Internet Engineering Task Force (IETF)
Request for Comments: 6298
Obsoletes: 2988
Updates: 1122
Category: Standards Track
ISSN: 2070-1721

V. Paxson
ICSI/UC Berkeley
M. Allman
ICSI
J. Chu
Google
M. Sargent
CWRU

June 2011

Computing TCP's Retransmission Timer

Расчет таймера повтора передачи в TCP

Аннотация

Этот документ задает стандартный алгоритм для отправителей TCP[1], который требуется использовать при расчете и поддержке таймера повторной передачи. Документ расширяет обсуждение параграфа 4.2.3.1 в RFC 1122 и обновляет требование к поддержке алгоритма со **следует** на **должно**. Документ отменяет действие RFC 2988.

Статус документа

Документ относится к категории Internet Standards Track.

Документ является результатом работы IETF[2] и представляет согласованный взгляд сообщества IETF. Документ прошел открытое обсуждение и был одобрен для публикации IESG[3]. Дополнительную информацию о стандартах Internet можно найти в разделе 2 в RFC 5741.

Информацию о текущем статусе документа, ошибках и способах обратной связи можно найти по ссылке http://www.rfc-editor.org/info/rfc6298.

1. Введение

Протокол управления передаче (TCP) [Pos81] использует таймер повтора для обеспечения доставки данных в отстутсвие обратной связи от удаленного получателя. Длительность этого таймера называют RTO (retransmission timeout). В RFC 1122 [Bra89] указано, что значение RTO следует рассчитывать в соответствии с [Jac88].

В этом документе представлен алгоритм установки RTO. Кроме того, здесь расширено обсуждение параграфа 4.2.3.1 в 1122 и изменено требование к поддержке алгоритма со **следует** на **должно**. В RFC 5681 [APB09] описан алгоритм, используемый TCP для начала передачи после завершения RTO и отправки повтора. Данный документ не меняет поведения, описанного в RFC 5681 [APB09].

В некоторых ситуациях отправителю TCP может быть выгодно быть более консервативным, нежели позволяет описанный в этом документе алгоритм. Однако TCP **недопустимо** быть более энергичным (агрессивным) нежели позволяет описанный здесь алгоритм. Этот документ отменяет действие RFC 2988 [PA00].

Ключевые слова **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не нужно** (SHALL NOT), **следует** (SHOULD), **не следует** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе интерпретируются в соответствии с [Bra97].

2. Базовый алгоритм

Для расчете текущего RTO отправитель TCP поддерживает две переменные состояний - SRTT (сглаженное время кругового обхода) и RTTVAR (вариации времени кругового обхода). Предполагается дискретность часов G секунд.

Правила расчета SRTT, RTTVAR и RTO приведены ниже.

(2.1) На время измерения периода кругового обхода (RTT) для сегмента, переданного отправителем, ему **следует** установить RTO не более 1 секунды, хотя указанное в (5.5) увеличение тайм-аута сохраняется.

[1]Transmission Control Protocol - протокол управления передачей.
[2]Internet Engineering Task Force.
[3]Internet Engineering Steering Group.

Отметим, что в предыдущей версии документа использовалось начальное значение RTO = 3 сек. [PA00]. Реализация TCP **может** продолжать использовать это значение (или любое другое больше 1 сек.). Изменение нижней границы начального значения RTO обсуждается в Приложении A.

(2.2) При первом измерении RTT (R) хост **должен** установить

```
SRTT <- R
RTTVAR <- R/2
RTO <- SRTT + max (G, K*RTTVAR)
```

где K = 4.

(2.3) При последующем измерении RTT (R') хост **должен** установить

```
RTTVAR <- (1 - beta) * RTTVAR + beta * |SRTT - R'|
SRTT <- (1 - alpha) * SRTT + alpha * R'
```

Используемое для обновления RTTVAR значение SRTT - это значение самого SRTT до обновления с использованием второго назначения, т. е. обновление значений RTTVAR и SRTT **должно** выполняться в указанном порядке.

Указанные выше расчеты **следует** выполнять со значениями alpha=1/8 и beta=1/4 (в соответствии с [JK88]).

После расчета хост **должен** обновить RTO <- SRTT + max (G, K*RTTVAR)

(2.4) Если при расчете RTO получается значение меньше 1 сек., его **следует** округлять до 1 сек.

Традиционно реализации TCP используют грубые часы для измерения RTT и запуска RTO, что ведет к большому минимальному значению RTO. Исследования показывают, что для сохранения умеренности (консервативности) TCP и предотвращения ненужных повторов требуется большое значение минимального RTO [AP99]. Поэтому данная спецификация требует большого минимального значения RTO в качестве умеренного подхода, но признает что в будущем исследования могут показать иное.

