

УДК 004.42

Построение модели единой службы аутентификации, авторизации и учета пользователей информационных ресурсов Университета

*Рыбальченко М. А., студент
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Программное обеспечение ЭВМ и информационные технологии»*

*Научный руководитель: Остриков С.П., к.т.н, доцент
Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Программное обеспечение ЭВМ и информационные технологии»
ostrikov@bmstu.ru*

Введение

В последнее десятилетие имитационное моделирование стало пользоваться особым вниманием профессионалов не только в области науки, но и в различных направлениях экономики и бизнеса. Причина растущей популярности этой технологии связана с особой сложностью социальных, экономических и производственных процессов современного мира, поскольку из всех доступных на сегодняшний день методов именно эксперименты с максимально приближенной к реальности компьютерной моделью позволяют наиболее эффективно оценить жизнеспособность бизнес-систем и процессов[1].

Имитационное моделирование применяется в ситуациях, когда необходимо найти оптимальное решение для успешного функционирования комплексной системы со сложными взаимосвязями и большим количеством используемых технологий[1].

Обоснование необходимости построения модели единой системы аутентификации

На этапе внедрения любой сложной информационной системы возникает вопросы о количестве и характеристиках серверов для узлов системы, необходимой скорости передачи данных в сети, количестве оборудования для построения сети и т.д.

В условиях существующей инфраструктуры вуза, при внедрении единой системы аутентификации, необходимо выполнение определенных требований и условий, накладываемых системой. Возникает необходимость определения некоторых характеристик системы, по которым возможна оценка ее работоспособности в существующей инфраструктуре. Возникает вопрос о возможном использовании

имеющихся серверов вуза, и необходимости закупки дополнительного оборудования. Поскольку разработка и тестирование распределенной системы возможна с использованием одного физического сервера, то при развертывании и внедрении полученного решения возникает множество вариантов:

- Использование одного физического сервера для всех механизмов аутентификации;
- Использование для каждого механизма аутентификации собственного физического сервера;
- Использование нескольких физических серверов для одного механизма аутентификации, например, для CAS, и одного физического сервера для всех остальных механизмов аутентификации.

Дополнительно возникают вопросы о технических характеристиках серверов, числе серверов системы хранения учетных записей, работоспособности системы в моменты выполнения репликации, а также распределения нагрузки между внешними и внутренними пользователями.

Для ответа на эти вопросы экономически целесообразно разработать компьютерную модель, детально описывающая структуру системы, ее топологию, оснащенность оборудованием, техникой, поскольку стоимость закупки необходимого оборудования и реальное развертывание распределенной системы высока. При этом ошибки в несоблюдении требований и условий, накладываемых системой, могут значительно увеличить стоимость решения[1].

Таким образом проведение эксперимента по работоспособности системы с использованием имитационного моделирования позволяет с минимальными затратами, без необходимости развертывания множества серверов, определить различные характеристики как всей системы в целом, так и отдельных ее компонентов, а также оценить работоспособность полученного решения при ожидаемом уровне загруженности.

Единая служба аутентификации, авторизации и учета пользователей Университета

В данной статье рассматривается вопрос построения модели и проведения имитационного моделирования единой службы аутентификации, авторизации и учета пользователей вуза.

Архитектура данной распределенной системы и некоторые технические характеристики были подробно описаны в следующих статьях:

- Рыбальченко М.А., Остриков С.П., «Архитектура единой службы аутентификации, авторизации и учета пользователей информационных ресурсов университета», Электронный журнал «Молодежный научно-технический вестник» №01, январь 2014 (<http://sntbul.bmstu.ru/doc/681183.html>)
- Рыбальченко М.А., Остриков С.П., «Разработка единой системы управления пользователями в корпоративных приложениях Университета», Электронный журнал «Молодежный научно-технический вестник» №04, апрель 2013 (<http://sntbul.bmstu.ru/doc/551855.html>)

Выбор метода моделирования

Математическая модель может быть исследована различными методами – аналитическими или имитационными. В некоторых случаях наличие имитационной модели делает возможным применение математических методов оптимизации[2]. Для использования аналитических методов необходимо математическую модель преобразовать к виду явных аналитических зависимостей между характеристиками и параметрами системы и внешних воздействий. Однако это удастся лишь для сравнительно простых систем и при наличии хорошо разработанной теории исследуемых объектов[2].

При имитационном моделировании динамические процессы системы – оригинала подменяются процессами, имитируемыми в абстрактной модели, но с соблюдением таких же соотношением длительностей и временных последовательностей отдельных операций.

В данном случае для определения работоспособности распределенной системы, учитывая ее сложность, наиболее предпочтителен метод имитационного моделирования. В частности, разработанную систему возможно представить, как систему массового обслуживания (СМО), что позволяет построить дискретно-событийную модель системы[3].

