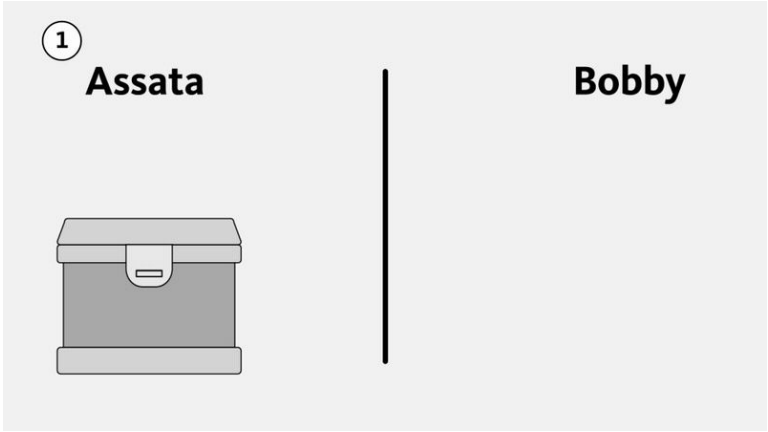
Алгоритм обміну ключами Діффі-Хелмана для двох та більше абонентів. Коректність алгоритму.

Вступ

Протокол узгодження ключів

Протокол узгодження ключів — різновид обміну ключами, коли дві або більше сторони виконують певні кроки аби потім спільно користуватись отриманим таємним значенням. Головна особливість протоколу узгодження ключів полягає в тому, що кожна сторона робить однаковий внесок в обчислення спільного ключа (на відміну від протоколів обміну або поширення ключа)



Протокол Діффі-Геллмана

Протокол Діффі-Геллмана — це метод обміну криптографічними ключами.
Один з перших практичних прикладів узгодження ключа, що дозволяє двом учасникам, що не мають жодних попередніх даних один про одного, отримати спільний секретний ключ із використанням незахищеного каналу зв'язку. Цей ключ можна використати для шифрування наступних сеансів зв'язку, що використовують шифр з симетричним ключем.

Історія створення

Діффі й Геллман запропонували у 1976 році алгоритм для створення криптографічних систем з відкритим ключем, який базується на складності обчислення дискретного логарифма. Алгоритм Діффі–Геллмана може бути використаний для розподілу ключів (генерування секретного ключа), але його не можна використовувати для шифрування повідомлення.

Алгоритм узгодження ключа Діффі–Геллмана запатентовано у США та Канаді. Ліцензію на цей патент разом з іншими патентами в області криптографії з відкритим ключем отримала група Public Key Partners (PKP). Термін дії патенту США вибіг у 1997 році.

Криптографічна стійкість

Криптографічна стійкість алгоритму Діффі–Геллмана заснована на передбачуваній складності проблеми дискретного логарифмування. Однак, хоча вміння вирішувати проблему дискретного логарифмування дозволить зламати алгоритм Діффі–Хеллмана, зворотне твердження ще не доведене.

Необхідно відзначити, що простий алгоритм Діффі–Геллмана працює тільки на лініях зв'язку, надійно захищених від модифікації. Тоді, коли в каналі можлива модифікація даних, з'являється можливість атаки «людина посередині»

Модифікації

Протокол Діффі-Геллмана дозволяє двом сторонам обчислити спільний таємний ключ на основі власних таємних ключів та відкритих ключів один одного. Але третя сторона, якій відомі лише відкриті ключі, не зможе обчислити спільний таємний ключ цих двох.

Від початку заснування було створено численні модифікації протоколу. Були створені варіанти для узгодження ключів не лише між двома сторонами, але й більшою кількістю учасників.

