

МАТЕМАТИЧЕСКИЕ МОДЕЛИ ОБЛАЧНОЙ СИСТЕМЫ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ОБРАЗОВАТЕЛЬНОГО РЕСУРСНОГО ЦЕНТРА

Полежаев П. Н., Коннов А. Л., Кириллов А. С.

Кафедра компьютерной безопасности и математического обеспечения информационных систем, центр информационных технологий, кафедра геометрии и компьютерных наук, Оренбургский государственный университет

Оренбург, Российская Федерация

E-mail: {newblackpit, andrey_konnov}@mail.ru, alexey.s.kirillov@gmail.com

В работе описываются построенные математические модели облачной системы и программного обеспечения образовательного ресурсного центра, который используется для организации коллективного удаленного доступа учащихся и студентов к платному программному обеспечению, установленному в облаке. Данные модели будут служить основой для разрабатываемых алгоритмов создания виртуальных машин со всем необходимым программным обеспечением, алгоритмов планирования экземпляров виртуальных машин классов с учетом ограничений лицензирования ПО, а также методов маршрутизации сетевого трафика внутри центра обработки данных облачной системы с использованием программно-конфигурируемых сетей.

ВВЕДЕНИЕ

В настоящее время во многих странах СНГ большинство образовательных учреждений функционирует в условиях недостаточного финансирования на закупку ПО, необходимого в учебном процессе, что негативно сказывается на качестве образования.

Данная проблема может быть решена созданием образовательных ресурсных центров (ОРЦ) с возможностью удаленного доступа к ним образовательных учреждений. Каждый ОРЦ представляет собой центр обработки данных (ЦОД), поверх него развернута облачная система с виртуальными машинами, содержащими все необходимое платное ПО.

Преподаватель образовательного учреждения с помощью Web-интерфейса ОРЦ может создать виртуальный класс, доступ к которому будет осуществляться удаленно с компьютеров образовательного учреждения, что соответствует принципу DaaS (Desktop as a Service) в облачной терминологии. При создании класса преподаватель может указать нужное количество экземпляров виртуальных машин, которые строятся на основе одного из существующих публичных дисковых образов или с нуля. Преподаватель также указывает все необходимое ПО для автоматической установки, а затем он выбирает время проведения занятий в данном виртуальном классе.

В зарезервированное время согласно расписанию виртуальные машины класса будут развернуты. Пользователи осуществляют к ним доступ удаленно с использованием браузера с одним из установленных плагинов JRE, JavaFX, AIR или Silverlight.

Планировщик ОРЦ учитывает ограничения платного ПО при планировании виртуальных машин: лимиты количества экземпляров ПО, од-

новременно работающих на виртуальных машинах в сети, наличие сервера лицензирования, который не является частью виртуального класса, но виртуальная машина, где он установлен, должна запускаться параллельно с классом, наличие аппаратных ключей (в этом случае виртуальная машина, в которой ПО нужен подобный ключ, привязывается к конкретному физическому серверу).

Совместное использование несколькими учреждениями платного ПО и вычислительных ресурсов ОРЦ приводит к значительной экономии денежных средств. Образовательные учреждения платят только за фактическое время использования вычислительных ресурсов и ПО, им не нужно заботиться о покупке и установке пакетов обновлений.

С другой стороны, это приводит к снижению требований к аппаратному обеспечению физических классов образовательного учреждения – для работы с ОРЦ достаточно наличие маломощных компьютеров с установленным браузером и быстрым Интернетом.

В рамках настоящего исследования планируется создать прототип подобного ОРЦ на базе облачной системы Оренбургского государственного университета. Для него будут разработаны алгоритмы создания виртуальных машин со всем необходимым ПО, алгоритмы планирования экземпляров виртуальных машин классов с учетом ограничений лицензирования ПО, а также методы маршрутизации сетевого трафика внутри ЦОД с использованием программно-конфигурируемых сетей [1].

Ранее нами уже была разработана структурная модель центра обработки данных [2], в рамках данной статьи описываются математические модели облачной системы и программного обеспечения ОРЦ. Данные модели станут осно-

вой для разрабатываемых алгоритмических решений.

I. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОБЛАЧНОЙ СИСТЕМЫ

Модель облачной системы ОРЦ может быть формализована в виде набора вида:

$$Cloud = (DataCenter, VMs, Classrooms, Images, vHDDs, Flavors, Users, Programs),$$

где *DataCenter* представляет собой физический ЦОД с коммутаторами OpenFlow (его описание представлено в [2]), *VMs* – множество всех виртуальных машин, *Classrooms* – виртуальные классы, *Images* – дисковые образы виртуальных машин, *vHDDs* – виртуальные жесткие диски для хранения персональных данных пользователей, *Flavors* – типы конфигураций виртуальных машин, *Users* – пользователи ОРЦ, *Programs* – набор платного и бесплатного ПО.

