядки на блоки.*

***Ключові слова**: криптосистема, модель, алгоритм, шифровка, розшифрування, ключ, блок шифрування, ідентифікатор, сигнатура, шифр заміни, шифр підстановки, взаємодоповнюючі сигнатури.*

Сьогодні розробляються різні методи забезпечення захисту даних, що викликає особливе ставлення до її секретності, безпеки. Всі ці методи називаються криптографією. Виключити можливість доступу, що є несанкціонованим, до інформації будь-якого типу дозволяють деякі криптографічні протоколи. У разі їх передачі ці протоколи можуть забезпечувати ідентифікацію зашифрованої інформації. У ряді деяких випадків за допомогою шифрування можна гарантувати певну безпеку цієї інформації на заданому рівні. Основне завдання при оборотному зміну інформації - підвищення надійності цих змін. У деяких системах, що дозволяє зберігати, передавати і обробляти інформацію, можуть використовуватися різні методи шифрування. Таким чином, шифрування дозволяє підвищити секретність інформації, що представляється актуальною проблемою,

Шифрування - оборотне перетворення інформації з метою приховування від неавторизованих осіб, з наданням, в цей же час, авторизованим користувачам доступу до неї. Головним чином, шифрування служить завданням дотримання конфіденційності інформації, що передається. Важливою особливістю будь-якого алгоритму шифрування є використання ключа, який стверджує вибір конкретного перетворення з сукупності можливих для даного алгоритму.

Блоковий шифр - різновид симетричного шифру, що оперує групами біт фіксованої довжини - блоками, характерний розмір яких змінюється в межах 64-256 біт.

За допомогою шифрування забезпечуються три стану безпеки інформації:

– конфіденційність - шифрування використовується для приховування інформації від неавторизованих користувачів при передачі або при зберіганні;

– цілісність - шифрування використовується для запобігання зміни інформації при передачі або зберіганні;

– ідентифіковані - шифрування використовується для аутентифікації джерела інформації та запобігання відмови відправника інформації від того факту, що дані були відправлені саме їм.

Щоб прочитати зашифровану інформацію, приймаючій стороні необхідні ключ і дешифратор (пристрій, що реалізує алгоритм розшифрування).

Ідея шифрування полягає в тому, що зловмисник, перехопивши зашифровані дані і не маючи до них ключа, не може ні прочитати, ні змінити передану інформацію. Крім того, в сучасних криптосистемах (з відкритим ключем) для шифровки і розшифровування даних можуть використовуватися різні ключі. Однак, з розвитком крипто аналізу, з'явилися

методики, що дозволяють дешифрувати закритий текст без ключа. Вони засновані на математичному аналізі переданих даних.

Цілі шифрування. У наш час шифрування застосовується для зберігання важливої інформації в ненадійних джерелах і передачі її по незахищеним каналам зв'язку. Така передача даних представляє із себе два взаємно зворотних процесу:

– перед відправленням даних по лінії зв'язку або перед приміщенням на зберігання вони піддаються зашифровки.

– для відновлення вихідних даних із зашифрованих до них застосовується процедура розшифрування.

Шифрування спочатку використовувалося тільки для передачі конфіденційної інформації. Однак згодом шифрувати інформацію почали з метою її зберігання в ненадійних джерелах. Шифрування інформації з метою її зберігання застосовується і зараз, це дозволяє уникнути необхідності в фізично захищеному сховище.

Шифром називається пара алгоритмів, що реалізують кожне із зазначених перетворень. Ці алгоритми застосовуються до даних з використанням ключа. Ключі для шифровки і для розшифрування можуть відрізнятися, а можуть бути однаковими. Секретність другого (розшифровується) з них робить дані недоступними для несанкціонованого ознайомлення, а таємність першого (шифрувального) унеможливлює внесення неправдивих даних. У перших методах шифрування використовувалися однакові ключі, однак в 1976 році були відкриті алгоритми із застосуванням різних ключів. Збереження цих ключів в секретності і правильне їх поділ між адресатами є дуже важливим завданням з точки зору збереження конфіденційності інформації, що передається [1, 2].