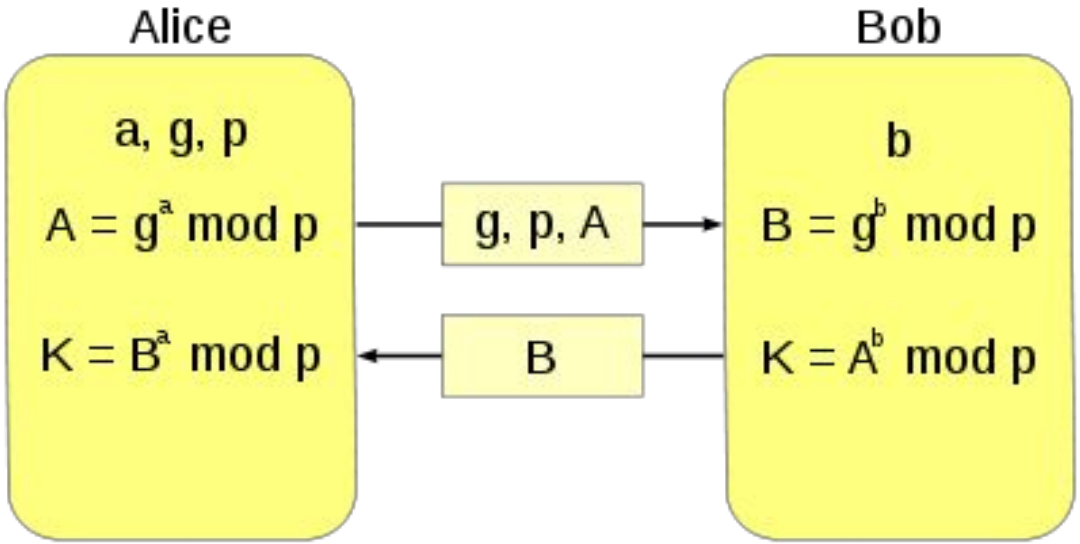
Опис алгоритму для двох абонентів

Опис алгоритму (Перший етап)

Припустимо, що обом абонентам відомі деякі два числа g і p, які не є секретними і можуть бути відомі також іншим зацікавленим особам. Для того, щоб створити більш невідомий нікому секретний ключ, обидва абоненти генерують великі випадкові числа: перший абонент – число a, другий абонент – число b. Потім перший абонент обчислює значення $A = g^a \bmod p$ і пересилає його другому, а другий обчислює $B = g^b \bmod p$ і передає першому. Передбачається, що зловмисник може отримати обидва цих значення, але не модифікувати їх (тобто у нього немає можливості втрутитися в процес передачі).

Опис алгоритму (Другий етап)

На другому етапі перший абонент на основі наявного в нього a і отриманого по мережі B обчислює значення $B^a \bmod p = g^{ab} \bmod p$, а другий абонент на основі наявного в нього b і отриманого по мережі A обчислює значення $A^b \bmod p = g^{ab} \bmod p$. Неважко побачити, що в обох абонентів вийшло одне і те ж число: $K = g^{ab} \bmod p$. Його вони і можуть використовувати в якості секретного ключа, оскільки тут зловмисник зустрінеться з практично нерозв'язною (за розумний час) проблемою обчислення $g^{ab} \bmod p$ по перехоплених $g^a \bmod p$ і $g^b \bmod p$, якщо числа p, a, b вибрані достатньо великими.



$K = A^b \bmod p = (g^a \bmod p)^b \bmod p = g^{ab} \bmod p = (g^b \bmod p)^a \bmod p = B^a \bmod p$

Що ж на практиці?

У практичних реалізаціях, для a і b використовуються числа порядку 10^{100} і p порядку 10^{300}. Число g не обов’язково має бути великим і зазвичай має значення в межах першого десятка. Необхідно зазначити, що даний алгоритм працює тільки на лініях зв'язку, надійно захищених від модифікації. Якби він був застосовний на будь-яких відкритих каналах, то давно зняв би проблему розповсюдження ключів і, можливо, замінив собою всю асиметричну криптографію.

Що ж на практиці?

Однак, у тих випадках, коли в каналі можлива модифікація даних, з'являється можливість атаки людина посередині. Атакуючий замінює повідомлення переговорів про ключ на свої власні і, таким чином, отримує два ключі – свій для кожного з законних учасників протоколу. Далі він може перешифровувати листування між учасниками, своїм ключем для кожного, і, таким чином, ознайомитися з їх повідомленнями, залишаючись непоміченим.

Приклад алгоритму для двох абонентів

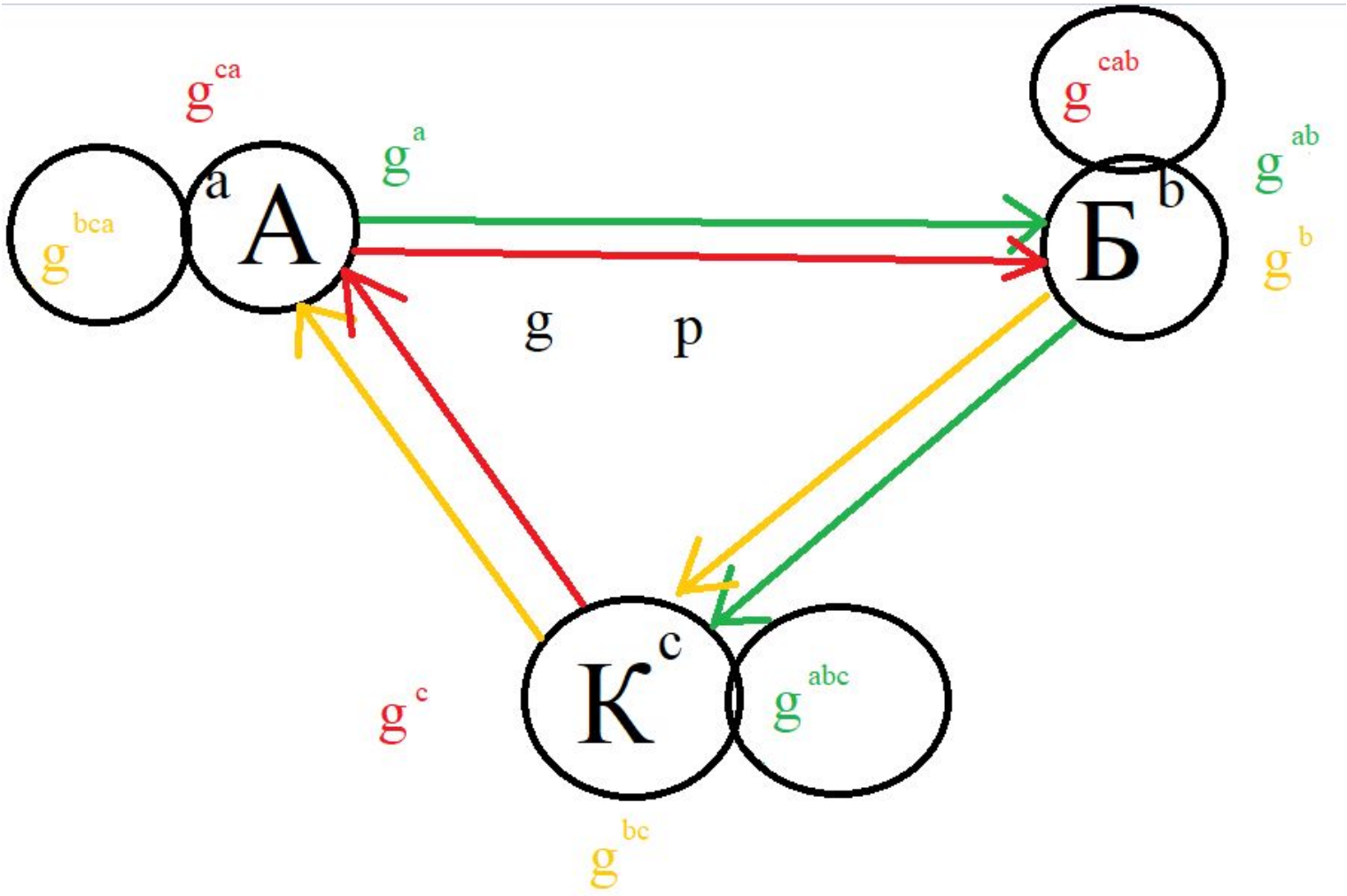
Приклад алгоритму для двох абонентів

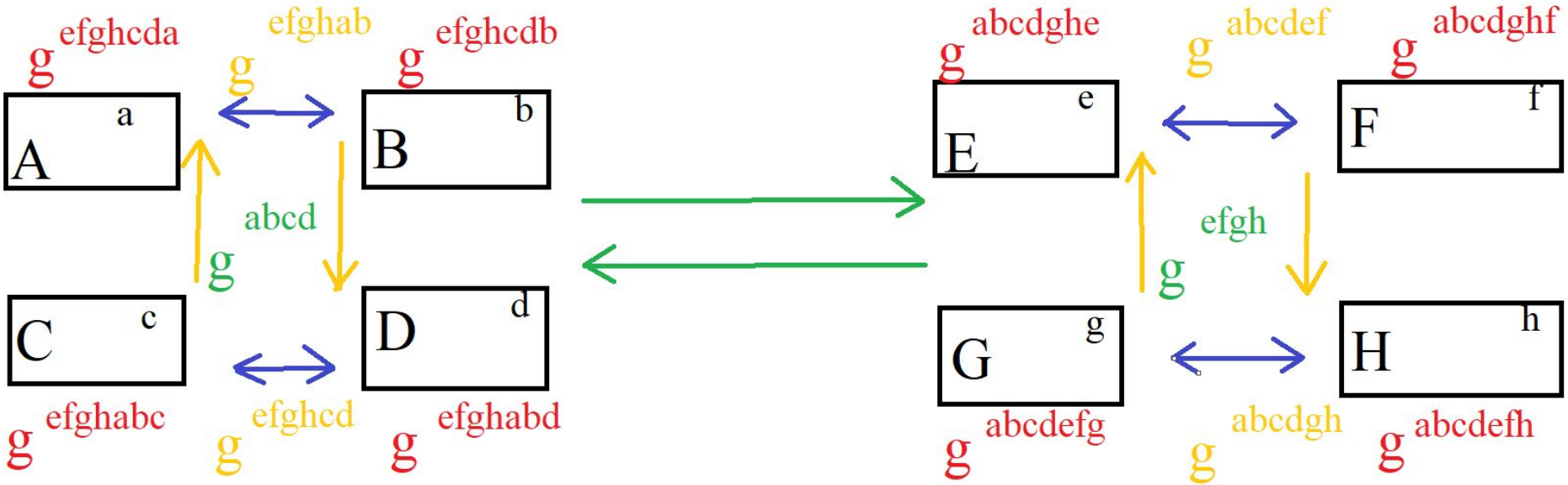
	Аліса Абонент	Єва Прослуховувач	Боб Абонент
$g^x \bmod p$	g = 5 p = 23	g = 5 p = 23	g = 5 p = 23
a,b - приватні ключі	a = 6	a = ? b = ?	b = 15
A,B - публічні ключі	$A = g^a \bmod p = 8$ B = 19	A = 8 B = 19	$B = g^b \bmod p = 19$ A = 8
s - секретний ключ	$s = B^a \bmod p =$ $= 19^6 \bmod 23 = 2$	s = ?	$s = A^b \bmod p =$ $= 8^{15} \bmod 23 = 2$

Опис алгоритму для більше ніж двох абонентів

Опис алгоритму для більше ніж двох абонентів

Протокол Діффі-Геллмана застосовний не лише для узгодження ключа між двома учасниками. Брати участь в узгодженні ключа може будь-яка кількість учасників через ітеративне виконання протоколу узгодження і обмін незасекреченими проміжними даними.


g ca
g cab
g a
g ab
a
A
b
Б
g bca
g b
g p
c
К
g c
g abc
g bc


g efghcda
efghab
g efghcdb
A a
g
B b
abcd
g
C c
D d
g efghabc
g efghcd
g efghabd
g abcdghe
g abcdef
g abcdghf
E e
F f
efgh
g
G g
H h
g abcdefg
g abcdgh
g abcdefh

