

Малий В.О.

МОДЕЛИРОВАНИЕ И СРАВНЕНИЕ ХАРАКТЕРИСТИК ПРОТОКОЛОВ НА БАЗЕ АППАРАТА СЕТЕЙ ПЕТРИ

За последние годы протокол управления передачей (TCP) в сочетании с интернет протоколом (IP) стали наиболее распространенной формой организации сетей между компьютерами. В связи с ведущимися разработками высокоскоростных сетей и приложений, которые используют TCP, проблемы поведения TCP имеют повышенную важность [1]. Поведению (т.е. производительность, количество потерянных пакетов, стабильность и т.д.) во время начальной стадии необходимо уделять отдельное внимание, так как большинство приложений используют TCP для коротких передач, которые завершаются еще до того как TCP перейдет к устойчивому поведению [2].

Один из подходов в исследовании проблем поведения заключается в изучении работы конкретной реализации TCP. Таким образом, автором, с помощью моделирования, детально изучается поведение нескольких реализаций TCP (Tahoe, Reno, SACK, Vegas [2,3]). На основе этих наблюдений предлагается несколько простых изменений, которые при определенной ситуации существенно улучшают поведение протокола.

На основании особенностей реализации алгоритмов, была построена модель для каждого протокола. Важным моментом в построении модели является сверхточное отображение поведения протокола в реальной сети с тем, чтобы результаты, полученные экспериментальным путем, можно было переносить в реальную сеть.

В ходе моделирования были получены следующие результаты.

- При экспоненциальном увеличении окна во время Медленного старта в протоколе TCP Tahoe, если размер буфера меньше $1/3$ произведения ширины полосы на задержку распространения, то происходит переполнение буфера, что приводит ко второму Медленному старту и уменьшению производительности.
- TCP Reno пытается избежать этого эффекта уменьшая окно вдвое при обнаружении потери. Хотя это и обеспечивает лучшую производительность при идеальных условиях, TCP Reno в своей нынешней форме слишком уязвим, что бы быть заменой Tahoe.
- Использование "грубого" таймера приводит к катастрофической потере производительности, если таймаут является идентификатором потери пакета. Поэтому необходимо использование более сложной обработки оценки времени кругового обращения.
- Идеальное поведение при множественной потере пакетов представляет TCP SACK, однако использование данной реализации возможно лишь в тех случаях когда оба конца соединения (и отправитель и получатель) поддерживают данную опцию.
- При отсутствии опции SACK возможна ситуация многократной быстрой передачи, что приводит к значительному уменьшению

производительности. Во избежание данного эффекта необходимо дополнительно использовать переменную "send_high", отражающую максимальный номер посланного пакета, либо, что более универсально, анализировать время отправления лже- потерянного пакета. Литература:

1. Van Jacobson, "Description of TCP Reno"

2. RFC 2581- TCP Congestion control

3. RFC 2582 - The NewReno Modification to TCP's Fast Recovery Algorithm