Teoria Data Security 2

Jacopo Manetti

May 2023

1 Probabilità di errore in Miller-Rabin

1.1 Traccia

Lo scopo dell'esercizio è dimostrare che, nel caso in cuil n è un numero composto dispari non Carmichael, sia il test di Fermat che di Miller-Rabin hanno probabilità di errore $\leq 1/2$ (per il caso in cui n è Carmichael, si rimanda alle Note, Appendice A.1. Si ricordi che questa limitazione superiore può essere migliorata a $\leq 1/4$, con una prova più sofisticata). (a) Siano $x, y \in Z_n^*$, dove x è un testimone di Fermat di n mentre y non è testimone di Fermat di n. Dimostrare che $xy \bmod n$ è un testimone di Fermat di n in Z_n^*. (b) Se x è un testimone di Fermat e $y, y' \in Z_n^*$ sono due diversi non testimoni di Fermat, dimostrare che $xy \not\equiv_n xy'$. (c) Dai due punti precedenti, concludere che se c'è almeno un testimone di Fermat (cioè n non è Carmichael), allora in Z_n^*, per ciascuno dei non testimoni c'è almeno un testimone distinto. Dunque ci sono almeno $|Z_n^*|/2 = \phi(n)/2$ testimoni di Fermat in Z_n^*. (d) Concludere che, se n è composto dispari non Carmichael, la probabilità che l'algoritmo ritorni vero è $\geq 1/2$.

1.2 Svolgimento

(a) Siano $x, y \in Z_n^*$, dove x è un testimone di Fermat di n mentre y non è testimone di Fermat di n. Per definizione, dato che x è un testimone di Fermat, abbiamo:

$$x^{n-1} \equiv_n 1 \tag{1}$$

e dato che y non è un testimone di Fermat, abbiamo:

$$y^{n-1} \not\equiv_n 1 \tag{2}$$

Ora consideriamo il prodotto modulo n di x e y:

$$(xy)^{n-1} \bmod n \tag{3}$$

Per il teorema di Euler, sappiamo che:

$$a^{n-1} \equiv_n 1 \quad \forall a \in Z_n^* \tag{4}$$

E quindi:

$$(xy)^{n-1} \equiv_n x^{n-1} y^{n-1} \tag{5}$$

Dato che $x^{n-1} \equiv_n 1$, otteniamo:

$$x^{n-1} y^{n-1} \equiv_n y^{n-1} \tag{6}$$

Ma sappiamo che $y^{n-1} \not\equiv_n 1$. Quindi, abbiamo dimostrato che:

$$(xy)^{n-1} \not\equiv_n 1 \tag{7}$$

Questo implica che $xy \bmod n$ è un testimone di Fermat di n in Z_n^*.

(b) Supponiamo per assurdo che $xy \equiv_n xy'$, dove x è un testimone di Fermat e $y, y' \in Z_n^*$ sono due diversi non testimoni di Fermat. Allora possiamo scrivere:

$$xy \equiv_n xy' \implies x(y - y') \equiv_n 0 \tag{8}$$

Ma sappiamo che x è un testimone di Fermat, quindi $x^{n-1} \equiv_n 1$. Quindi x non può essere divisibile per n, e quindi deve essere $y - y'$ divisibile per n. Ma questo significa che $y \equiv_n y'$, il che è in contraddizione con l'ipotesi che y e y' siano diversi. Quindi, $xy \not\equiv_n xy'$.

(c) Dai punti (a) e (b), sappiamo che se c'è almeno un testimone di Fermat, allora per ogni non testimone di Fermat in Z_n, c'è almeno un testimone di Fermat distinto. Quindi, ci sono almeno $|Z_n| / 2 = \phi(n)/2$ testimoni di Fermat in Z_n^*.

(d) Se n è composto dispari non Carmichael, allora sappiamo che esiste almeno un testimone di Fermat in Z_n. Dato che ci sono almeno $\phi(n)/2$ testimoni di Fermat in Z_n, la probabilità che l'algoritmo scelga un testimone di Fermat come base è almeno $\frac{\phi(n)}{2}/\phi(n) = \frac{1}{2}$. Pertanto, se n è composto dispari non Carmichael, la probabilità che l'algoritmo di Fermat o Miller-Rabin ritorni vero (cioè, identifichi erroneamente n come primo) è al massimo $\leq \frac{1}{2}$.

Da notare che nel caso dell'algoritmo di Miller-Rabin, la probabilità di errore può essere ulteriormente ridotta ripetendo il test con diverse basi scelte casualmente. Per esempio, dopo k iterazioni indipendenti con basi diverse, la probabilità di errore diventa $\leq \frac{1}{2^k}$.

