Тема 15. ЙМОВІРНІСНЕ ШИФРУВАННЯ. КВАНТОВА КРИПТОГРАФІЯ

1. Ідея ймовірнісного шифрування

Поняття ймовірнісного шифрування було винайдено Шафі Голдвассером (Shafi Goldwasser) і Сільвією Мікалі. Хоча їх теорія дозволяє створити найбезпечнішу із винайдених криптосистем, рання реалізація була неефективною, але більш пізніші реалізації все змінили.

Ідея ймовірнісного шифрування: усунути витікання інформації у криптографії з відкритими ключами. Так як криптоаналітик завжди може розшифрувати випадкове повідомлення відкритим ключем, він може отримати деяку інформацію. При умові, що у нього є шифротекст $C=E_K(M)$, і він намагається отримати відкритий текст M, він може вибрати випадкове повідомлення M' і зашифрувати його: $C'=E_K(M')$. Якщо $C'=C$, то він вгадав правильний відкритий текст. В протилежному випадку він робить наступну спробу.

Крім того, ймовірнісне шифрування дозволяє уникнути навіть часткового витікання інформації про справжнє повідомлення. При використанні криптографії з відкритими ключами крипто аналітик може розізнати дещо про біти: XOR 5-го, 17-го та 39-го біт рівне 1, і т.д. При ймовірнісному шифруванні залишається скритою і така інформація.

Таким способом можна витягнути не багато інформації, але потенційно можливість криптоаналітика розшифрувати випадкове повідомлення вашим відкритим ключем може створити певні проблеми. Кожен раз, шифруючи повідомлення, криптоаналітик може витягнути трохи інформації. Ніхто не знає, наскільки важлива ця інформація.

Ймовірнісне шифрування намагається усунути витік такої інформації. Ціль цього методі полягає в тому, щоб ні в обчисленнях, які виконуються над шифротекстом, ні перевірка будь-яких інших відкритих текстів не могли дати криптоаналітику ніякої інформації про відповідний відкритий текст.

При ймовірнісному шифруванні алгоритм шифрування є ймовірнісним, а не детермінованим. Іншими словами, багато шифротекстів при розшифруванні

дають даний відкритий текст, і конкретний шифротекст, що використовується в будь-якому конкретному шифруванні, вибирається випадковим чином.

$C_1=E_K(M)$, $C_2= E_K(M)$, $C_3= E_K(M)$, ... $C_i= E_K(M)$,

$M=D_K(C_1)=D_K(C_2)=D_K(C_3)=...=D_K(C_i)$.

При ймовірнісному шифруванні крипто аналітику більше не вдасться шифрувати будь-які відкриті тексти в пошуках правильного шифротекста. Для ілюстрації, нехай у криптоаналітика є шифротекст $C_i= E_K(M)$. Навіть якщо він правильно вгадає M, отриманий при шифруванні $E_K(M)$ результат буде зовсім іншим шифротекстом C: C_j. Порівнюючи C_i та C_j, він не може по їх співпаданню визначити правильність своєї здогадки.

Навіть якщо у крипто аналітика є відкритий ключ шифрування, відкритий текст і шифротекст, він не може без закритого ключа довести, що шифротекст є результатом шифрування конкретного відкритого тексту. Навіть виконавши весь пошук, він може довести тільки, що кожний можливий відкритий текст є можливим відкритим текстом.

У цій схемі шифротекст завжди буде більший за відкритий текст. Цього неможливо уникнути, оскільки це є результатом того, що багато шифротекстів розшифровуються в один і той самий відкритий текст. В першій схемі ймовірнісного шифрування шифротекст отримувався настільки більшим відкритого, що він був не корисним.

2. Ефективна реалізація ймовірнісного шифрування

Мануель Блюм (Manual Blum) та Гольдвассер отримали ефективну реалізацію ймовірнісного шифрування з допомогою генератора псевдовипадкових бітів Blum Blum Shub (BBS, тема 12, пункт 5).

Генератор BBS оснований на теорії квадратичних залишків. Існує два простих числа, p та q, конгруентних 3 по модулю 4. Це закритий ключ. Їх добуток, $n=pq$, є відкритим ключем. Безпека схеми опирається на складність розкладання n на множники.

Для шифрування повідомлення M спочатку вибирається випадкове число x, взаємно просте з n. Потім обчислюється $x_0=x^2 \ mod \ n$. x_0 – служить

початковою послідовністю для генератора псевдовипадкових бітів BBS, а вихід генератора використовується в якості потокового шифру. Побітно виконується XOR M з виходом генератора. Генератор видає біти b_i (молодший значущий біт x_i, де $x_i=x_{i-1}^2 \bmod n$), тому

$M=M_1, M_2, M_3, \ldots M_t$

$c=M_1 \oplus b_1, M_2 \oplus b_2, M_3 \oplus b_3, \ldots M_t \oplus b_t$, де t – це довжина відкритого тексту. І остання дія: добавити останнє обчислене значення, x_t, до кінця повідомлення.

Розшифрувати це повідомлення можна тільки одним способом – отримати x_0 і з цієї стартової послідовності запустити генератор BBS, виконуючи XOR виходу з шифротекстом. Тільки той хто знає p та q, зможе розшифрувати повідомлення.

Цю схему можна зробити ще швидшою, використовуючи всі відомі безпечні біти x_i, а не тільки молодший значущий біт. З таким покращенням ймовірнісне шифрування Blum-Goldwasser виявляється швидшим RSA і не допускає витікання інформації про відкритий текст. Крім того, можна довести, що складність розкриття цієї схеми рівна складності розкладу n на множники.

З іншої сторони, ця схема зовсім небезпечна по відношенню до розкриття з вибраним шифротекстом. За молодшими значущими бітами правильних квадратичних залишків можна обчислити квадратний корінь будь-якого квадратичного залишку. Якщо це вдасться, то вдасться і розкласти на множники.

3. Ймовірнісні моделі шифру

На основі відображень шифру та ймовірнісних моделях відкритого тексту і множини ключів можна побудувати ймовірнісну модель шифру. Позначимо деякі розподіли ймовірностей (р.й.):

$P_{вх}$ – р.й. на множині відкритих текстів;

$P_{кл}$ – р.й. на множині ключів;

$P_{ш}$ – р.й. на множині шифрованих текстів;

$P_{вх,к}$ – загальний р.й. на множині пар відкритих текстів та ключів;

$Р_{вх,ш}$ – загальний р.й. на множині пар відкритих та шифрованих текстів;

$Р_{вх/ш}$ – умовне р.й. на множині відкритих текстів (при умові що шифрований текст фіксований).

Нехай *a* – відкритий текст, *z* – ключ, *y* – шифрований текст, *E(a,z)* – криптограмма, отримана в результаті шифрування відкритого тексту *a* ключем *z*.

Зазвичай вважають, що при шифруванні ключ *z* вибирається незалежно від відкритого тексту *a*. Тому $P_{вх,к}(a,z)=P_{вх}(a) * P_{кл}(z)$

Загальні і умовні розподіли ймовірностей визначаються з формул:

$$P_{ш}(y) = \sum_{(a,z):E(a,z)=y} P_{вх}(a) \cdot P_{кл}(z),$$

$$P_{вх,ш}(a,y) = \sum_{z:E(a,z)=y} P_{вх}(a) \cdot P_{кл}(z),$$

$$P_{вх/ш}(a/y) = P_{вх,ш}(a,y)/P_{ш}(y),$$

де остання рівність випливає з визначення умовної ймовірності і справедливе при умові коли розподіл $P_{ш}(y)>0$.

Таким чином, маючи розподіли ймовірностей на множині відкритих текстів та ключів та знаючи сімейство шифрів, можна обчислити як розподіл ймовірності на множині шифрованих текстів, так і різні загальні і умовні розподіли ймовірностей.

Використовуючи ймовірнісну модель шифру, Шеннон вперше сформулював поняття абсолютно стійкого шифру. [12]

4. Що таке квантова криптографія?

Квантова криптографія вводить природну невизначеність квантового світу. З її допомогою можна створювати лінії зв’язку, які не можливо послухати, не вносячи перешкод у передачу. Закони фізики захищають такий квантовий канал, навіть якщо той хто підслуховує може виконувати будь-які дії, навіть якщо він має доступ до не обмежуваної обчислювальної потужності, навіть якщо *P=NP*. Шарль Бене, Жіль Брассар, Клод Крепо та інші розширили цю ідею, описавши квантовий розподіл ключів, квантову передачу з забуванням, квантові обчислення з декількома учасниками.

Ці ідеї так би й залишилися предметом обговорення фанатів криптографії, але Банне і Брассар розробили діючу модель. Тепер у нас є експериментальна квантова криптографія.

У відповідності з законами квантової механіки частинки насправді не знаходяться на одному місці, а з визначеною ймовірністю існують зразу у багатьох місцях. Однак, це так тільки до тих пір, поки не приходить вчений і не виміряє частинку, що виявилася в даному конкретному місці. Але ви міряти всі параметри частинки (наприклад, координати та швидкість) одночасно неможливо. Якщо міряти одну з цих двох величин, то акт вимірювання знищує всяку можливість виміряти другу величину. Невизначеність є фундаментальною властивістю квантового світу.

Цю невизначеність можна використовувати для генерації секретного ключа. Подорожуючи, фотони коливаються у визначеному напрямі, вверх-вниз, вліво-вправо, або, що більш ймовірно, під якимось кутом. Звичайне сонячне світло неполяризоване, фотони коливаються у всіх можливих напрямках. Коли напрям коливань багатьох фотонів співпадає, вони є поляризовані. Поляризаційні фільтри пропускають тільки ті фотони, які поляризовані у визначеному напрямку, а інші блокуються.

Нехай є імпульс горизонтально поляризованих фотонів. Якщо вони спробують пройти через горизонтальний фільтр, то у них це вийде вдало. Якщо повільно повертати фільтр на 90°, кількість фотонів, що можуть пройти, ставатиме все менше і менше, і на кінець жоден фотон не зможе пройти через фільтр. Це перечить здоровому глузду. Здається, що навіть незначний поворот фільтра повинен зупинити усі фотони, так як вони горизонтально поляризовані. Але в квантовій механіці кожна частинка з визначеною ймовірністю може змінити свою поляризацію і проскочити через фільтр. Якщо кут відхилення фільтру невеликий, то ймовірність висока, в протилежному випадку – рівна нулю.

5. Протокол передачі з використанням квантової криптографії

Поляризацію можна змінити у будь-якій системі координат: двох напрямах, що розходяться під прямим кутом. Прикладами систем координат є прямокутна – горизонтальне та вертикальне направлення, діагональна – ліва та права діагоналі. Якщо імпульс фотонів поляризований в заданій системі координат, то при вимірюванні в тій же системі координат можна взнати поляризацію. При вимірюванні в неправильній системі координат – отримається випадковий результат. Ця властивість використовується для генерації секретного ключа:

1. Відправник А посилає Б послідовність фотонних імпульсів. Кожен з імпульсів випадковим чином поляризований в одному із чотирьох напрямків: горизонтальному, вертикальному, ліво- та праводіагональному.

   Наприклад, А посилає для Б: | | / — \ — | — /
2. У Б є детектор поляризації. Він може настроїти свій детектор на вимірювання прямокутної або діагональної поляризації. Одночасно міряти ту і іншу у нього не вийде, йому не дозволить квантова механіка. Вимірювання однієї поляризації не дасть виміряти іншу. І так, він встановлює свої детектори випадковим чином:

   X + + X X X + X + +

   Тепер, якщо Б правильно налаштує свій детектор, він зареєструє правильну поляризацію. Якщо він налаштує детектор на вимірювання прямокутної поляризації, то імпульс буде поляризований прямокутно, він взнає, яку поляризацію фотонів вибрав А. Якщо він налаштує детектор на вимірювання діагональної поляризації, а імпульс буде поляризований прямокутно, то результат вимірювання буде випадковим. Б не зможе визначити різницю. У наведеному прикладі він може отримати такий результат: / | — \ / \ — / — |
3. Б повідомляє А по незахищеному каналу, які настройки він використовував.

4. А повідомляє Б, які настройки були правильними. В нашому прикладі детектор був встановлений правильно для імпульсів 2, 6, 7 і 9.
5. А та Б залишають тільки правильно виміряні поляризації. В нашому прикладі вони такі: * | * * * \ — * — *

   З допомогою раніше приготовленого коду А та Б перетворюють в біти ці результати вимірювань поляризації. Наприклад, горизонтальна і ліводіагональна можуть означати одиницю, а вертикальна та праводіагональна – нуль. В нашому прикладі вони обидва отримають 0 0 1 1.

Отже, А та Б отримали чотири біта. З допомогою цієї системи вони можуть генерувати стільки бітів скільки їм необхідно. В середньому Б правильно вгадує в 50% випадків, тому для генерації *n* біт А доведеться вислати *2n* фотонних імпульсів. Вони можуть використовувати ці біти як секретний ключ симетричного алгоритму або забезпечити абсолютну безпеку, отримавши достатньо біт для використання в якості одноразового блокноту.

Плюсом є те, що С не зможе підслухати. Так як і Б, йому доведеться вгадати тип поляризації, і як і у Б, половину здогадок буде неправильна. Так як неправильні вимірювання змінюють поляризацію фотонів, то при підслуховуванні С вносить помилки в передачу. Якщо це так, то А та Б отримають різні бітові послідовності. Отже, А та Б закінчують протокол такими діями:

6. А та Б порівнюють декілька бітів своїх рядків. Якщо є розходження вони взнають про підслуховування. Якщо рядки не відрізняються, то вони відкидають використані для порівняння біти і використовують, ті що залишилися.

Покращення цього протоколу дозволяє А та Б використовувати свої біти навіть у присутності С. Вони можуть порівнювати тільки парність бітових множин. Тоді, якщо не виявлено розходжень, їм доведеться відкинути тільки один біт підмножини. Це виявить підслуховування з ймовірністю 50%, але якщо вони звірять таким чином *n* різних бітових підмножин, ймовірність С підслухати і залишитися непомітною буде рівна $1/2^n$.

В квантовому світі не буває пасивного підслуховування. Якщо С намагатиметься розкрити усі біти, він обов’язково порушить канал зв’язку.