(2.5) Для RTO **можно** задать максимальное значение, но оно должно быть не больше 60 сек.

3. Выборка RTT

В TCP **должен** применяться алгоритм Karn [KP87] для выборки RTT, т. е. выборку RTT **недопустимо** делать с использованием повторно передаваемых сегментов (неясно, к какому из сегментов относится полученный отклик). Единственным случаем, когда TCP может безопасно делать выборку RTT с повторно передаваемым сегментом, является применение опции TCP [JBB92], поскольку временные метки устраняют неоднозначность подтверждений.

Традиционно реализации TCP выполняют одно измерение RTT за раз (обычно за период RTT). Однако при использовании временных меток каждое подтверждение (ACK) может служить для выборки RTT. В RFC 1323 [JBB92] предполагается, что соединениям TCP с большим окном насыщения следует выполнять многократную выборку RTT в окне данных для предотвращения эффекта сглаживания RTT. Реализация TCP **должна** выполнять не менее 1 измерения RTT за период RTT (если это не противоречит алгоритму Karn).

При небольшом окне насыщения исследования показывают, что синхронизация каждого сегмента не улучшает точность оценки RTT [AP99]. Кроме того, при многократной выборке за период RTT, значения alpha и beta, определенные в разделе 2 могут содержать неадекватную историю RTT. Метод изменения значений этих констант в настоящее время остается темой для изучения.

4. Дискретность часов

К уровню дискретности часов G, используемому для измерения RTT и разных переменных состояния не предъявляется каких-либо требований. Однако, если при расчете RTO значение K*RTTVAR = 0, элемент дисперсии **должен** округляться до G сек. (т. е. должно использоваться уравнение из 2.3).

```
RTO <- SRTT + max (G, K*RTTVAR)
```

Опыт показывает, что менее дискретные часы (<= 100 мсек) дают несколько лучший результат.

Отметим, что в [Jac88] описано несколько хитростей, которые могут повысить точность при использовании грубых часов. Эти методы широко применяются в современных реализациях TCP.

5. Поддержка таймера RTO

Реализация **должна** поддерживать таймер(ы) так, чтобы повтор передачи сегмента никогда не происходил слишком рано, т. е. до истечения RTO с момента предшествующей передачи этого сегмента. Ниже приведен **рекомендуемый** алгоритм для таймера повторной передачи.

(5.1) При каждой отправке пакета с данными (включая повторы) запускается таймер на время RTO (текущее), если он еще на запущен.

(5.2) Таймер отключается после подтверждения всех ожидающих этого данных.

(5.3) При получении ACK с подтверждением новых данных таймер перезапускается на время RTO (текущее).

При завершении отсчета таймера повтора выполняются перечисленные ниже действия.

(5.4) Повторяется передача самого раннего из неподтвержденных получателем TCP сегментов.

(5.5) Хост **должен** установить RTO не более RTO*2 (выключение таймера). Максимальное значение, указанное в (2.5), может задавать верхнюю границу для удвоения.

(5.6) Запускается таймер на время RTO (удвоенное в соответствии с 5.5 значение RTO).

(5.7) Если отсчет таймера завершается в ожидании ACK для сегмента SYN и реализация TCP применяет RTO меньше 3 сек., значение RTO **должно** быть увеличено до 3 сек. В начале передачи данных (после 3-этапного согласования).

Это отличается от предложенного в прежней версии документа [PA00] и рассмотрено в Приложении A.

Отметим, что после повтора передачи, как только будет выполнено новое измерение RTT (это может произойти лишь при отправке и подтверждении доставки новых данных), выполняется расчет, описанный в разделе 2, включая расчет RTO, что может привести к «коллапсу» RTO (снижение после экспоненциального роста в соответствии с правилом 5.5).

Реализация TCP **может** сбросить SRTT и RTTVAR после многократного изменения (backoff) таймера, поскольку в этой ситуации вполне вероятна некорректность текущих значений SRTT и RTTVAR. После сброса этих значений их следует инициализировать следующей выборкой RTT в соответствии с правилом 2.2, а не 2.3.

6. Вопросы безопасности

Этот документ требует от протокола TCP ожидать в течение заданного интервала перед повтором передачи неподтвержденного сегмента. Атакующий может вынудить отправителя TCP задать большое значение RTO, дополнительно задержав пакет или подтверждение. Однако возможность добавить задержку пакета часто совпадает с возможностью обеспечить потерю пакета, поэтому сложно сказать, может ли злоумышленник получить дополнительное преимущество от задержки по сравнению с простым отбрасыванием некоторых пакетов TCP.

