Indicazioni per il tool JaMeter

Augusto Ciuffoletti

12 gennaio 2004

1 Introduzione

Il tool JaMeter ha la stessa struttura del protocollo ping: da un lato un nodo che spedisce messaggi di misura, dall'altro un nodo che risponde prima possibile ai messaggi con un messaggio di eco.

Il tool per la misura di JA, dal lato del probe, apre una sessione persistente con il target, utilizzando il protocollo UDP. I parametri statici della sessione sono:

1. gli indirizzi IP dei due host
2. la porta di ingresso scelta dal probe
3. il periodo tra due misure successive
4. la lunghezza del messaggio
5. il fattore di scala dei timestamp
6. le strutture socket utilizzate per la comunicazione
7. la "firma" del target

I parametri dall'1 al 5 vengono comunicati dall'utente all'atto dell'invocazione, e quelli dal 3 al 5 sono pure negoziati con il target. I parametri 6 e 7 sono calcolati dal probe al momento dell'apertura della sessione.

Il probe deve mantenere uno stato della sessione che consiste nella registrazione dei parametri. Una sessione sul probe corrisponde ad un processo che si sveglia ad intervalli determinati, ed ogni volta esegue un roundtrip.

(OBSOLETO I valori in uscita consistono nei timestamps significativi di ogni roundtrip) (DAFARE I valori in uscita corrispondono al timestamp corrispondente all'istante iniziale del roundtrip, e ai tre intervalli corrispondenti alle latenze dei messaggi di andata e di ritorno, e dal ritardo di elaborazione indotto dal target.)

Il target non mantiene uno stato delle sessioni a cui partecipa: si vuole che la sua implementazione sia più leggera possibile. Quindi il target corrisponde ad un processo che risponde ai messaggi che arrivano su una certa porta (determinata e nota al probe). La risposta viene inviata su una porta del probe, da questo indicata in ciascun messaggio.

Il probe predispone due porte: una per la spedizione ed una per la ricezione. La scelta della prima porta compete al kernel, mentre la seconda rientra tra i parametri operativi del probe.

Può essere opportuna una forma di autenticazione basata su una chiave rilasciata dal target stesso. La chiave non è personale di un certo probe, ma è condivisa da tutti le sessioni che condividono quel target. (Gli IP di provenienza di messaggi con chiavi sbagliate vengono registrate DA FARE).

Il problema della lunghezza dei timestamps devee essere preso in considerazione: timestamps troppo lunghi o che prefigurano un valore massimo sono fonte di guai. Limitando l'estensione del timestamp a 32 bit (4 ottetti) si può avere una discreta definizione ed un orizzonte di durata sufficiente. La dinamica è di circa 10^9. Ad esempio, utilizzando parte di un timestamp NTP:

```
risoluzione: 1 microsecondo
periodo: >1 ora
```

Poichè tra i parametri operativi del probe c'è un fattore di scala, la rappresentazione può essere collocata nel range più opportuno, con una latitudine di scelta. Il fattore di scala indica la posizione del punto decimale, decrementata di 10 unità, a partire dal bit meno significativo nella rappresentazione su 32 bit: dalla posizione 10 (indicata da un fattore di scala 0, corrispondente alla virgola prima del 11o bit) alla posizione 25 (indicata da un fattore di scala 15, corrispondente alla virgola prima del 26-esimo bit).

La tabella 1 (calcolata in modo automatico) fa corrispondere ad un fattore di scala la risoluzione ed il periodo del timestamp.

Scegliendo un fattore di scala opportuno per un certo link, il vincolo è che:

scala	risoluzione (μsec)	periodo (min)
15	0.030	2.1
14	0.060	4.2
13	0.119	8.5
12	0.238	17.1
11	0.477	34.1
10	0.954	68.2
9	1.907	136.5
8	3.815	273.1
7	7.629	546.1
6	15.259	1092.2
5	30.518	2184.5
4	61.035	4369.1
3	122.070	8738.1
2	244.141	17476.2
1	488.281	34952.5
0	976.562	69905.1

Tabella 1: Corrispondenza tra fattore di scala, risoluzione e periodo

- un roundtrip abbia una durata sempre inferiore a due volte il periodo,
- la probabilità che il roundtrip cada a cavallo di una rotazione sia bassa, in quanto questo evento può invalidare la lettura,
- che la durata tipica di un roundtrip sia ben superiore (almeno 10 volte) alla risoluzione del timestamp.

Con le tecnologie che conosco, un fattore di scala 10 sembra adeguato a situazioni generiche:

```
risoluzione: 1 microsecondo circa
periodo: 1 ora circa
```

quindi viene per il momento impostato nel programma (DAFARE).

