Бэкдор на основе криптосистемы Эль-Гамаля

Генерация ключей в криптосистеме Эль-Гамаля:

1) Генерируется случайное простое число p.
2) Выбирается целое число g – первообразный корень p.
3) Выбирается случайное целое число такое, что (1 < x < p-1)
4) Вычисляется $y = g^x \ mod \ p$
5) Открытый ключ – (y,g,p), а закрытый – x

Идея бэкдора:

1) Злоумышленники имеют простое число P и генератор чисел G. Данные параметры формируют открытый ключ злоумышленников Y.
2) Программа злоумышленников, установленная на компьютерах пользователей, формирует параметры p и g, а также закрытый ключ x и открытый ключ y. Параметры и ключи формируются на основе параметров злоумышленников, а также открытом ключе.
3) Суть в том, что открытые параметры p и g – не рандомизированы, а сформированы из закрытого ключа пользователя. Также добавлен дополнительны элемент сокрытия формирования открытых параметров из закрытого ключа путем добавления функций рандомизации.

Алгоритм формирования открытого ключа пользователя:

Пусть P – k-битное простое число злоумышленников. X ∈ {1,2, …, P-1} – закрытый ключ злоумышленников, G – генератор Z_p^* и Y ≡ $G^X(modP)$ – открытый ключ. Все параметры и ключи имеют размерность k-битам.

1) Закрытый ключ пользователя x случайно выбирается из набора {1, 2, …, P-1}
2) Случайно выбирается целое число k.
3) Закрытый ключ x шифруется, используя схему Эль-Гамаля с параметрами злоумышленников (простого числа P и открытого ключа Y):
$$c_2 \equiv \ x * \ Y^k(modP)$$
4) Используя функцию рандомизации R_1 с ключом K + i (i := 0), формируется параметр c_2'
$$c_2' = \ R_1(c_2)$$
такую, что $R_1: \{0,1\}^k \to \{0,1\}^k$
5) Далее может быть 2 варианта:
   1. c_2' не является простым или $c_2' \leq x$, тогда i = i + 1 и c_2 генерируется заново (п. 3)
   2. c_2' является простым и $c_2' > x$, тогда формируется параметр
$$c_1 \equiv \ G^k(mod \ P)$$
6) Используя функцию рандомизации R_2 с ключом K + j (j := 0), формируется параметр c_1'
$$c_1' = \ R_2(c_1)$$
такую, что $R_2: \{0,1\}^k \to \{0,1\}^k$
7) Далее может быть 2 варианта:

1. Если $c_1' \geq c_2'$ или c_1' не является генератором $Z_{c_2'}^*$, тогда j = j + 1 и параметр c_1 генерируется заново (п. 5)
2. Если $c_1' < c_2'$ и является генератором $Z_{c_2'}^*$, тогда

$$p = c_2'$$
$$g = c_1'$$
$$y = g^x \ mod \ p$$

Тем самым сформировался открытый ключ пользователя y, а также параметры p и g.

Стоит отметить, что размерность параметров и открытого ключа пользователя та же, что и размерность параметров и открытого ключа злоумышленников.
Также использование функций генераций R_1 и R_2 необходимо, чтобы параметр p выглядел рандомизированным для пользователя.

Алгоритм восстановления закрытого ключа:
Злоумышленник знает про параметры p и g пользователя, т.к. они отрытые. Восстановим по этим параметрам закрытый ключ x. Также отметим, что p = c_2' и g = c_1'.

1) Используется обратная функция генерации R_2 с ключом K + j, чтобы получить параметр c_1

$$c_1 = R_2^{-1}(g)$$

Так как возможно несколько возможных значений j, то также существует несколько значений c_1. Злоумышленник сразу может отбросить значения, которые больше или равны P, так как $c_1 < P$

2) Используется обратная функция генерации R_1 с ключом K + i, чтобы получить параметр c_2

$$c_2 = R_1^{-1}(p)$$

Так как возможно несколько возможных значений i, то также существует несколько значений c_2. Злоумышленник сразу может отбросить значения, которые больше или равны P.

3) Злоумышленнику необходимо использовать свой закрытый ключ X для получения закрытого ключа пользователя x.

$$\frac{c_2}{c_1^X} = \frac{x * Y^k}{(G^k)^X} = \frac{x * (G^X)^k}{G^{k*X}} = \frac{x * G^{X*k}}{G^{k*X}} = x$$

Пример работы бэкдора:
Алиса и Боб – пользователи, Ева – злоумышленник

1) Генерация ключей Евой:
   - Пусть P = 23993, а G = 15765. При этом G – первообразный корень P.
   - Пусть X = 9237 – приватный ключ, а публичный:

$$Y \equiv G^X \equiv 15765^{9237} \equiv 6211 \ (mod \ 23993)$$

2) Генерация ключей Алисой:
   - Генерируется случайный закрытый ключ: x = 19243

- Генерируется случайное число k = 7661
- Используя закрытый ключ, а также параметры Y и P Евы:

$$c_2 \equiv x * Y^k \equiv 19243 * 6211^{7661} \equiv 21843\ (mod\ 23993)$$

- Рандомизируется параметры:

$$c_2' = R_1(c_2) = R_1(21843) = 27337$$

Отметим, что c_2' - простое и $c_2' > x$

- Вычислям следующий параметр:

$$c_1 \equiv G^k \equiv 15765^{7661} \equiv 7495\ (mod\ 23993)$$

- Рандомизируется параметры:

$$c_1' = R_2(c_1) = R_2(7495) = 10023$$

- Так как $c_1' < c_2'$ и c_1' - первообразный корень c_2', программа выдает пользователю следующие параметры:

$$p = c_2' = 27337$$
$$g = c_1' = 10023$$
$$y \equiv g^x \equiv 10023^{19423} \equiv 13027\ (\mathrm{mod}\ 27337)$$

3) Боб шифрует сообщение с помощью открытого ключа Алисы. Пусть сообщение m = 0809, k = 1487

$$r \equiv g^k \equiv 10023^{1487} \equiv 16434\ (\mathrm{mod}\ 27337)$$
$$s \equiv \mathrm{m} * y^k \equiv 809 * 13027^{1487} \equiv 17176\ (\mathrm{mod}\ 27337)$$

Боб отправляет сообщение (r, s) = (16434, 17176) Алисе

4) Расшифровка сообщения Алисой.

$$m = \frac{s}{r^x} = \frac{17176}{16434^{19423}} = 809\ (mod\ 27337)$$

5) Получение закрытого ключа Алисы Евой:
   - Вычисление c_1 используя параметр g и обратную функцию рандомизции R_2:

$$c_1 = R_2^{-1}(g) = R_2^{-1}(10023) = 7495$$

   - Вычисление c_2 используя параметр p и обратную функцию рандомизции R_1:

$$c_2 = R_1^{-1}(p) = R_1^{-1}(27337) = 21843$$

   - Получение закрытого ключа Алисы используя собственный закрытый ключ X:

$$x \equiv \frac{c_2}{c_1^X} \equiv \frac{21843}{7495^{9237}} \equiv 19423\ (\mathrm{mod}\ 23993)$$

Теперь Ева с помощью полученного закрытого ключа Алисы может расшифровывать переписку Алисы и Боба.