Internet в значительной степени полагается на корректную реализацию алгоритма RTO (а также описанных в RFC 5681 алгоритмов) для сохранения стабильности сети и предотвращении перегрузок. Злоумышленник может вынудить конечные точки TCP более энергично реагировать на перегрузку, создавая подтверждения до того, как получатель реально примет данные, что ведет к небезопасному снижению RTO. Но для этого требуется аккуратно подделать подтверждения, что достаточно сложно, если злоумышленник не может отслеживать трафик на пути между отправителем и получателем. Кроме того, даже если злоумышленник сможет вынудить отправителя к снижению RTO, представляется, что это не поможет усилить атаку (по сравнению с другими повреждениями, которые могут нанести соединению подставные пакеты), поскольку отправитель TCP будет возвращать значение таймера при потере некорректно переданных пакетов в результате реальной перегрузки.

Рассмотренные в RFC 5681 [APB09] вопросы безопасности применимы и к данному документу.

7. Отличия от RFC 2988

Этот документ снижает начальное значение RTO с 3 секунд [PA00] до 1, если только не были потеряны SYN или ACK для SYN, когда следует возвращаться к принятому ранее значению RTO (3 сек.) до начала передачи данных.

8. Благодарности

Описанный в этом документе алгоритм RTO предложен Van Jacobson в работе [Jac88].

Большинство данных, вызвавших замену начального значения RTO (с 3 секунд на 1), получено от Robert Love, Andre Broido и Mike Belshe.

9. Литература

9.1. Нормативные документы

[APB09] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion Control", RFC 5681, September 2009.

[Bra89] Braden, R., Ed., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, October 1989.

[Bra97] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[JBB92] Jacobson, V., Braden, R., and D. Borman, "TCP Extensions for High Performance", RFC 1323, May 1992.

[Pos81] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

9.2. Дополнительная литература

[AP99] Allman, M. and V. Paxson, "On Estimating End-to-End Network Path Properties", SIGCOMM 99.

[Chu09] Chu, J., "Tuning TCP Parameters for the 21st Century", http://www.ietf.org/proceedings/75/slides/tcpm-1.pdf, July 2009.

[SLS09] Schulman, A., Levin, D., and Spring, N., "CRAWDAD data set umd/sigcomm2008 (v. 2009-03-02)", http://crawdad.cs.dartmouth.edu/umd/sigcomm2008, March, 2009.

[HKA04] Henderson, T., Kotz, D., and Abyzov, I., "CRAWDAD trace dartmouth/campus/tcpdump/fall03 (v. 2004-11-09)", http://crawdad.cs.dartmouth.edu/dartmouth/campus/tcpdump/fall03, November 2004.

[Jac88] Jacobson, V., "Congestion Avoidance and Control", Computer Communication Review, vol. 18, no. 4, pp. 314-329, Aug. 1988.

[JK88] Jacobson, V. and M. Karels, "Congestion Avoidance and Control", ftp://ftp.ee.lbl.gov/papers/congavoid.ps.Z.

[KP87] Karn, P. and C. Partridge, "Improving Round-Trip Time Estimates in Reliable Transport Protocols", SIGCOMM 87.

[PA00] Paxson, V. and M. Allman, "Computing TCP's Retransmission Timer", RFC 2988, November 2000.

Приложение A. Обоснование снижения Initial RTO

Выбор разумного начального значения RTO требует учета приведенных ниже противоречивых требований.

1. Начальное значение RTO следует делать достаточно большим, чтобы для большинства сквозных путей избежать ненужных повторов и связанного с ними негативного влияния на производительность.

2. Начальное значение RTO должно быть достаточно мало, чтобы обеспечить своевременное обнаружение потерь, пока еще не определено значение RTT.

Традиционно в TCP устанавливается начальное значение RTO 3 секунды [Bra89] [PA00]. Данный документ предлагает снизить его до 1 секунды с учетом приведенных ниже оснований.

- Современные сети быстрее, чем были в момент выбора начального значения RTO 3 секунды.

- Исследования показали, что время кругового обхода более чем для 97,5% крупномасштабных наблюдений составляет менее 1 сек. [Chu09], что позволяет счесть значение 1 сек. подходящим по критерию 1 см. (выше).

- Кроме того, исследования показали, что наблюдаемая частота повторов передачи при 3-этапном согласовании составляет около 2%. Это говорит, что снижение начального RTO даст преимущества большинству соединений.