На даний момент існує величезна кількість методів та способів шифрування [9, 10]. Головним чином ці методи діляться, в залежності від структури використовуваних ключів, на симетричні методи і асиметричні методи. Крім того, методи шифрування можуть мати різну криптостійкості і по-різному обробляти вхідні дані - блокові шифри і потокові шифри. Всіма цими методами, їх створенням і аналізом займається наука криптографія.

Рис. 1. Класифікація криптографічних методів

Симетричні криптосистеми. Існують також симетричні криптосистеми (симетричне шифрування, симетричні шифри) - спосіб шифрування, в якому для шифрування і розшифрування застосовується один і той же криптографічний ключ. До винаходу схеми асиметричного шифрування єдиним існуючим способом було симетричне шифрування.

Ключ алгоритму повинен зберігатися в секреті обома сторонами. Алгоритм шифрування вибирається сторонами до початку обміну повідомленнями [3, 4].

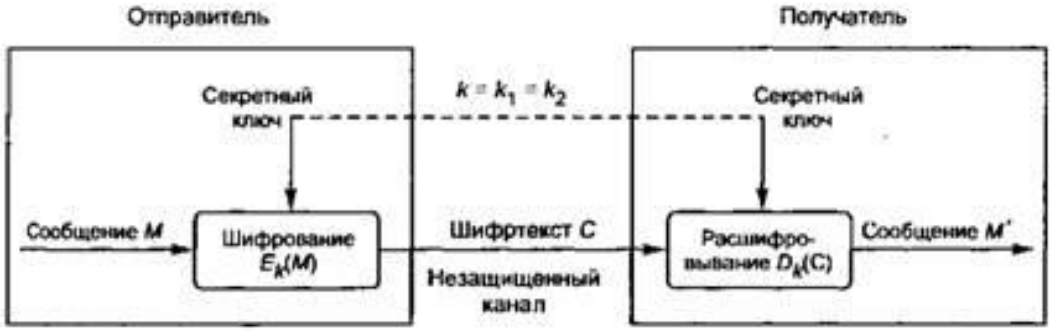

Рис. 2. Схема симетричної криптосистеми шифрування

Блоковий шифр. Блоковий шифр - різновид симетричного шифру, що оперує групами біт фіксованої довжини - блоками, характерний розмір яких змінюється в межах 64-256 біт. Якщо вихідний текст (або його залишок) менше розміру блоку, перед шифруванням його доповнюють.

Блоковий шифр є важливою компонентою складовою багатьох криптографічних протоколів і широко використовується для захисту даних, що передаються по мережі [5-8].

Даний метод в ГОСТ 28147-89 називається "режим простої заміни".

Шифрування може бути описано наступним чином:

Ci = F (Pi) для будь-яких i від 1 до N.

Тут Ci і Pi - блоки зашифрованого і відкритого текстів відповідно, а F - функція блочного шифрування. Розшифрування по тій же схемі. Тобто Pi = F-1 (Ci).