Виртуальная машина VM_i характеризуется следующими параметрами:

$$VM_i = (Flavor_i, Image_i, vHDD_i, State_i(t)).$$

Здесь $Flavor_i$ – тип конфигурации виртуальной машины, $Image_i$ – ее дисковый образ, $vHDD_i$ – персональный жесткий диск пользователя, $State_i(t)$ – состояние в момент t .

Каждый дисковый образ $Image_i$ может быть описан в виде набора:

$$Image_i = (InstPrograms_i, d_i(t)),$$

где $InstPrograms_i \subseteq Programs$ – набор установленного программного обеспечения (полностью или только отдельные модули), $d_i(t)$ – текущий размер дискового образа, выделенный облаком.

Каждый виртуальный класс $Classroom_i$ представляет собой совокупность групп виртуальных машин, связанных между собой виртуальной сетью:

$$Classrom_i = (VMGroups_i, Users_i, Admin_i),$$

где $VMGroups_i = (VMgroup_{i1}, \dots, VMgroup_{iG_i})$ – совокупность групп виртуальных машин, $Users_i \subseteq Users$ – множество пользователей, которым разрешен доступ к виртуальному классу, $Admin_i \in Users$ – его администратор. Наличие нескольких групп виртуальных машин позволяет создать отдельные группы для учеников/студентов и учителей/преподавателей

II. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Любая программа ОРЦ может быть представлена набором ее модулей. Каждый программный модуль определяет совокупность файлов программ, библиотек, настроек, данных, ключей реестра и т.п., которые должны быть установлены на одну виртуальную машину. Если программа состоит из нескольких модулей,

то они могут быть установлены на одной или нескольких виртуальных машинах. В последнем случае взаимодействие между ними осуществляется по сети.

Структура программы с учетом лицензионных ограничений может быть представлена в виде взвешенного неориентированного графа ее программных модулей:

$$Program_i = (Modules_i, Relations_i, MaxMLimit_i, MinRLimit_i, MaxRLimit_i).$$

Его вершинами являются отдельные модули $Modules_i$, а ребрами $Relations_i = \{\{X, Y\}\}$ – неориентированные связи между некоторыми модулями X и Y . Заметим, что каждое неориентированное ребро $\{X, Y\}$ является совокупностью из двух ориентированных дуг (X, Y) и (Y, X) .

Отображение $MaxMLimit_i: Relations_i \rightarrow N$ для каждого программного модуля $X \in Modules_i$ задает максимальное количество его экземпляров $MaxMLimit(X)$, которые могут быть запущены/установлены в ОРЦ в соответствии с его лицензией. Если лицензия не ограничивает максимальное количество экземпляров, то $MaxMLimit(X) = \infty$.

Функции $MinRLimit_i: Relations_i \times Relation_i \rightarrow N$ и $MaxRLimit_i: Relations_i \times Relation_i \rightarrow N$ для каждой ориентированной связи (X, Y) задают соответственно минимальные и максимальные кратности, приписываемые к модулю X . Смысл минимальных кратностей ребра $\{X, Y\}$ заключается в том, что совместная работа модулей X и Y возможна при условии, что в виртуальном классе будет минимум $MinRLimit_i(X, Y)$ экземпляров модуля X и $MinRLimit_i(Y, X)$ – модуля Y . Максимальные кратности ограничивают количество экземпляров соответствующих модулей в соответствии с лицензионными ограничениями.

ЗАКЛЮЧЕНИЕ

Разработанные математические модели позволяют формализовать современные облачные системы, включая OpenStack и OpenNebula, адаптировать их для реализации ОРЦ, а также учитывать лицензионные ограничения установленного программного обеспечения. В будущем данные модели станут основой для разрабатываемых алгоритмических решений.

Исследования выполнены при поддержке РФФИ (проекты №12-07-31089 и №13-07-97046).

1. McKeown, N. Openflow: enabling innovation in campus networks / N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, J. Turner // ACM SIGCOMM Computer Communication Review. – 2008. – Vol. 38. – P. 69–74.
2. Полежаев, П.Н. Математическая модель распределенного вычислительного центра обработки данных с программно-конфигурируемыми сетями его сегментов // Вестник «Оренбургского государственного университета». – 2013. – №5(154). – P. 198–204.