2 Timing attack contro esponenziazione modulare (Kocher 1996)

2.1 Traccia

Il tempo di esecuzione della moltiplicazione modulare tra due numeri di k bit, $a \times b \bmod n$, non è una costante che dipende solo da k, ma varia al variare dei due argomenti a e b. Di conseguenza, anche il tempo di esecuzione dell'algoritmo di esponenziazione per il calcolo di $c^d \bmod n$, una volta fissato l'esponente d di k bit, non è costante, ma varia al variare della base (ciphertext) c. In altre parole, si è visto sperimentalmente che, considerando diverse basi $c_1, c_2, \ldots$ scelte casualmente, i tempi ottenuti possono essere considerati come estrazioni i.i.d. da una variabile aleatoria T, che obbedisce ad una fissata distribuzione di probabilità. Denotando poi con T_j il tempo impiegato dall'iterazione j-ma ($j = k-1, \ldots, 0$) del ciclo for, si può porre

$$T = T_{k-1} + T_{k-2} + \cdots + T_0$$

dove le T_j sono a loro volta v.a. che obbediscono a una certa distribuzione di probabilità. Inoltre, si vede sperimentalmente che i tempi relativi a iterazioni diverse sono tra loro indipendenti. Nel calcolo di questi tempi, per semplicità, si tiene conto solo delle operazioni di moltiplicazione modulare, essendo trascurabili i tempi imputabili ad altre istruzioni (overhead per la gestione del ciclo for, assegnamenti etc.).

Queste caratteristiche possono essere sfruttate per condurre un attacco basato sull'osservazione ripetuta dei tempi di esecuzione dell'algoritmo, per diversi ciphertext c scelti a caso dall'attaccante. L'attacco funziona con tutte le varianti note dell'algoritmo. Esso verrà qui illustrato nel caso dell'algoritmo da sinistra a destra, discusso a lezione.,

(a) Si ricordi che la varianza di una variabile aleatoria a valori reali X di media μ, quando esiste finita, è data da: $var(X) \stackrel{def}{=} E\left[(X-\mu)^2\right] = E\left[X^2\right] - \mu^2$. Provare che per due variabili aleatorie indipendenti X e Y, vale $var(X+Y) = var(X) + var(Y)$.

(b) Si assuma che l'attaccante conosca già i bit da d_{k-1} a d_{i+1} (compresi) dell'esponente. Egli vuole ora dedurre il bit $d_i (k-1 \geq i \geq 0)$. Egli ipotizza un valore $d'_i \in \{0,1\}$ per d_i e, servendosi di una copia identica del dispositivo (smart-card) da attaccare, emula l'algoritmo fino all'iterazione di indice i-mo inclusa. Egli annota il tempo impiegato da tale esecuzione, $T' = T'_{k-1} + \cdots + T'_{i+1} + T'_i$. Confronta poi tale tempo con il tempo di esecuzione T del dispositivo che esegue l'operazione completa, con la medesima base. Dare una formula per $T - T'$.

(c) Dare una formula per $var\,(T - T')$ nei due casi: $d'_i = d_i$ (valore ipotizzato corretto), $d'_i \neq d_i$ (valore ipotizzato scorretto). Nel caso che $d_i \neq d'_i$, le variabili T_i e T'_i possono essere considerate (approsimativamente) indipendenti. Si assuma per semplicità che le T_j e le T'_j abbiano tutte la stessa varianza v.

(d) Supponiamo che l'attaccante ripeta l'operazione del punto (b) due volte: prima ipotizzando $d'_i = 0$, e poi che $d'_i = 1$. Formulare una regola che permetta all'attaccante di stabilire se $d_i = 0$ o $d_i = 1$.

(e) Dire come può essere stimata, in pratica, $var\,(T - T')$, in base alla legge debole dei grandi numeri.

(f) Concludere che l'attaccante può dedurre i bit dell'esponente uno alla volta con la tecnica sopra esposta, partendo dal bit più significativo (ponendo cioè all'inizio $i = k - 1$).

(g) Si riconsideri il punto (c): perché al posto della varianza non può semplicemente essere usata la media di $T - T'$?

Traccia. Per il punto (b), si osservi che deve essere $T'_j = T_j$ per $j = k - 1, \ldots, i + 1$. Per (c), ci si rifaccia all'indipendenza tra le T_j e al punto (a): porre attenzione a valutare correttamente $var\,(T_i - T'_i)$, nel caso $d'_i \neq d_i$. Per (d), si chiamino T'^0 e T'^1 i tempi parziali ottenuti ponendo $d'_i = 0, 1$ rispettivamente, nell'esperimento del punto (b): il valore corretto di d_i corrisponde alla varianza minore tra $var\left(T - T'^0\right)$ e $var\left(T - T'^1\right)$ (perché?). Per (e), si tenga presente che la varianza è un valore atteso, e dunque può essere stimata come una media aritmetica, usando N estrazioni i.i.d. di $T - T'$, per N abbastanza grande:

$$var\,(T - T') \approx \frac{1}{N}\sum_{j=1}^{N}\left(t^{(j)} - t'^{(j)}\right)^2 - \left(\frac{1}{N}\sum_{j=1}^{N}\left(t^{(j)} - t'^{(j)}\right)\right)^2$$

Le $t^{(j)} - t'^{(j)}$ rappresentano qui estrazioni indipendenti di $T - T'$, ciascuna ottenuta misurando i tempi di esecuzione T e T' relativi ad un ciphertext (base) c_j generato casualmente dall'attaccante, per $j = 1, \ldots, N$.