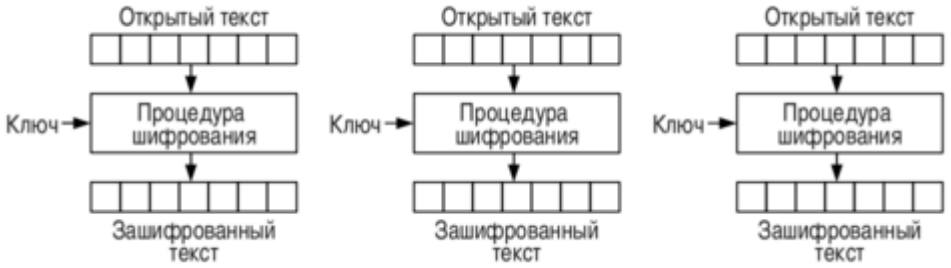

Рис. 3. Шифрування в режимі ECB

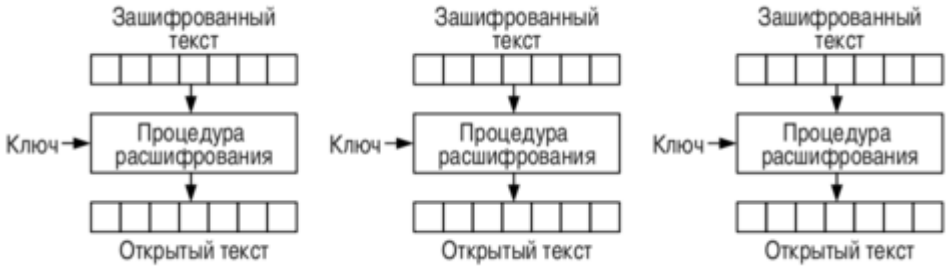

Рис. 4. Розшифрування в режимі ECB

Вступна інформація по цілям і завданням. Виходячи з визначення блочного шифру, в даній статті розглядається інший підхід до даного типу шифрування: передбачається такий алгоритм шифрування, який буде оперувати блоками різної довжини.

Визначення розмірів блоків - завдання моделі шифрування.

Так само з цього випливає, що в разі нестачі вихідного тексту (повідомлення, або його залишок) виконання доповнення даних блоку неприпустимо, так як повна ідентифікованість буде порушена.

Завдання:

– побудова структури ключа шифрування;

– створення методів обробки інформації, що дозволяють виконати основну мету роботи;

– створення певної кількості методів найпростіших змін інформації; кількість цих методів має визначатися структурою ключа.

В результаті виконання поставлених завдань повинен вийти симетричний блоковий алгоритм шифрування, який оперує блоками інформації різного розміру.

Базовий ключ шифрування, опис структури. На рисунку 5 показана загальна структура ключа шифрування.

Рис. 5. Загальна структура базового ключа.

Також на даному зображенні 5 описані методи виконання процесу шифрування, згідно її послідовності:

– методи вибору даних, що визначають розміри блоків; в прикладі реалізації - це кількість байтів, які визначають кількість інформації, що обробляється в подальшому;

– метод обробки первинно обраних даних; якщо за першим пунктом структури ключа введений ключ містить одиницю - ця частина структури не враховується;

– методи шифрування блоків для всього файлу - перелік сигнатур шифрування.

Наведена структура дозволить обробляти інформацію блоками різного розміру.

Всього присутні 10 сигнатур, і порядок їх вказівки в водимо ключі має значення - саме тому при розшифруванні ці сигнатури необхідно обробляти в зворотному порядку.

Теоретично, довжина такого ключа шифрування може бути необмеженою, однак у прикладі реалізації обмеження довжини ключа становить 12 символів: два ідентифікатора обробки первинних даних і десять сигнатур.

Мінімальний розмір блоку шифрування в прикладі реалізації складе 1 байт. Якщо встановлений розмір дорівнює нулю - обробка наступного блоку не виконується (так як він відсутній), і зчитуються нові первинні дані.

Тобто, кількість базових ключів складає 13 333 333 332 - порядкові 12 мільярдів і 1 333 333 332 записи чисел з провідними нулями.

Так само безпосередньо перед обробкою вхідних даних введений ключ шифрується власними сигнатурами. При зашифровки зашифрований ключ записується в вихідний набір даних. При розшифруванні з вхідного набору зчитується еквівалентну кількість даних, і виконується порівняння з зашифрованим ключем; при збігу розшифрування триває, при розбіжності відображається відповідне повідомлення, і обробка вхідного набору даних припиняється.

Опис методів первинної обробки інформації. Якщо перша частина структури введеного ключа більше одиниці - в прикладі реалізації буде лічено вказану кількість байтів, і ці байти необхідно обробити так, щоб отримати одне число, яке буде визначати розмір наступного блоку інформації. Всього необхідно десять математичних операцій над ліченими даними:

– сума значень (максимум 2 550);

– середнє арифметичне (максимум 255);

– половина суми значень (максимум 1 275);
– максимальне значення (максимум 255);
– мінімальне не нульовий значення (максимум 255);
– сума парних значень (максимум 2 540);
– сума непарних значень (максимум 2 550);
– твір кількості парних значень на загальну кількість (максимум 100);
– твір кількості непарних значень на загальну кількість (максимум 100);
– твір довжини рядка значень на кількість значень (максимум 300).

