

Секция «Вычислительная математика и кибернетика»

О подходе к обеспечению отказоустойчивости контроллера в программно-конфигурируемых сетях

Пашков Василий Николаевич

Аспирант

Московский государственный университет имени М.В. Ломоносова, Факультет вычислительной математики и кибернетики, Москва, Россия

E-mail: vasyapashkov@gmail.com

В настоящее время активно развивается новый подход к построению архитектуры компьютерных сетей, называемый программно-конфигурируемыми сетями (ПКС) [1]. Наиболее известной реализацией подхода ПКС является технология OpenFlow [2]. Подход заключается в вынесении функций управления трафиком и управления состоянием из сетевых элементов (маршрутизаторов, коммутаторов) в программируемый контроллер, функционирующий на физическом сервере. Контроллер, таким образом, обеспечивает логически-централизованное управление сетью с помощью приложений, формирующих правила управления потоками данных на основе глобального представления сети (ГПС). ГПС содержит топологию сети, состояние сетевых элементов и таблиц передачи данных (forwarding table) коммутаторов. Однако контроллер в такой архитектуре сети является единой точкой отказа, поэтому актуальной является проблема обеспечения отказоустойчивости контроллера в ПКС сети.

Одним из подходов к решению этой проблемы является построение контроллера как распределенной системы, в которой в случае отказа одного из контроллеров управление его сегментом перехватывается другим контроллером. В настоящее время известно несколько подходов к построению распределенного контроллера: HyperFlow [3], Onix [4], Kandoo [5].

В данной работе предлагается новый подход к построению распределенного контроллера, обеспечивающий масштабируемость и повышающий отказоустойчивость ПКС сети. Поскольку согласно [2] контроллер может быть соединен с несколькими коммутаторами, а коммутатор может быть соединен с несколькими контроллерами. В каждый момент времени коммутатор управляется одним контроллером – мастером, остальные контроллеры, поддерживающие с ним соединения, являются подчиненными. Подчиненные контроллеры могут только отслеживать состояние коммутатора. Контроллеры, контролирующие состояние одного коммутатора, можно объединить в групповой контроллер (ГК). Для повышения отказоустойчивости соединения между контроллерами ГК дублируются. Для обеспечения быстрого перехвата управления между контроллерами ГК в случае отказа они должны иметь согласованное представление фрагментов ГПС. ГПС для каждого контроллера хранится в специальной структуре данных – NIB (Network Information Base). Таким образом, необходимо обеспечить согласованность NIB контроллеров. Обновление содержимого NIB разных контроллеров может происходить как из-за получения сообщений от контроллеров ГК, так и от коммутатора. Сообщения могут теряться, дублироваться из-за ненадежности каналов связи, может изменяться их порядок вследствие задержек.

В рамках данной работы на основе проведенного анализа методов репликации данных в распределенных системах разработан алгоритм синхронизации реплик NIB на

основе алгоритма Paxos [6], который позволяет обеспечивать согласованность данных в случае отказа контроллеров, в том числе и мастер-контроллера.

Литература

1. Open Networking Foundation. Software-Defined Networking: The New Norm for Networks. ONF White Paper, April 13, 2012.
2. Open Networking Foundation. OpenFlow Switch Specification. Version 1.3.1 (Wire Protocol 0x04) September 6, 2012.
3. A. Tootoonchian and Y. Ganjali. HyperFlow: A Distributed Control Plane for OpenFlow. In Proc. INM/WREN, San Jose, CA, April 2010.
4. T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, T. Hama, and S. Shenker, “Onix: A distributed control platform for large-scale production networks,” in OSDI, Oct 2010.
5. S. Hassas Yeganeh, Y. Ganjali. Kandoo: a framework for efficient and scalable offloading of control applications. Proceedings of the first workshop on Hot topics in software defined networks (HotSDN '12), pp. 19-24.
6. L. Lamport. Fast Paxos. Distributed Computing, 19(2):79–103, Oct. 2006.