УДК 004.056.53

КРИПТОГРАФИЧЕСКИЕ МЕТОДЫ КАК СРЕДСТВО ЗАЩИТЫ ОТ КИБЕРТЕРРОРИЗМА

Яворская А.Н.
Донецкий национальный университет, Донецкий юридический институт

Борьба с преступлениями в сфере использования компьютерных систем является на сегодня одной из важнейших задач. Для того чтобы эта борьба была максимально эффективной необходимы исследования данной формы преступности, ее составляющих, а также глубокое изучение организационно-правовых и программных средств и мероприятий, которые предотвращают ее массовое распространение. Учитывая то, что на уровне юридических норм урегулировать все отношения по защите информации в компьютерных системах невозможно, в нормативных актах специально определяются принципы государственной политики Украины в сфере защиты информации. Перечислю некоторые важнейшие принципы: комплексность, полнота и непрерывность методов защиты информации; открытость нормативно-правовых актов по защите информации, которые не являются государственной тайной; обязательность защиты инженерно-техническими средствами информации, которая является государственной тайной; финансовая обеспеченность системы защиты информации за счет Государственного бюджета Украины, бюджета Автономной Республики Крым, местных бюджетов, и других источников; иерархичность построения организационных структур системы защиты информации. Рассматривая основные принципы, мы можем проследить, как формируется и проводится государственная политика Украины в сфере защиты информации [3].

Компьютерные системы, в свою очередь, могут быть приспособлены к защите хранящейся на них информации от всех людей, за исключением, естественно, тех, кому разрешен доступ к ним, путем зашифрования данных в формы, весьма устойчивые к попыткам взлома. Разные люди понимают под шифрованием разные вещи. Дети играют в игрушечные шифры и секретные языки. Это, однако, не имеет ничего общего с настоящей криптографией. Настоящая криптография (strong cryptography) должна обеспечивать такой уровень секретности, чтобы можно было надежно защитить критическую информацию от расшифровки крупными организациями - такими как мафия, транснациональные корпорации и крупные государства. Настоящая криптография в прошлом использовалась лишь в военных целях. Однако сейчас, со становлением информационного общества, она становится центральным инструментом для обеспечения конфиденциальности и защиты от высокотехнологического терроризма – кибертерроризма, т.е. преднамеренных атак на информацию, которые создают опасность для жизни и здоровья людей или наступление других тяжких последствий, если также действия были совершены с целью нарушения общественной безопасности [1].

Предметом криптографии является один из классов методов, предназначенных для защиты процессов информационного взаимодействия от отклонений от их нормального течения, вызванных целенаправленными воздействиями со стороны субъектов, называемых злоумышленниками. От прочих методов защиты информации криптографические методы отличаются тем, что основаны на преобразовании информации по секретным алгоритмам (алгоритм, какая-либо деталь которого держится в секрете, и включает в себя открытые алгоритмы, часть параметров которых держится в тайне).

По мере образования информационного общества, крупным государствам, а также опасным субъектам несанкционированного доступа к компьютерной информации (хакеры, пираты, кибертеррористы) становятся доступны технические средства тотального надзора за миллионами людей. Поэтому криптография становится одним из основных инструментов обеспечивающих конфиденциальность и доверие, корпоративную безопасность и защищенность от различного вида атак, позволяющих проникнуть в атакуемую сеть для достижения неправомерных целей. Но для полной нейтрализации угроз кибертерроризма необходима консолидация всего мирового сообщества по ликвидации социальных, экономических и идеологических корней этого явления [2].

Рассмотрим некоторые наиболее часто используемые современные методы защиты информации – такие как RSA, SSL и Kerberos.

RSA (буквенная аббревиатура от фамилий Rivest, Shamir и Adleman) — криптографический алгоритм с открытым ключом. Криптографические системы с открытым ключом используют так называемые необратимые функции, которые обладают следующим свойством:

Если известно x, то $f(x)$вычислить относительно просто

Если известно $y = f(x)$, то для вычисления x нет простого (эффективного) пути.

Под однонаправленностью понимается не теоретическая однонаправленость, а практическая невозможность вычислить обратное значение, используя современные вычислительные средства, за обозримый интервал времени.

В основу криптографической системы с открытым ключом RSA положена задача умножения и разложения составных чисел на простые сомножители, которая является вычислительно однонаправленной задачей.

В криптографической системе с открытым ключом каждый участник располагает как открытым ключом (*public key*), так и секретным ключом (*secret key*). Каждый ключ — это часть информации. В криптографической системе RSA каждый ключ состоит из пары целых чисел. Каждый участник создаёт свой открытый и секретный ключ самостоятельно. Секретный ключ каждый из них держит в секрете, а открытые ключи можно сообщать кому угодно или даже публиковать их. Открытый и секретный ключи каждого участника обмена сообщениями образуют «согласованную пару» в том смысле, что они являются **взаимно обратными**, т.е $\forall$ сообщения $M \in W$, где W — множество допустимых сообщений. $\forall$ открытого и секретного ключа P и S $\exists$ соответствующие функции шифрования $E_p()$ и расшифрования $D_s()$:

$$M = D_s(E_p(M)),$$

$$M = E_p(D_s(M)).$$

RSA-ключи генерируются следующим образом[5]: выбираются два случайных простых числа *p* и *q* заданного размера (например, 1024 бита каждое). Вычисляется их произведение *n* = *pq*, которое называется **модулем**. Вычисляется значение функции Эйлера от числа *n*:

$$\varphi(n) = (p-1)(q-1).$$

Выбирается целое число *e* ($1 < e < \varphi(n)$), взаимно простое со значением функции $\varphi(n)$. Обычно в качестве *e* берут простые числа, содержащие небольшое количество единичных битов в двоичной записи, например, простые числа Ферма 17,

257, или 65537. Число *e* называется **открытой экспонентой** (*public exponent*).Время, необходимое для шифрования с использованием быстрого возведения в степень, пропорционально числу единичных бит в *e*. Слишком малые значения *e*, например 3, потенциально могут ослабить безопасность схемы RSA.

Вычисляется число *d*, мультипликативно обратное к числу *e* по модулю $\varphi(n)$, то есть число, удовлетворяющее условию:

$$de \equiv 1 \pmod{\varphi(n)}, \text{или: } de = 1 + k\varphi(n),$$

где *k* — некоторое целое число. Число *d* называется **секретной экспонентой**. Обычно, оно вычисляется при помощи расширенного алгоритма Евклида.

Пара *P* = (*e*,*n*) публикуется в качестве **открытого ключа RSA** (*RSA public key*).

Пара *S* = (*d*,*n*) играет роль **секретного ключа RSA** (*RSA private key*) и держится в секрете.

Предположим, сторона B хочет послать стороне A сообщение M.Сообщением являются целые числа лежащие от 0 до $n-1$, т.е $M \in D = \mathbb{Z}_n$.

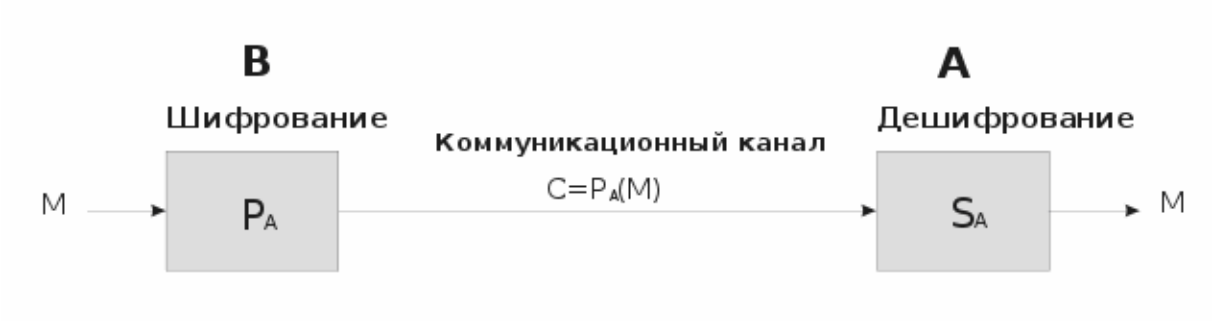

Рисунок 1 – Общая схема информационного канала

Алгоритм шифрования:

Взять *открытый ключ* (e, n)стороны A.
Взять открытый текст M. Передать шифрованное
сообщение:$P_A(M) = M^e \mod n$

Алгоритм дешифрования:

Принять зашифрованное сообщение C.
Применить свой *секретный ключ* (d, n) для расшифровки
сообщения:$S_A(C) = C^d \mod n$

Другим не менее важным методом является SSL (*Secure Sockets Layer* — уровень защищённых сокетов)— криптографический протокол, который обеспечивает установление безопасного соединения между клиентом и сервером. SSL изначально разработан компанией Netscape Communications.

Протокол SSL состоит из двух подпротоколов: протокол SSL записи и рукопожатия. Протокол SSL записи определяет формат, используемый для передачи данных. Протокол SSL включает рукопожатие с использованием протокола SSL записи для обмена сериями сообщений между сервером и клиентом, во время установления первого соединения. Для работы SSL требуется, чтобы на сервере имелся SSL-сертификат.

SSL предоставляет канал, имеющий 3 основные свойства:

6. ***Аутентификация***. Сервер всегда аутентифицируется, в то время как клиент аутентифицируется в зависимости от алгоритма.

7. ***Надёжность***. Обмен сообщениями включает в себя проверку целостности.
8. ***Частность канала***. Шифрование используется после установления соединения и используется для всех последующих сообщений.

В протоколе SSL все данные передаются в виде записей - объектов, состоящих из заголовка и передаваемых данных. Передача начинается с заголовка. Заголовок содержит либо два, либо три байта кода длины. Причём, если старший бит в первом байте кода равен единице, то запись не имеет заполнителя и полная длина заголовка равна двум байтам, иначе запись содержит заполнитель и полная длина заголовка равна трём байтам. Код длины записи не включает в себя число байт заголовка. Длина записи 2х байтового заголовка:

RecLength=(byte[0] & 0x7F<<8) | byte[1];

Здесь byte[0] и byte[1] первый и второй полученные байты. Длина записи 3х байтового заголовка:

RecLength = (byte[0] & 0x3F<<8)|byte[1]; Escape = (byte[0] & 0x40)!=0; Padding = byte[2];

Здесь Padding определяет число байтов добавленных отправителем к исходному тексту, для того чтобы сделать длину записи кратной размеру блока шифра, при использовании блочного шифра. Теперь отправитель «заполненной» записи добавляет заполнитель после имеющихся данных, и шифрует это всё это. Причем содержимое заполнителя никакой роли не играет. Из-за того, что известен объём передаваемых данных, то заголовок может быть сформирован с учетом Padding. В свою очередь получатель записи дешифрует всё поле данных и получает полную исходную информацию. Затем производится вычисление значения RecLength по известному Padding, и заполнитель из поля данных удаляется. Данные записи SSL состоят из 3х компонент:

MAC_Data[Mac_Size] — (Message Authentication Code) — код аутентификации сообщения

Padding_Data[Padding] — данные заполнителя

Actual_Data[N] — реальные данные

Когда записи посылаются открытым текстом, очевидно, что никакие шифры не используются. Тогда длина Padding_Data и MAC_Data равны нулю. При использовании шифрования, Padding_Data зависит от размера блока шифра, а MAC_Data зависит от выбора шифра. Пример вычисления MAC_Data:

MacData = Hash(Secret, Actual_Data, Padding_Data, Sequence_Number);

Значение Secret зависит от того, кто (клиент или сервер) посылает сообщение. Sequence_Number — счетчик, который инкрементируется как сервером, так и клиентом. Здесь Sequence_Number представляет собой 32х битовый код, передаваемый хэш-функции в виде 4х байт, причем первым передается старший байт. Для MD2, MD5 MAC_Size равен 16 байтам (128 битам). Для 2х байтового заголовка максимальная длина записи равна 32767 байтов, а для 3х байтного заголовка 16383 байтов.

Целесообразно привести положительные стороны при применении данного метода:

1. Криптографическая безопасность: SSL устанавливает безопасное соединение между двумя сторонами.
2. Совместимость: Программисты, независимо друг от друга могут создавать приложения использующие SSL, которые впоследствии будут способны успешно обмениваться криптографическими параметрами без всякого знания кода чужих программ.

3. Расширяемость: SSL стремится обеспечить рабочее пространство, в котором новые открытые ключи и трудоемкие методы шифрования могут быть включены по мере необходимости.

Еще один метод, который я хотела бы рассмотреть это сетевой протокол аутентификации Kerberos, позволяющий безопасно передавать данные через незащищённые сети для безопасной идентификации. Также является набором бесплатного ПО от Массачусетского технологического института (Massachusetts Institute of Technology (MIT)), разработавшего этот протокол. Её организация направлена в первую очередь на клиент-серверную модель и обеспечивает взаимную аутентификацию — оба пользователя через сервер подтверждают личности друг друга. Сообщения, отправляемые через протокол Kerberos, защищены от прослушивания и атак повторного воспроизведения. Kerberos является одним из вариантов протокола Нидхема-Шрёдера[4], основан на симметричной криптосистеме и требует третье доверенное лицо (сервер). Расширение Kerberos позволяет использовать открытые ключи в процессе аутентификации.

Итак, поскольку компьютерный терроризм уже представляет собой угрозу сегодняшних дней, необходимо закрепить на законодательном уровне обязанность государственных частных структур принять технические меры по защите компьютерной информации, а также возможность использования криптографии как действенное средство защиты информации от кибертерроризма.

Литература

[1] Голубев В.А. Кибертерроризм – угроза национальной безопасности.-http://www.crime-reseach.ru/articles/Golubev_Cyber_Terrorism/.

[2] Голубєв В.О., Гавловський В.Д., Цимбалюк В.С. Інформаційна безпека: проблеми боротьби зі злочинами у сфері використання комп'ютерних технологій / За заг.ред.доктора юридичних наук, професора Р.А.Калюжного.-Запоріжжя:Просвіта,2001.-С.19.

[3] Калюжний Р.А., Колпак Р.Л. Застосування інформаційних технологій організованою злочинністю для впливу на суспільство.-Боротьба з організованою злочинністю і корупцією(теорія та практика)//Науково-практичний журнал.-№3.-2001.-С.160.

[4] Шнайер Б. Протокол Kerberos // Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си (Applied Cryptography. Protocols, Algorithms and Source Code in C). – М.: Триумф, 2002. – С. 81. — 816 с.

[5] A. Menezes, P. van Oorschot, S. Vanstone. 8.2. RSA public-key encryption // Handbook of Applied Cryptography. – CRC-Press, 1996. – 816 p.