Тобто, теоретична максимальна довжина блоку шифрування в прикладі реалізації може скласти 2 550 байт.

Первинні дані так само будуть оброблятися сигнатурами ключа: при зашифровки обробка виконується після встановлення розміру блоку, при розшифруванні - перед встановленням розміру. Оброблені первинні дані записуються в вихідний набір даних.

Опис сигнатур ключа. Структура базового ключа передбачає 10 сигнатур шифрування:

– позитивний зсув значень на кількість значень; якщо уявити, що значення - це вертикальний параметр, а кількість значень - горизонтальний параметр, тоді такий шифр заміни виконує позитивний зсув вертикальних параметрів на горизонтальний параметр;
– негативний зсув значень на кількість значень;
– позитивний зсув значень на половину кількості значень;
– негативне зміщення значень на половину кількості значень;
– повний реверс значень - перше значення замінюється останнім, друге передостаннім, і так далі;
– половинний реверс значень - масив значень ділиться на два масиви, і виконується повний реверс кожної половини масиву;
– заміна першої половини значень на другу половину;
– попарний реверс значень - перше значення замінюється другим, третє значення замінюється четвертим, і так далі;
– попарне витяг значень - непарні позиції (1, 3, 5, ...) записуються в перший набір значень, парні позиції (2, 4, 6, ...) записуються в другій набір значень; два набори значень об'єднуються;
– попарно вставка значень - значення з непарними позиціями замінюються значеннями з порядковими позиціями 1 - n / 2, значення з парними позиціями замінюються значеннями з порядковими позиціями n / 2 + 1 - n.

За списком пари сигнатур 1-2, 3-4 і 9-10 є взаємодоповнюючі сигнатурами, тобто, якщо при зашифровки використовується перша сигнатура, то при розшифруванні необхідно використовувати другу сигнатуру.

П'ята, шоста, сьома і восьма сигнатури є самодостатніми, що в свою чергу створює проблему неефективних наборів сигнатур - все ключі, які містять ці сигнатури, і запис сигнатур є паліндромом, в результаті не змінять вхідні дані. Незалежно від порядку розстановки сигнатур з таким набором завжди існує 136 400 неефективних ключів (0,001023 відсотків від загальної кількості).

Проблему неефективних ключів можна вирішити, якщо відмовитися від самодостатніх сигнатур, для чого необхідно створити дві пари взаємообернених сигнатур.

Завдання реалізованої моделі криптосистеми. Завдання моделі шифрування:

– вхідний ключ шифрування буде визначати характер визначення даних, які і послужать для подальшого визначення розміру блоку шифрування;
– після визначення розміру (наступного) блоку шифрування кількість даних (отриманий розмір) має перетворюватися довільним чином; методи перетворення так само будуть визначатися ключем;
– дані, що визначають розмір блоку шифрування так само будуть піддаватися шифрування;

– ключ шифрування при зашифровки шифруватиметься згідно з пунктом 2.1, і записуватися в початок вихідних даних.

Алгоритм програмної реалізації моделі криптосистеми. Виходячи з усього перерахованого вище, можна реалізувати модель шифрування з підтримкою базових алгоритмів зміни вхідних даних (алгоритм шифрування інформації без заданої фіксації довжини (АШД)):

– при початку обробки файлу запускається головний потік обробки. Нижче наведено фрагмент коду потокової функції, яка виконує обробку файлу:

```
// опрацювання вхідних даних
     bool generalresult=true;
     string outfname;
     int i;
     if (encrypting)
          outfname=filename+".sdcs";
     else
          …
// опрацювання зовнішніх даних
     int endcount,required;
     filestatus=input.Position;
     while (input.Position<maxfilestatus)
     {
          if (statuscode>2)
          {
               generalresult=false;
               goto end;
          } else {}
          if (progress.pausethread!=null)
               if (progress.pausethread.IsAlive)
                    try
                    {
                         progress.pausethread.Join();
                    } catch {}
               else {}
          …
// вивільнення ресурсів
     end:;
     filestatus=-1;
     input.Unlock(0,maxfilestatus);
     input.Close();
     input.Dispose();
     input=null;
     output.Close();
     output.Dispose();
     output=null;
     countdata=signatures=null;
     if (!generalresult)
          File.Delete(outfname);
     else {}
     outfname=null;
     GC.Collect();
} catch
```

```
        {
            statuscode=2;
        }
    }
```

– у тілі функції головного потоку виконується ініціалізація необхідних програмних полів (локальні і глобальні змінні), первісна перевірка ключа шифрування;

– у тілі циклу самої обробки викликається дочірня функція установки розміру блоку шифрування, функція обробки зчитувальних масивів (всередині якої викликаються кінцеві функції сигнатур).

Нижче наведено фрагмент коду функції, який визначає розмір блоку шифрування.

```
private int SetCount(byte[] countdata)
{
    int
        result=0,sum=0,i,
        length=countdata.Length;
    for (i=0;i<length;i++)
        sum+=countdata[i];
    switch (nkeydata[1])
    {
        case 0: // сума значень
            result=sum;
            break;
        …
        case 9: //
            string temp="";
            for (i=0;i<length;i++)
                temp+=countdata[i].ToString();
            result=temp.Length*length;
            temp=null;
            //GC.Collect();
            break;
    }
    return result;
}
```

По завершенні обробки масиви записуються в вихідний набір даних.

Може статися випадок, коли кількість отриманих байтів не відповідає необхідному кількості (зазвичай, це завершення файлу) - для таких випадків передбачена функція регулювання розміру масиву лічених байтів, в якості параметрів приймає як масив (з необхідним розміром), так і число, яке повинно стати нової довжиною прийнятого масиву.

Під час тестування АШД було використано ключ шифрування 990123456789: комп'ютер, на якому проводилося тестування, виконує обробку зі швидкістю 13 762 420 байтів в секунду, або 13,125 мегабайтів в секунду.

Висновки. В результаті був запропонований алгоритм шифрування інформації без заданої фіксації довжини (АШД), при виконанні якого розмір блоку шифрування буде залежати від ключа і шифрованих даних, де головним завданням є відсутність фіксованої довжини, а також була побудована модель криптосистеми, що дозволяє шифрувати інформацію блоками різного розміру.

На основі алгоритму шифрування інформації без заданої фіксації довжини ми маємо:

– вхідний ключ шифрування, який визначає характер визначення даних;

– отриманий розмір даних, який перетворюється довільним чином після визначення розміру наступного блоку шифрування;

– дані, що визначають розмір блоку шифрування так само піддаються шифрування.

Це дозволяє швидко зашифровувати великий об'єм (розмір) даних без необхідності розбивати рядки на блоки.

Список використаних джерел

1. А. П. Алферов, А. Ю. Зубов, А. С. Кузьмин, А. В. Черемушкин. Основы Криптографии. - Гелиос АРВ, 2002.

2. Мао В. Современная криптография: Теория и практика. - М.: Вильямс, 2005.

3. Гатчин Ю. А., Коробейников А. Г. Основы криптографических алгоритмов. Учебное пособие. - СПб.: СПбГИТМО(ТУ), 2002.

4. Коробейников А. Г. Математические основы криптографии. Учебное пособие. СПб: СПб ГИТМО (ТУ), 2002.

5. Брюс Шнайер. "Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си". - М.: Триумф, 2002.

6. Габидулин Э. М., Кшевецкий А. С., Колыбельников А. И. Защита информации: учебное пособие. - М.: МФТИ, 2011.

7. Баричев С. Г., Гончаров В. В., Серов Р. Е. Основы современной криптографии: 3-е изд. — М.: Диалог-МИФИ, 2011.

8. А. П. Алферов, А. Ю. Зубов, А. С. Кузьмин, А. В. Черемушкин. "Основы криптографии". - М.: Гелиос АРВ, 2002.

9. Багатоканальні апаратно-орієнтовані процесори симетричного блокового шифрування: Автореф. дис... канд. техн. наук: 05.13.05 / Т.А. Коркішко ; Нац. ун-т "Львів. політехніка". - Л., 2002. 19 с.