В качестве входного потока заявок мы имеем запросы на проведение аутентификации пользователей. Обслуживающие аппараты – механизмы аутентификации – выполняют обработку заявок за заданное время. При этом имеется очередь заявок на обработку в каждом отдельном обслуживающем аппарате. Также, при достижении определенного времени ожидания заявка теряется (*TTL TCP* пакета).

Ограничения на детализацию модели

При разработке модели, возможно рассмотрение системы с различными уровнями детализации. В данном случае необходимо построить модель с достаточной

достоверностью для оценки работоспособности системы при ожидаемом уровне нагрузки.

Модель должна отображать этапы непосредственно проведения аутентификации, учитывая задержки отправки\получения запросов серверов аутентификации, задержки получения ответа от служб каталогов, распределения потоков пользователей между приложениям и между серверами аутентификации.

При данной детализации модель позволит получить представление о времени выполнения аутентификации, количестве аутентификации в единицу времени, числе активных сессий пользователей, распределении нагрузки между серверами.

Были введены следующие ограничения к детализации модели:

- Модель не отображает процессы передачи данных по сети. Здесь введено ограничение, что передача данных происходит без потерь, на скорости 100 Мбит\с. Скорость передачи данных внешних и внутренних пользователей считается одинаковой.
- Модель не отображает процессы работы служб каталогов: выполнение запросов *add\delete\search\modify*. При получении характеристик моделируемой системы важно только фиксирование времени обработки запроса на аутентификацию: служба каталогов получает запрос, далее фиксируется только время получения ответа.

Входные параметры модели

Важным этапом при построении модели распределенной системы является получение входных параметров ее узлов. Данные параметры были получены с использованием средств нагрузочного тестирования, для серверов аутентификации, и собранной статистики работы для приложений.

Параметры серверов аутентификации

Параметры серверов аутентификации и служб каталогов были получены на компьютерах со следующими характеристиками:

- Процессор – 2 x *Intel Xeon* 2,6 ГГц;
- Объем оперативной памяти – 4 Гб;
- Дисковая подсистема – 2 x 160 Гб;
- Сетевой адаптер – 100 Мбит.

Параметры RADIUS серверов

Для получения характеристик выполнения запросов *RADIUS* сервера можно воспользоваться файлом *radius.log*, в котором ведется статистика обо всех успешных и

неудачных запросах.

Чтобы получить итоговые показатели по всем запросам *RADIUS* сервера был применен следующий скрипт:

```
#!/bin/sh
awk '
BEGIN {
    auth_sum=0; auth_cnt=0; auth_min=0; auth_max=0; start_sum=0; start_cnt=0;
    start_min=0; start_max=0; update_sum=0; update_cnt=0;
    update_min=0; update_max=0; stop_sum=0; stop_cnt=0; stop_min=0; stop_max=0;
};

{
    if($4>'0')
    {
        if($3=="auth:")
        {
            auth_sum+=$4;
            auth_cnt+='1';
            if(auth_min=='0' || auth_min>$4){auth_min=$4};
            if(auth_max<$4){auth_max=$4}
        };
        if($3=="start:")
        {
            start_sum+=$4;
            start_cnt+='1';
            if(start_min=='0' || start_min>$4){start_min=$4};
            if(start_max<$4){start_max=$4}
        };
        if($3=="update:")
        {
            update_sum+=$4;
            update_cnt+='1';
            if(update_min=='0' || update_min>$4){update_min=$4};
            if(update_max<$4){update_max=$4}
        };
        if($3=="stop:")
        {
            stop_sum+=$4;
            stop_cnt+='1';
            if(stop_min=='0' || stop_min>$4){stop_min=$4};
            if(stop_max<$4){stop_max=$4}
        }
    }
};
END{
    if(auth_cnt=='0'){auth_cnt='0.00001'};
    if(start_cnt=='0'){start_cnt='0.00001'};
    if(update_cnt=='0'){update_cnt='0.00001'};

```

```

        if(stop_cnt=='0'){stop_cnt='0.00001'};
        print "auth (avg / min / max [cnt]):\t" auth_sum/auth_cnt "\t/ " auth_min "\t/ "
auth_max "\t[" int(auth_cnt) "];
        print "start (avg / min / max [cnt]):\t" start_sum/start_cnt "\t/ " start_min "\t/ "
start_max "\t[" int(start_cnt) "];
        print "update (avg / min / max [cnt]):\t" update_sum/update_cnt "\t/ " update_min
"\t/ " update_max "\t[" int(update_cnt) "];
        print "stop (avg / min / max [cnt]):\t" stop_sum/stop_cnt "\t/ " stop_min "\t/ "
stop_max "\t[" int(stop_cnt) "];
    }'

```

В результате были получены следующие характеристики *RADIUS* сервера:

- Среднее количество полученных/переданных пар *RADIUS Request/Response* пакетов в единицу времени: 25 пакетов\сек.
- Наименьшее время обработки запроса пользователя: 77 мс.
- Наибольшее время обработки запроса пользователя: 471 мс.
- Среднее время обработки запроса пользователя: 207 мс.
- Закон распределения для времени обработки запроса пользователя: равномерное распределение.