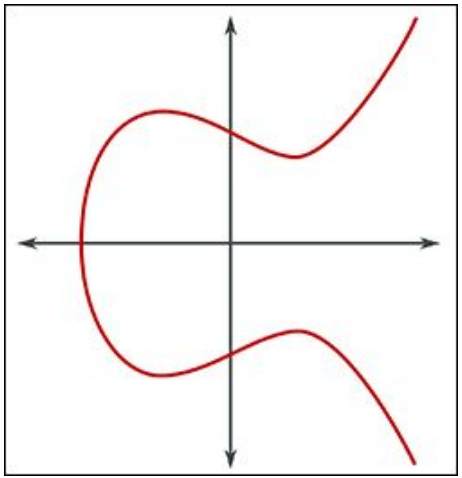
Приклад алгоритму для трьох абонентів

Приклад алгоритму для трьох абонентів

	Аліса Абонент	Боб Абонент	Керол Абонент
$g^x \bmod p$	g = 7 p = 29	g = 7 p = 29	g = 7 p = 29
a,b,c - приватні ключі	a = 4	b = 8	c = 9
A,B,C,AB,AC,BC - публічні ключі	$A = g^a \bmod p = 23$	$AB = A^b \bmod p = 23$	$s = AB^c \bmod p = 7$
	$s = BC^a \bmod p = 7$	$B = g^b \bmod p = 7$	$BC = B^c \bmod p = 20$
s - спільний секретний ключ	$CA = C^a \bmod p = 7$	$s = CA^b \bmod p = 7$	$C = g^c \bmod p = 20$

Варіації обміну ключами Діффі-Хеллмана

Еліптична крива Діффі-Хеллмана



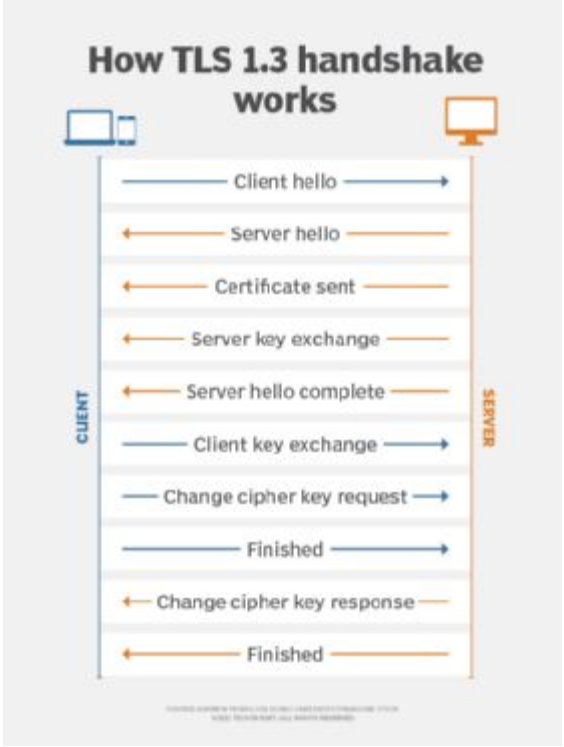
Еліптична крива Діффі-Хеллмана використовує переваги алгебраїчної структури еліптичних кривих, щоб дозволити її реалізації досягти подібного рівня безпеки з меншим розміром ключа. 224-бітний ключ з еліптичною кривою забезпечує такий же рівень безпеки, як і 2048-бітний ключ RSA. Це дозволяє зробити обмін більш ефективним і зменшити вимоги до зберігання.
Окрім меншої довжини ключа та того факту, що він покладається на властивості еліптичних кривих, еліптична крива Діффі-Хеллмана працює так само, як і стандартний обмін ключами Діффі-Хеллмана.

Дізнатись більше: https://medium.com/swlh/understanding-ec-diffie-hellman-9c07be338d4a

TLS

TLS, є протоколом, який використовується для захисту більшої частини Інтернету, може використовувати обмін Діффі-Хеллмана трьома різними способами: анонімним, статичним і ефемерним. На практиці слід реалізувати лише ефемерний Діффі-Хеллмана, оскільки інші варіанти мають проблеми з безпекою.

- Анонімний Діффі-Хеллман – ця версія обміну ключами Діффі-Хеллмана не використовує жодної аутентифікації, що робить її вразливою для атак «людина посередині». Її не слід використовувати чи впроваджувати.
- Статичний Діффі-Хеллман – ця версія використовує сертифікати для автентифікації сервера. Вона не аутентифікує клієнта за замовчуванням і не забезпечує пряму секретність.
- Ефемерний Діффі-Хеллман – вважається найбезпечнішою реалізацією, оскільки забезпечує ідеальну пряму секретність. Зазвичай він поєднується з таким алгоритмом, як DSA або RSA, для автентифікації однієї або обох сторін у з'єднанні. Ефемерний Діффі-Хеллман використовує різні пари ключів під час кожного запуску протоколу. Це надає з'єднанню ідеальну пряму секретність, оскільки навіть якщо ключ буде скомпрометований у майбутньому, його не можна використовувати для розшифрування всіх минулих повідомлень.



Дізнатись більше:
https://www.techtarget.com/searchsecurity/definition/Transport-Layer-Security-TLS

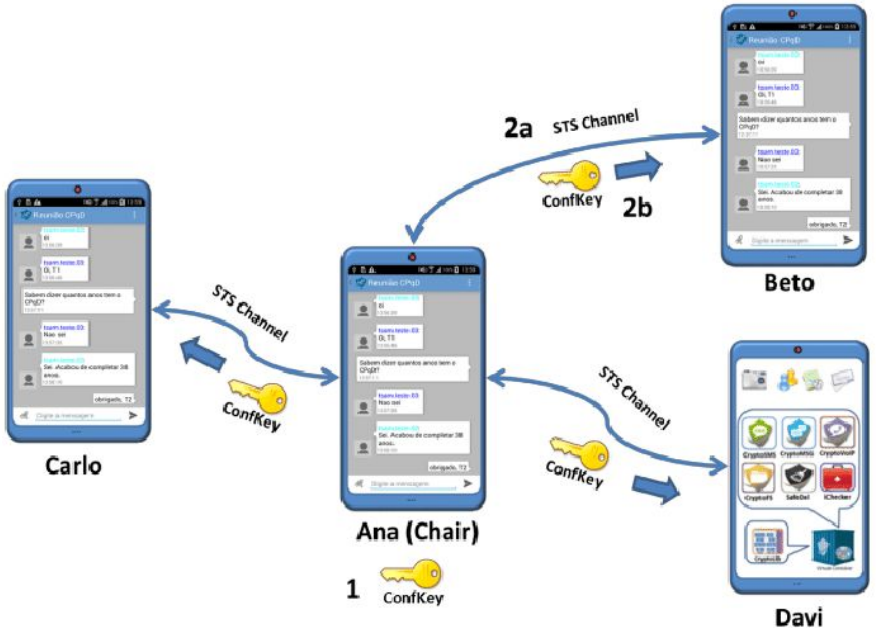
ElGamal

ElGamal — це алгоритм із відкритим ключем, побудований на основі обміну ключами Діффі-Хеллмана. Як і Діффі-Хеллман, він не містить положень щодо самостійної аутентифікації і зазвичай поєднується з іншими механізмами для цієї мети. ElGamal в основному використовувався в PGP, GNU Privacy Guard та інших системах, оскільки його головний конкурент, RSA, був запатентований. Термін дії патенту RSA закінчився в 2000 році, що дозволило його вільно впроваджувати після цієї дати. З тих пір ElGamal впроваджується не так часто.

Дізнатись більше: https://www.educba.com/elgamal-encryption/

STS

Протокол від станції до станції (STS) також заснований на обміні ключами Діффі-Хеллмана. Це ще одна ключова схема угоди, однак вона забезпечує захист від атак «людина посередині», а також ідеальну конфіденційність.
Для цього потрібно, щоб обидві сторони у з'єднанні вже мали пару ключів, яка використовується для аутентифікації кожної сторони. Якщо сторони ще не знайомі одна одній, сертифікати можна використовувати для підтвердження особистості обох сторін.



Дізнатись більше:
http://archive.dimacs.rutgers.edu/Workshops/Security/program2/boyd/node13.html

Підсумки

Переваги алгоритму

1. Відправнику та одержувачу не потрібні будь-які попередні знання один про одного.
2. Після обміну ключами передача даних може здійснюватися через незахищений канал.
3. Спільний доступ до секретного ключа безпечний.

Недоліки алгоритму

1. Алгоритм не може бути ефективно використаний для асиметричного обміну ключами.
2. Так само його не можна використовувати для цифрових підписів.
3. Оскільки він не аутентифікує жодну сторону в передачі, обмін ключами Діффі Хеллмана сприйнятливий до атаки «людина посередині».