Per maggiori informazioni sull'attacco, consultare per esempio la voce Wikipedia su Timing attack, e i riferimenti ivi contenuti:

https://en. wikipedia. org/wiki/Timing$_a ttack$.

2.2 Svolgimento

(a) Per due variabili aleatorie indipendenti X e Y, la varianza della loro somma è uguale alla somma delle loro varianze. Questo può essere dimostrato come segue:

$$\begin{aligned}
&\text{var(X+Y)} = \text{E}\left[((X+Y) - (E[X] + E[Y]))^2\right] \\
&= E\left[(X - E[X] + Y - E[Y])^2\right] \\
&= E\left[(X - E[X])^2\right] + 2E\left[(X - E[X])(Y - E[Y])\right] + E\left[(Y - E[Y])^2\right] \\
&= var(X) + 2cov(X, Y) + var(Y)
\end{aligned}$$

(9)

Dato che X e Y sono indipendenti, la loro covarianza è zero. Pertanto, $var(X+Y) = var(X) + var(Y)$.

(b) Per calcolare la differenza tra il tempo totale di esecuzione T e il tempo parziale T' ottenuto con l'ipotesi d'_i, possiamo sottrarre T' da T:

$$T-T' = (T_{k-1}+T_{k-2}+\cdots+T_{i+1}+T_i+T_{i-1}+\cdots+T_0)-(T'_{k-1}+T'_{k-2}+\cdots+T'_{i+1}+T'_i) \tag{10}$$

Dato che per $j = k-1, \ldots, i+1$, $T'_j = T_j$, la differenza si riduce a:

$$T - T' = T_i + T_{i-1} + \cdots + T_0 - T'_i \tag{11}$$

(c) Se $d'_i = d_i$, allora $T'_i = T_i$ e la varianza di $T - T'$ è:

$$var(T - T') = var(T_{i-1} + \cdots + T_0) = (i)v \tag{12}$$

Se invece $d'_i \neq d_i$, allora T'_i e T_i possono essere considerati indipendenti e la varianza è:

$$var(T-T') = var(T_i-T'_i)+var(T_{i-1}+\cdots+T_0) = 2v+(i)v = (i+2)v \tag{13}$$

(d) L'attaccante può sottrarre il tempo misurato con $d'_i = 0$, $T^{0'}$, e quello con $d'_i = 1$, $T^{1'}$, dal tempo totale T. L'ipotesi d'_i che minimizza la varianza di $T - T^{d'_i}$

corrisponderà al vero valore di d_i. La ragione per cui l'attaccante mira a minimizzare la varianza risiede nel fatto che il vero valore di d_i comporterà la minore variabilità nel tempo di esecuzione, perché i passaggi corrispondenti nell'algoritmo di decrittografia saranno consistenti.

In altre parole, se l'attaccante sceglie un valore errato per d_i, l'emulazione dell'algoritmo includerà passaggi non necessari (nel caso di $d'_i = 1$ quando $d_i = 0$) o ometterà passaggi necessari (nel caso di $d'_i = 0$ quando $d_i = 1$). Queste discrepanze introdurranno variabilità aggiuntiva nel tempo di esecuzione, portando a una varianza più grande.

(e) Secondo la legge debole dei grandi numeri, la varianza può essere stimata come la media degli quadrati dei campioni meno il quadrato della media dei campioni, per un numero sufficientemente grande di campioni. Questo può essere espresso come segue:

$$var(T - T') \approx \frac{1}{N}\sum_{j=1}^{N}(t^{(j)} - t'^{(j)})^2 - \left(\frac{1}{N}\sum_{j=1}^{N}(t^{(j)} - t'^{(j)})\right)^2 \tag{14}$$

(f) L'attaccante può usare questa tecnica per dedurre i bit dell'esponente uno alla volta, partendo dal bit più significativo. L'attaccante può assumere un valore per d'_i, misurare T', calcolare $T - T'$ e la sua varianza, e ripetere il processo per l'altro valore di d'_i. Il valore di d'_i che minimizza la varianza sarà il vero valore di d_i.

(g) Non si può semplicemente utilizzare la media di $T - T'$ perché l'obiettivo è di distinguere tra le due possibili ipotesi per d'_i. Anche se le medie di $T - T'^0$ e $T - T'^1$ potrebbero essere simili, le varianze possono essere significativamente diverse, indicando quale ipotesi per d'_i è corretta. Inoltre, la media non cattura la variabilità nei dati, che è cruciale per distinguere tra le due ipotesi.