10. Конфігуровані ядра комп'ютерних пристроїв симетричного блокового шифрування: Автореф. дис... канд. техн. наук: 05.13.05 / В.А. Мельник ; Нац. ун-т "Львів. політехніка". - Львів, 2004. 19 с.

References

1. A.P. Alferov, A. Yu. Zubov, A.S. Kuzmin, A.V. Cheremushkin. Cryptography Basics. - Helios ARV, 2002.

2. Mao V. Modern cryptography: Theory and practice - M .: Williams, 2005.

3. Gatchin Yu. A., Korobeinikov A. G. Fundamentals of cryptographic algorithms. Tutorial. - SPb .: SPbGITMO (TU), 2002.

4. Korobeinikov A. G. Mathematical foundations of cryptography. Tutorial. SPb: SPb GITMO (TU), 2002.

5. Bruce Schneier. "Applied cryptography. Protocols, algorithms, source codes in the C language." - M: Triumph, 2002.

6. Gabidulin E. M., Kshevetsky A. S., Kolybelnikov A. I. Information protection: a training manual - M: MIPT, 2011.

7. Barichev S. G., Goncharov V. V., Serov R. E. Fundamentals of modern cryptography - 3rd ed. - M: Dialog-MEPhI, 2011.

8. A. P. Alferov, A. Yu. Zubov, A. S. Kuzmin, A. V. Cheremushkin. "Basics of cryptography." - M: Helios ARV, 2002.

9. Bagatokanalny hardware and hardware processors of symmetric block encryption: Abstract. dis cand. tech. Sciences: 05.13.05 / T.A. Korkishko; Nat University "Lviv. Politechnika". - L., 2002. 19 p.

10. Configuring the core of computer devices for symmetric block encryption: Abstract. Dis. cand. tech. Sciences: 05.13.05 / V.A. Miller Nat University "Lviv. Politechnika". - Lviv, 2004 .- 19 p.

Лисицына Екатерина Владимировна,
старший преподаватель кафедры экономической кибернетики и инженерии программного обеспечения, Запорожский институт экономики и информационных технологий
e.lisicina@econom.zp.ua

КРИПТОГРАФИЧЕСКАЯ СИСТЕМА С ПОДДЕРЖКОЙ БАЗОВЫХ АЛГОРИТМОВ ИЗМЕНЕНИЯ ВХОДНЫХ ДАННЫХ

Статья посвящена рассмотрению существующих моделей безопасности данных. Предложенный алгоритм позволяет реализовать модель шифрования с поддержкой базовых алгоритмов изменения входных данных. На основе алгоритма шифрования информации без заданной фиксации длины, необходимо получить входной ключ шифрования, который определяет характер данных. Также реализована возможность получить размер данных, который превращается произвольным образом после определения размера следующего блока шифрования. Полученные данные, определяющие размер блока шифрования, так же шифруются. Это должно позволить быстро зашифровывать большой объем (размер) данных без необходимости разбивать строки на блоки.

***Ключевые слова**: криптосистема, модель, алгоритм, шифровка, расшифровка, ключ, блок шифрования, идентификатор, сигнатура, шифр замены, шифр подстановки, взаимодополняющие сигнатуры.*

Lisitsyna Kateryna Volodymyrivna,
Senior lecturer of the Department of Economic Cybernetics and Software Engineering, Zaporizhzhya Institute of Economics and Information Technologies
e.lisicina@econom.zp.ua

CRYPTOGRAPHIC SYSTEM WITH THE SUPPORT OF BASIC ALGORITHMS OF CHANGING INPUT DATA

This article considers existing models of data security. The proposed algorithm allows you to implement an encryption model with support for basic algorithms for changing input data.Based on the algorithm of encryption of information without a given fixed length, it is necessary to obtain an input encryption key, which determines the nature of the data definition It is also possible to obtain the size of the data, which turns arbitrarily after determining the size of the next encryption block. The received data determining the size of the encryption block is also encrypted. This should allow you to quickly encrypt a large amount (size) of data and you don't need to break the string into blocks.

***Keywords**: Cryptosystem, model, algorithm, encryption, decryption, key, encryption unit, identifier, signature, replacement code, substitution cipher, complementary signatures*.