- Однако примерно 2,5% соединений, исследованных в [Chu09], использовали RTT больше 1 секунды. Для таких соединений 1 секунда в качестве начального RTO гарантирует повтор передачи при организации соединения (он может оказаться ненужным).

  Для таких случаев документ предлагает использовать прежнее значение RTO (3 сек.) в фазе передачи данных. Поэтому влияние ложных повторов передачи будет незначительным - (1) в сеть передается дополнительный пакет SYN и (2) в соответствии с RFC 5681 [APB09] начальное окно насыщения будет ограничено 1 сегментом. Хотя обстоятельство (2) явно ставит такие соединения в невыгодное положение, этот документ по меньшей мере задает сброс RTO, чтобы соединение не сталкивалось постоянно с проблемами из-за короткого тайм-аута (при RTT больше 3 сек. проблемы в соединении сохранятся, но это не создает новых проблем для TCP).

  Кроме того, отмечено, что при использовании временных меток TCP может сделать выборку RTT даже при наличии ложных повторов, облегчая сходимость к корректной оценке RTT для значений больше 1 секунды.

В качестве дополнительной проверки результатов [Chu09] были проанализированы трассировки поведения клиентов, собранные в разное время в 4 точках, как показано в таблице.

Имя	***Период измерения***	***Число пакетов***	***Число соединений***	***Число клиентов***	***Число серверов***
LBL-1	10.2005 - 03.2006	292M	242K	228	74K
LBL-2	11.2009 - 02.2010	1.1B	1.2M	1047	38K
ICSI-1	11-18.09.2007	137M	2.1M	193	486K
ICSI-2	11-18.09.2008	163M	1.9M	177	277K
ICSI-3	14-21.09.2009	334M	3.1M	170	253K
ICSI-4	11-18.09.2010	298M	5M	183	189K
Dartmouth	4-21.01.2004	1B	4M	3782	132K
SIGCOMM	17-21.08.2008	11.6M	133K	152	29K

Данные LBL были получены в Lawrence Berkeley National Laboratory, ICSI - в International Computer Science Institute, SIGCOMM - из беспроводной сети для участников конференции SIGCOMM 2008 и Dartmouth - из беспроводной сети Dartmouth College. Две последних базы данных доступны в хранилище CRAWDAD [HKA04] [SLS09]. В таблице указаны даты сбора данных, число отобранных пакетов, число наблюдаемых соединений TCP, число отслеживаемых локальных клиентов и число удаленных серверов, с которыми были контакты. Рассматривались только соединения, инициированные вблизи точки отслеживания.

Анализ этих данных показывает, что распространенность повторной передачи SYN составляет от 0,03% (ICSI-4) приблизительно до 2% (LBL-1 и Dartmouth).

Затем данные были проанализированы для определения числа добавочных и ложных повторов передачи, которые могли бы возникнуть при начальном RTO в 1 сек. В большинстве баз данных доля соединений с ложными повторами была меньше 0,1%. Однако в данных Dartmouth около 1,1% соединений передавали ненужные повторы при снижении начального значения RTO. Это было связано с тем, что наблюдаемая сеть была беспроводной и в ней возникали добавочные задержки из-за радиочастотных эффектов.

Очевидно, что преимущество повторной передачи потерянных пакетов SYN растет при уменьшении начального RTO. В рассмотренных данных доля соединений с повтором SYN, позволившим повысить производительность по меньшей мере на 10% за счет сниженного в соответствии с этим документом начального значения RTO, составила от 43 (LBL-1) до 87% (ICSI-4). Доля соединений, которые могут повысить производительно как минимум на 50%, составила от 17 (ICSI-1 и SIGCOMM) до 73% (ICSI-4).

На основании доступных данных сделан вывод о том, что снижение начального значения RTO обеспечивает преимущество для многих соединений и лишь немногим наносит вред.

Адреса авторов

Vern Paxson

ICSI/UC Berkeley

1947 Center Street

Suite 600

Berkeley, CA 94704-1198

Phone: 510-666-2882

EMail: vern@icir.org

http://www.icir.org/vern/

Mark Allman

ICSI

1947 Center Street

Suite 600

Berkeley, CA 94704-1198

Phone: 440-235-1792

EMail: mallman@icir.org

http://www.icir.org/mallman/

H.K. Jerry Chu

Google, Inc.

1600 Amphitheatre Parkway

Mountain View, CA 94043

Phone: 650-253-3010

EMail: hkchu@google.com

Matt Sargent

Case Western Reserve University

Olin Building

10900 Euclid Avenue

Room 505

Cleveland, OH 44106

Phone: 440-223-5932

EMail: mts71@case.edu

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru