

А. С. Свиридов

Харківський національний університет радіоелектроніки, Харків, Україна

МЕТОД ПІДВИЩЕННЯ НАДІЙНОСТІ ПЕРЕДАЧІ ДАНИХ ПІД ЧАС ВИКОРИСТАННЯ ПРОТОКОЛУ TCP FREEZE ШЛЯХОМ ПІДВИЩЕННЯ ПРОПУСКНОЇ СПРОМОЖНОСТІ

У даній статті розглянуто протокол TCP, розглянуті його недоліки, а, зокрема, розриви TCP з'єднань. Розглянуто існуючі модифікації, які дозволяють усунути такі недоліки. Виявлено недоліки існуючих методів. Запропоновано метод управління передачею даних, в протоколі TCP Freeze, що дозволяє підвищити надійність передачі даних. Такий метод заснований на динамічній зміні, в залежності від умов, параметрів протоколу, які в своїй незмінній версії задаються статично. Висновки. Ефективність методу управління передачею даних, що використовується в протоколі TCP Freeze, може бути поліпшена в бездротовому гетерогенному середовищі за допомогою варіювання параметрів. Виявлено, що за умови рівності параметрів і їх значенні близькому до 20, пропускні спроможності протоколів TCP Freeze і TCP Reno будуть практично рівними. При більш низьких значеннях параметрів протокол TCP Reno показує кращу ефективність, ніж TCP Freeze, а при великих значеннях параметрів пропускна здатність протоколу TCP Freeze буде перевищувати таку у TCP Reno. Використання запропонованого методу дозволяє зменшити час передачі даних в мережі внаслідок реалізації алгоритму управління розміром плаваючого вікна протоколу, що згладжує флуктуації останнього і приводить до меншого числа повторних передач пакетів даних, а також обробку події хендовера.

Ключові слова: мережа, плаваюче вікно, TCP, протокол, TCP Freeze, передача пакетів, фази.

Вступ

Постановка задачі та аналіз літератури. Мобільні користувачі втрачають TCP-з'єднання, коли вони переміщуються або їх IP-адреси змінюються, або деякі вузли вздовж шляху маршрутизації зазнають невдачі. Крім того, мінливість якості бездротового середовища може привести до тимчасової втрати з'єднання, що призведе до завершення з'єднання TCP. Проблеми такого ж характеру зустрічаються і в кабельних мережах. Така поведінка мережі найчастіше для користувача буде досить незручною. Наприклад, після завантаження протягом 20 хвилин для користувача було би краще, щоб FTP-клієнт продовжував завантаження після тимчасового відключення, а не запуслав завантаження файлу спочатку. Іншим прикладом є випадок, коли користувачі роблять банківський переказ або виведення коштів, а лінія зв'язку переривається на кілька секунд, після чого користувачам необхідно виконувати всі дії спочатку. Таким чином, існує потреба підтримувати простими і ефективними способами роботу TCP з'єднань.

Головні недоліки роботи TCP-з'єднань. Першим недоліком є сталість TCP-з'єднання та причини його відключення. TCP-з'єднання переривається, якщо немає відповіді TCP keep-alive з іншого кінця в межах деякого порога. Тайм-аут Keep-alive – обов'язковий параметр, браузері, наприклад, Internet Explorer, часто встановлюють його на 1 секунду. Якщо відповіді не отримано протягом цього інтервалу з'єднання закривається. Ще одним недоліком є тайм-аут з'єднання. [1, 2] Як правило, цей коефіцієнт визначається розробником; немає деякого конкретного стандарту. Робота, пов'язана з проблемою постійного зв'язку, була представлена Юнгуан Чжан і Сон Дао [3]. Вони запропонували зіставити логічні та фізичні кінцеві точки з точки зору віртуальної адреси. Ця віртуальна адреса повинна складатися

з 3-х кортежів: дескриптор сокета, IP-адреса і номер порту TCP. Подібно концепції віртуальної адреси, був введений CID для однозначної ідентифікації з'єднання. CID може складатися з безлічі компонентів, таких як випадкове число, інформація про стан сокета, IP-адреса і номер порту TCP [4]. Інший спосіб – додати новий рівень поверх транспортно рівня, щоб приховати відключення безпосередньо з програми. Замість цього може бути вставлений спеціальний рівень під назвою Mobile Socket Layer [5].

Методи, описані вище, модифікують клієнт, або сервер, або обидва, для додавання додаткового рівня, який приховує відключення від TCP. Однак змінювати всі вузли недоцільно. Ще однією з ідей було введення концепції, яка перенаправляє весь трафік через додатковий проксі-сервер [6]. Цей проксі-сервер підтримує всі TCP-з'єднання для обох вузлів. Проксі-сервер може прозоро підключати з'єднання до альтернативних серверів, забезпечуючи при цьому узгодженість з'єднань.

Модифікація TCP Freeze, була вперше використана щоб запобігти перериванню локального TCP-сокета через збій з'єднання. Він також використовується в бездротових мережах з метою виявлення втрат і перевантаження [7, 8]. Метод TCP Freeze реалізований в стандартних реалізаціях TCP в тому сенсі, що, коли кінцевий хост-одержувач не має достатньої кількості вільного простору, він повідомляє що треба припинити відправку більшої кількості пакетів.

Основне призначення TCP Freeze – «заморозити» всі таймери TCP, коли вікно прийому заповнене. На сьогоднішній день не існує однозначного рішення, що дозволяє одночасно зі спільним використанням різних реалізацій транспортних протоколів в повному обсязі реалізувати потенціал пропускної здатності мережі.

У статті проводиться детальний аналіз приведених вище проблем і на його основі пропонується мо-

дифікація методу управління передачею даних, що використовується в протоколі TCP Freeze, що дозволяє у значній мірі більшість з них усунути. Ефективність запропонованої модифікації буде залежить від значень основних параметрів протоколу та від методів управління буферами, реалізованими в маршрутизаторах [9]. Крім методу управління перевантаженнями, що відрізняється від методу, реалізованого в інших модифікаціях протоколу TCP, дозволяє досягти більшої ефективності внаслідок меншої кількості повторних передач пакетів для більш коректної обробки події хендвера [10]. Протокол TCP Freeze реалізує метод, знижуючи вплив сполук з відносно більшої затримки поширення сигналу в з'єднанні.

Модифікацію протоколу TCP Freeze можна реалізувати, здійснивши декомпозицію методу управління передачею даних на різні фази функціонування, визначивши характер їх впливу на основні характеристики протоколу. Більш того, ефективність протоколу також залежить від дисципліни обслуговування череди маршрутизаторів, через які проходить встановлений маршрут. Таким чином, завжди існує незбійний компроміс між взаємною справедливістю та ефективністю використовуваних протоколів транспортного рівня, який залежить, зокрема, від використовуваних дисциплін управління чергами проміжних маршрутизаторів [11].

Таким чином, завдання створення та дослідження методів, що дозволяють протоколам транспортного рівня сучасних мереж досягати більшої пропускну здатності в різноманітних мережевих сценаріях, є актуальним.

Метою даної статті є розробка методу управління передачею даних для використання протоколом TCP Freeze з метою підвищення його пропускну здатності в сучасних мережах для підвищення надійності передачі.

Результати досліджень

В даний час існує цілий ряд протоколів, які керують передачею даних. Більшість з них є байт-орієнтованими протоколами з надійною доставкою даних і здійснюють доставку даних по маршруту з встановленням з'єднання. За останні роки були запропоновані різні модифікації, спрямовані на поліпшення протоколу TCP, однак вони переважно стосувалися фази відновлення втрат (протоколи TCP Tahoe, TCP Reno, TCP NewReno, M-TCP, I-TCP, Soop-TCP і TCP SACK) [12]. Значною мірою механізм управління передачею даних був змінений лише в протоколі TCP Freeze. Це досягнуто використанням оцінки доступного розміру смуги пропускання бездротового каналу, як ознаки перевантаження. Початок перевантаження визначається шляхом вимірювання різниці між поточним значенням часу передачі пакету даних в з'єднанні (T_n) і його мінімальним значенням, з подальшим коригуванням інтенсивності відправки пакетів в мережу (у відповідності зі значенням цієї різниці).

Реалізація механізму управління перевантаженнями протоколу TCP Freeze зводиться до послідовного виконання наступних кроків:

- 1) обчислення очікуваної (теоретично) швидкості відправки пакетів джерелом в мережу,
- 2) обчислення реальної швидкості відправки пакетів джерелом в мережу,
- 3) оцінка джерелом кількості пакетів, що знаходяться в черзі маршрутизатора, виконувана для кожного прийнятого пакета підтверджень,
- 4) обчислення чергового значення розміру плаваючого вікна на підставі різниці швидкостей, отриманої на попередньому кроці та значення розміру плаваючого вікна в попередній момент часу [13].

При близькості значень, очікуваної та реальної швидкостей джерела, з'єднання не може використовувати весь доступний розмір смуги пропускання каналу. Для збільшення коефіцієнта використання каналу необхідно збільшувати швидкість відправки даних джерелом. У разі, коли реальна швидкість значно нижче очікуваної, мережа стає перевантаженою та швидкість відправки даних джерелом потрібно знижувати. Після установки з'єднання, протокол TCP Freeze може перебувати в двох фазах: фазі повільного старту і в фазі запобігання перевантажень [14].

У фазі повільного старту, умовою входу в яку є успішне встановлення з'єднання, протокол TCP Freeze залишається до досягнення розміром плаваючого вікна деякого порогового значення γ (зазвичай рівного 1). Поки виконується умова $\Delta < \gamma$, відбувається додавання обсягу плаваючого вікна на один пакет за кожен другий інтервал часу передачі пакету. Отже, під час фази повільного старту, проводиться експоненціальне збільшення розміру плаваючого вікна протоколу TCP Freeze. Коли розмір плаваючого вікна W досягає порогового значення виходу з фази повільного старту, або починає дотримуватися умова $\Delta > \gamma$, протокол TCP Freeze увійде в фазу запобігання перевантажень. У фазі запобігання перевантажень протокол TCP Freeze функціонує на основі двох параметрів – α та β (зазвичай рівних значень 1 і 3, відповідно), змінюючи розмір плаваючого вікна у відповідності з виразом:

$$W(t) = \begin{cases} W(t-1) + 1, & \text{якщо } \Delta < \alpha; \\ W(t-1) - 1, & \text{якщо } \Delta > \beta; \\ W(t-1), & \text{в інших випадках,} \end{cases}$$

Втрата пакету може виявлятися внаслідок однієї з двох подій:

- закінчення часу таймера передачі пакета;
- послідовне отримання трьох однакових пакетів-підтверджень.

У першому випадку граничне значення для виходу з фази повільного старту встановлюється в значення половини значення поточного розміру плаваючого вікна, а розмір плаваючого вікна – в 1 пакет (або 2 пакети в деяких реалізаціях) та потім протокол TCP Freeze знову входить у фазу повільного старту. У іншому випадку джерело виконує швидку повторну передачу та швидке відновлення. Фактично, протокол TCP Freeze реалізує покращений механізм швидкої повторної передачі, заснований на більш точному внутрішньому таймері прото-

колу. Після виходу з фази, протокол TCP Freeze встановлює значення розміру свого плаваючого вікна в $3/4$ його поточного значення і знову входить в фазу запобігання перевантажень.

У гетерогенних мережевих сценаріях втрати пакетів можуть відбуватися при перевищенні сумарної кількості трафіку (як переданого, так і службового) ємності сполуки (C), що є критичною ділянкою на встановленому маршруті. Відсутності втрат, за такої умови буде виглядати таким чином:

$$\frac{W(t)}{T_{oc}(t)} + \frac{W_1(t)}{T_{oc}(t)} \cdot C,$$

тобто у стаціонарній точці

$$W(t) = T_{oc}(t) \cdot C - K/\sqrt{p(t)}.$$

Таким чином, обчислення розмірів плаваючого вікна для TCP Freeze гетерогенного мережного середовища буде виглядати таким чином.

$$W(t) = \ell(t) + T_p \cdot C - K/\sqrt{p(t)} + 1$$

У методі управління передачею даних, який використовується в протоколі TCP Freeze, величини α та β є константами. Завданням є таке управління розміром плаваючого вікна, при якому число пакетів, що знаходяться в буферній пам'яті вузлів на встановленому маршруті з'єднання, буде знаходитися в інтервалі між значеннями α та β . Керуючись таким принципом, в запропонованому методі реалізовано динамічна зміна значень параметрів α та β . Методи, що використовуються в фазі повільного старту та відновлення від перевантажень, залишаються незмінними (аналогічними оригінальним протоколом TCP Freeze), а змінам піддається тільки метод, який використовується в фазі запобігання перевантажень.

Протокол TCP Freeze виробляє обчислення дійсної швидкості кожен раз при успішній передачі плаваючого вікна. Функціонування запропонованого алгоритму, можна описати наступним чином. Відразу після встановлення з'єднання параметрам α та β присвоюються деякі початкові значення, α_0 та β_0 . Після успішної передачі кожного плаваючого вікна даних, проводиться динамічна зміна значень α та β на одиницю, відповідно до таких умов:

– якщо $T_n(t) > T_n(t-1)$ и $W(t) \leq W(t-1)$, то проводиться інкрементування значень α та β ;

– якщо $T_n(t) \leq T_n(t-1)$ и $W(t) > W(t-1)$, то проводиться декрементування значень α та β .

У разі виникнення втрат пакетів, значенням параметрів присвоюються їх початкові значення, після чого виконання алгоритму починається знов.

Для перевірки роботи методу була порівняна робота звичайного TCP Freeze, з конкурентоспроможним TCP Reno, та їх взаємодія. Моделювання протоколів TCP Freeze та TCP Reno в гетерогенному мережному середовищі показало, що протокол TCP Reno здатний досягати приблизно на 45% більше пропускної здатності, ніж TCP Freeze. Протокол TCP Freeze проявляє кращу пропорційність при роз-

поділі вільних ресурсів мережі, ніж TCP Reno, в однорідних мережних середовищах, але поводить себе набагато гірше ніж TCP Reno в гетерогенних.

Запропонований метод дозволяє дістатися динаміки протоколу TCP Freeze, порівнянної з такою для TCP Reno, в разі гетерогенного мережного середовища, в той же час зберігаючи її незмінною в разі гомогенного мережного середовища.

Даний метод також має ряд таких переваг:

– модифікації виробляються тільки в алгоритмі джерела, що не тягне за собою змін в структурі й інших елементах мережі;

– не проводиться додавання нових змінних в алгоритм, отже, не виникає потреби їх коректного завдання, оскільки запропонований алгоритм адаптивний до стану ділянки мережі, через яку встановлено маршрут з'єднання;

– результат роботи запропонованого модифікованого алгоритму управління передачею даних практично не відрізняється від оригінального, реалізованого в протоколі TCP Freeze, в разі бездротового гомогенного мережного середовища.

Висновки

Таким чином, в статті запропоновано метод управління передачею даних, призначений для використання в протоколі транспортного рівня бездротових мереж TCP Freeze.

Ефективність методу управління передачею даних, що використовується в протоколі TCP Freeze, може бути поліпшена в бездротовому гетерогенному середовищі за допомогою варіювання параметрів α та β . Виявлено, що за умов рівності параметрів і їх значенні близькому до 20 (значення вираховане з процесу моделювання), пропускні спроможності протоколів TCP Freeze і TCP Reno будуть практично рівними. При більш низьких значеннях параметрів α та β , протокол TCP Reno показує кращу ефективність, ніж TCP Freeze, а при великих значеннях параметрів α та β пропускна здатність протоколу TCP Freeze буде перевищувати таку у TCP Reno. Однак, у розмірах плаваючих вікон обох сполук протоколом TCP Freeze спостерігаються варіації, які обумовлені рівністю значень параметрів α та β . При збільшенні значень параметрів α та β збільшуються варіації значень розміру плаваючого вікна, що негативно впливає на загальну ефективність протоколу. Зокрема, простежується закономірність збільшення варіацій значень розміру плаваючого вікна при збільшенні параметра β . Збільшення параметра α при невеликих значеннях β практично не впливає на величину розкиду.

Використання методу дозволяє зменшити час передачі даних в мережі внаслідок реалізації алгоритму управління розміром плаваючого вікна протоколу, що згладжує флуктуації останнього і приводить до зменшення повторних передач пакетів даних, а також обробку події хендовера. **Напрямок подальших досліджень** – вирішення задачі оптимальності вибору початкових параметрів протоколу TCP Freeze.

СПИСОК ЛІТЕРАТУРИ

1. Таненбаум Э. Компьютерные сети. 4-е изд. [Текст] / Э. Таненбаум // СПб.: Питер, 2003. – 992 с.
2. Телекоммуникационные системы и сети: Учебн. пос. В 3-х т. Т. 3. Мультисервисные сети / В.В. Величко, Е.А. Субботин, В.П. Шувалов, А.Ф. Ярославцев, под ред. В.П. Шувалова. – М.: Горячая линия – Телеком, 2005. – 592 с.
3. Y. Zhang and S. Dao, "A Persistent Connection Model for Mobile and Distributed Systems," in Proc. Int. Conf. on Computer Communications and Networks., 1995, p. 300.
4. Кучук Г.А. Управление ресурсами инфокоммуникаций [Текст] / Г.А. Кучук, Р.П. Гахов, А.А. Пашнев.
5. Manner J. Mobility Related Terminology [Текст] / J. Manner, M. Kojo. – Network Working Group, RFC 3753. 2004
6. Гладун, А. Я. Аналіз моделей процедур хендвера в бездротових мережах, базованих на технології ATM / А. Я. Гладун, О. І. Проценко // Наукові записки НаУКМА. – 2003. – Т. 21: Комп'ютерні науки. – С. 85–93.
7. Кучук Г.А. Метод прогнозування фрактального трафіка / Г.А. Кучук, О.О. Можасв, О.В. Воробйов // Радіоелектронні і комп'ютерні системи. – 2006. – № 6 (18). – С. 181-188.
8. Кучук, Г.А. Модель процесса эволюции топологической структуры компьютерной сети системы управления объектом критического применения / Г.А. Кучук, А.А. Коваленко, А.А. Янковский // Системы обработки информации: сборник научных трудов. – Х.: ХУ ВС, 2014. – Вып. 7 (123). – С. 93 – 96.
9. Bakre, A. I-TCP: Indirect TCP for mobile hosts [Текст] / A. Bakre, B. Badrinath // Proceedings of the 15th International Conference on Distributed Computing Systems (ICDCS), May 30 – June 2, 1995, Canada. – Vancouver, 1995. – P. 136–143.
10. R. Ekwall, P. Urban, and A. Schiper, "Robust TCP Connections for Fault Tolerant Computing," in Proc. Int. Conf. on Parallel and Distributed Systems., 2003, pp. 501-508.
11. W-E. Eddy and J. Ishac, "Location Management in a Transport Layer , Mobility Architecture," NASA/TM-2005-213844., Aug. 2005.
12. Kuchuk, G., Kharchenko, V., Kovalenko, A. and Ruchkov, E. (2016), "Approaches to Selection of Combinatorial Algorithm for Optimization in Network Traffic Control of Safety-Critical Systems", Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2016), P. 384–389.
13. D. Funato, K. Yasuda, and H. Tokuda, "TCP-R: TCP mobility support for continuous operation," in Proc. IEEE Int. Conf. on Network Protocols., 1997, pp. 229-236.
14. D-A. Maltz and P. Bhagwat, "MSOCKS: An Architecture for Transport Layer Mobility," in Proc. IEEE Conf. on Computer Communications., 1998, vol. 3, pp. 1037-1045.

Рецензент: д-р техн. наук, проф. М. А. Павленко,

Харківський національний університет Повітряних Сил імені Івана Кожедуба, Харків

Received (Надійшла) 22.02.2018

Accepted for publication (Прийнята до друку) 16.05.2018

Метод повышения надежности передачи данных при использовании протокола TCP Freeze путем повышения пропускной способности

А.С. Свиридов

В статье рассмотрен протокол TCP, рассмотрены его недостатки, а в частности, разрывы TCP соединений. Рассмотрены существующие модификации, которые позволяют убрать такие недостатки. Выявлена недостатки существующих методов. Предложен метод управления передачей данных, в протоколе TCP Freeze, что позволяет повысить надежность передачи данных. Такой метод основан на динамическом изменении, в зависимости от условий, параметров протокола, которые в его исходной версии задаются статически. Выводы. Эффективность метода управления передачей данных, используемый в протоколе TCP Freeze, может быть улучшена в беспроводном гетерогенной среде с помощью варьирования параметров. Выявлено, что при условии равенства параметров и их значения близком к 20 пропускные способности протоколов TCP Freeze и TCP Reno будут практически равны. При более низких значениях параметров протокол TCP Reno показывает лучшую эффективность, чем TCP Freeze, а при больших значениях параметров пропускная способность протокола TCP Freeze будет превышать таковую в TCP Reno. Использование предложенного метода позволяет уменьшить время передачи данных в сети вследствие реализации алгоритма управления размером плавающего окна протокола, сглаживает флуктуации последнего и приводит к меньшему числу повторных передач пакетов данных, а также обработку события хендвера.

Ключевые слова: сеть, плавающее окно, TCP, протокол, TCP Freeze, передача пакетов, фазы.

Method of growth of data transmission reliability time of use of TCP Freeze protocol by increase transmission capacity

A.S. Svyrydov

This article discusses the TCP protocol, its shortcomings are examined, and in particular the TCP connections breaks. Existing modifications that allow to remove such shortcomings are considered. The nonoptimality of existing methods is revealed. The method of data transmission control is proposed in the TCP Freeze protocol, which allows to increase the data transmission time. This method is based on dynamic change, depending on the conditions, protocol parameters, which are statically assigned in its original version. Conclusions The efficiency of the data transfer control method used in the TCP Freeze protocol can be improved in a wireless heterogeneous environment by varying the parameters. It was found that if the parameters and their values are equal to the nearest 20, the TCP Freeze and TCP Reno bandwidths will be practically equal. At lower values of parameters TCP Reno shows better performance than TCP Freeze, and at high values of parameters the TCP Freeze bandwidth will exceed TCP Reno's. The use of the proposed method allows to reduce the time of data transmission in the network due to implementation of the algorithm for controlling the size of the floating window of the protocol, which smooths the fluctuations of the latter and leads to a smaller number of retransmissions of data packets, as well as handling the event handover.

Keywords: network, floating window, TCP, protocol, TCP Freeze, packet transmission, phase.