Rispetto al timestamp NTP il valore del timestamp viene ottenuto con uno shift a destra di tutti e 8 gli ottetti del timestamp NTP di 10+s (con s fattore di scala). Questo può essere così realizzato:

```
- x = 10+s
- shift a destra di x bit della parte decimale del timestamp NTP
- shift a sinistra di 32-x bit della parte intera del timestamp NTP
- OR dei due risultati
```

Gli shift inseriscono 0 nelle posizioni libere.

I clock dei due nodi, probe e target, possono essere caratterizzati da un offset e da un drift arbitrari. Si pone il problema di calcolare differenze tra timestamps, visto che questa è l'unica operazione sui timestamps realmente indispensabile. Per realizzarla si ricorre all'aritmetica in modulo:

```
$x - $y =
    if ( $y > $x ) {
return ($x+(~$y));
    } else {
return ($x-$y);
    }
```

E' l'operazione più "pulita" che sono riuscito a fare. Può risultare ridondante, basterebbe `($x-$y)` in aritmetica a 32 bit, ma evita possibili overflow. E' da ritoccare per aritmetiche con più di 32 bit.

La differenza `($x-$y)` normalmente cresce con `$x` e decresce con `$y`. Tuttavia, quando si ha transizione da `$x>$y` a `$x<$y` o viceversa, intesi come interi a 32 bit senza segno, si ha una discontinuità che va gestita: infatti si passa da valori prossima a `0x0`, a valori prossimi a `0xFFFFFFFF` o viceversa. La situazione si presenta nel caso di clock ben sincronizzati e con tempi di comunicazione dello stesso ordine dell'offset del clock. Il problema non è risolto nella versione presente (BUG), ma viene rilevato con un warning (non testato). Una possibile soluzione potrebbe essere l'aggiunta sistematica di un offset fittizio, appena viene rilevato il problema: se l'offset fittizio è consistente il problema non si dovrebbe ripresentare per molto tempo.

```
    ( $fwdDelay>>1 < $bwdDelay>>1) && ( $precfwd>>1 > $precbwd>>1) &&
warn ("bug: caso a\n");
    ( $fwdDelay>>1 > $bwdDelay>>1) && ( $precfwd>>1 < $precbwd>>1) &&
warn ("bug: caso b\n");
```

Si assume che il roundtrip sia sempre inferiore al periodo dei timestamp, altrimenti le differenze non vengono calcolate correttamente. Al momento il timeout sul roundtrip è fissato a 10 secondi, che è inferiore al periodo massimo (con fattore di scala uguale a 15 è di poco più di 2 minuti. Tuttavia un roundtrip a cavallo della scadenza del periodo può essere invalidato (BUG, non rilevato).

2 Tipi di pacchetti

I pacchetti potrebbero essere dei tipi sequenti:

- un pacchetto di apertura della sessione (open)
- un pacchetto di risposta all'apertura della sessione (reply)
- un pacchetto di misura dal probe al target (forward)
- un pacchetto di eco dal target al probe (backward)

In apertura, il probe invia un pacchetto di apertura, a cui può fare riscontro un pacchetto di risposta.

Se viene ricevuto il pacchetto di risposta, la sessione viene aperta con i parametri ricevuti dal target.

2.1 I pacchetti di apertura

Servono essenzialmente a chiedere al target gli estremi di autenticazione per iniziare la sessione ed a negoziare i parametri della sessione. Il messaggio di open indica la lunghezza dei messaggi di misura, la loro frequenza (media), la scala dei tempi e la porta di ricezione del probe. Su questa base il target può decidere se accettare o meno le misure. Il messaggio di reply indica una controproposta del target per lunghezza, frequenza e scala, e la chiave di autenticazione. La mancata risposta entro un timeout indica non disponibilità.

2.2 Il formato dei pacchetti di apertura

2.2.1 Messaggio di apertura (open)

Versione e tipo di pacchetto (8 bit) 4 bit ciascuno.

Non assegnato (8 bit) .

Lunghezza (16 bit) La lunghezza del pacchetto di misura, questo header compreso (ma escluso lo header IP).

Fattore di scala dei tempi (4 bit) La collocazione della virgola fissa nella rappresentazione in secondi.

Frequenza (28 bit) Espressa nella stessa scala dei tempi dei timestamps.

Non assegnato (16 bit) .

Porta (16 bit) La porta su cui il probe riceve gli eco.

```
=====================================
|   |   |         |                 |
| V | T |  vuoto  |   Lunghezza     |
|   |   |         |                 |
=====================================
|    |                              |
| Sc |          Frequenza           |
|    |                              |
=====================================
|                 |                 |
|      vuoto      |     Porta       |
|                 |                 |
=====================================
|        |        |        |        |
```

2.2.2 Messaggio di risposta all'apertura (reply)

Versione e tipo di pacchetto (8 bit) 4 bit ciascuno.

Non assegnato (8 bit) .

Lunghezza (16 bit) La lunghezza del pacchetto di misura, questo header compreso (ma escluso lo header IP).

Fattore di scala dei tempi (4 bit) La collocazione della virgola fissa nella rappresentazione in secondi.

Frequenza (28 bit) Espressa nella stessa scala dei tempi dei timestamps.

Firma (32 bit) La chiave di autenticazione richiesta dal target.

```
=====================================
|   |   |         |                 |
| V | T |  vuoto  |   Lunghezza     |
|   |   |         |                 |
=====================================
|    |                              |
| Sc |           Frequenza          |
|    |                              |
=====================================
|                                   |
|               Chiave              |
|                                   |
=====================================
|        |        |        |        |
```

2.3 I pacchetti di misura

Contengono i timestamp: uno del probe e due del target. Il timestamp inserito dal probe serve ad identificare il roundtrip: verrà controllato dal probe stesso ed è più breve del timestamp del target, ma ha lo stesso fattore di scala: sostituisce l'indicazione del periodo dei pacchetti di apertura. Il target inserisce i timestamp relativi ai tempi di ricezione della richiesta, e spedizione della eco.

Il pacchetto di andata contiene un campo (disponibili 64 bit) per l'autenticazione del mittente.

Il timestamp del probe viene codificato shiftando di altri 4 bit il timestamp ottenuto nel fattore di scala dei timestamp del target.

2.3.1 Pacchetto forward

Versione e tipo di pacchetto (8 bit) 4 bit ciascuno.

Non assegnato (8 bit) .

Lunghezza (16 bit) La lunghezza del pacchetto di misura, questo header compreso (ma escluso lo header IP).

Fattore di scala dei tempi (4 bit) La collocazione della virgola fissa nella rappresentazione in secondi.

Timestamp (28 bit) Espresso nella stessa scala dei tempi dei timestamps.

Non assegnato (16 bit) .

Porta (16 bit) La porta su cui il probe riceve gli eco.

Firma (32 bit) La chiave di autenticazione richiesta dal target.

```
=====================================
|   |   |         |                 |
| V | T |  vuoto  |   Lunghezza     |
|   |   |         |                 |
=====================================
|    |                              |
| Sc |     Timestamp (probe)        |
|    |                              |
=====================================
|               |                   |
|      vuoto    |     Porta         |
|               |                   |
=====================================
|                                   |
|        autenticazione             |
|                                   |
=====================================
|        |        |        |        |
```

2.3.2 Pacchetto backward

Versione e tipo di pacchetto (8 bit) 4 bit ciascuno.

Non assegnato (8 bit) .

Lunghezza (16 bit) La lunghezza del pacchetto di misura, questo header compreso (ma escluso lo header IP).

Fattore di scala dei tempi (4 bit) La collocazione della virgola fissa nella rappresentazione in secondi.

Timestamp (28 bit) Espresso nella stessa scala dei tempi dei timestamps.

Non assegnato (16 bit) .

Porta (16 bit) La porta su cui il probe riceve gli eco.

Timestamp (32 bit) Corrispondente al tempo di ricezione del messaggio forward (t_1).

Timestamp (32 bit) Corrispondente al tempo di spedizione del messaggio di eco backward (t_2).

```
=====================================
|   |   |         |                  |
| V | T |  vuoto  |   Lunghezza      |
|   |   |         |                  |
=====================================
|    |                               |
| Sc |     Timestamp (probe)         |
|    |                               |
=====================================
|                                    |
|   Timestamp ricezione (target)     |
|                                    |
=====================================
|                                    |
|  Timestamp spedizione (target)     |
|                                    |
=====================================
|        |        |        |         |
```

Tabellina riassuntiva dei timestamp, per un valore del fattore di scala uguale a 0:

```
<--------------------- timestamp NTP ------------------------>

-------|-------|-------|-------*-------|-------|-------|-------|

                            |------- timestamp target ------|
```

```
                            |----- timestamp probe -----|

                                |--- ind. periodo ----------|
```

Tabellina riassuntiva dei timestamp, per un valore del fattore di scala uguale a 8 (verificare):

```
<---------------------- timestamp NTP ------------------------->

-------|-------|-------|-------*-------|-------|-------|-------|

                   |------- timestamp target ------|

                   |----- timestamp probe -----|

                       |--- ind. periodo ----------|
```