Параметры CAS серверов

Для получения характеристик выполнения аутентификации с использованием *Jasig* CAS сервера можно использовать библиотеку *Zorka CAS Monitoring* (<http://zorka.io/install/cas.html>).

Zorka CAS Monitoring позволяет через веб-интерфейс отображать подробную статистику об выполнении запросов, выданных *ticket* и др. Данный инструмент очень удобен для мониторинга работы системы, однако он не предоставляет функционал для проведения нагрузочного тестирования.

В условиях, когда системы еще не была введена в эксплуатацию, имеет смысл провести нагрузочное тестирование для определения характеристик CAS сервера. Для выполнения нагрузочного тестирования использовался инструмент *Apache JMeter* (<http://jmeter.apache.org/>).

Воспользовавшись инструментом *Apache JMeter* была собрана статистика работы CAS сервера:

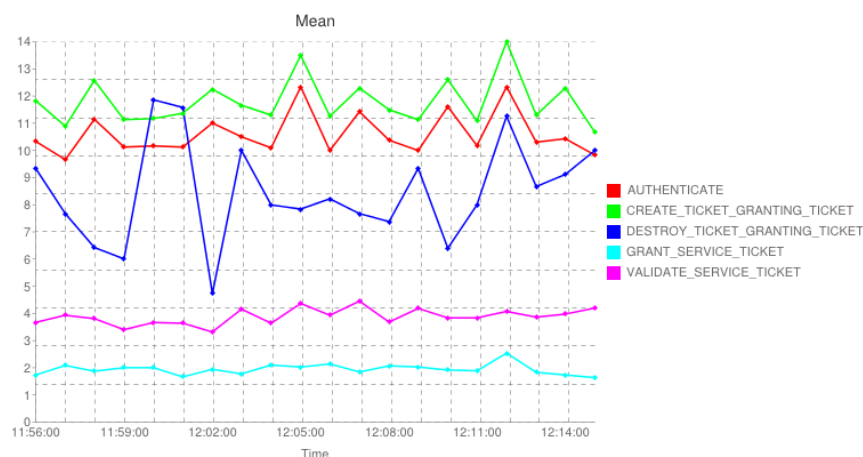


Рис. 1. Статистика выполнения аутентификации CAS сервером

Проанализировав показатели, полученные с использование *Apache JMeter*, а также настройки CAS сервера, были выделены следующие ключевые характеристики:

- Максимальная длительность SSO сессии пользователя 60 минут.
- Повторный ввод пароля после трех неудачных попыток подключения: 10 минут.
- Наименьшее время обработки запроса пользователя: 94 мс
- Наибольшее время обработки запроса пользователя: 1123 мс
- Среднее время обработки запроса пользователя: 365 мс
- Закон распределения для времени обработки запроса пользователя: нормальное распределение

Параметры приложений

Для приложений важнейшим показателем является частота обращений пользователей для проведения аутентификации. Необходимо определить с какой интенсивностью и какое число пользователей производят аутентификацию в соответствующих приложениях[4].

Данные параметры можно получить из статистики использования каждого отдельного приложения, если оно уже эксплуатируется в рамках вуза, или же делать предположение о количестве возможных пользователей внедряемого приложения.

Отношение количества внешних и внутренних пользователей.

Всего в МГТУ им. Н.Э. Баумана работают около 9000 сотрудников, а также обучаются 15000 студентов и аспирантов[5].

Предполагается отношение внешних пользователей ко внутренним примерно 20/80. Данное соотношение мотивировано тем, что для сотрудников использование информационных ресурсов Университета в большей степени актуально в рабочее время, находясь в стенах вуза. Студенты также будут создавать большое число подключений, например, к *WiFi*, находясь в Университете. Возможность обращения, например, к Электронному Университету, за пределами вуза заинтересует студентов, однако их число будет значительно меньше, чем общее число студентов, подключающихся к информационным ресурсам Университета, находясь в его стенах.

Также необходимо учитывать, что отношение внутренних и внешних пользователей будет менять в течении дня. Так во второй половине дня число внешних пользователей будет увеличиваться (студенты и сотрудники обращаются к ресурсам Университета из дома).

Стоит отметить, что в течении дня можно выделить два пика активности пользователей:

- В начале рабочего дня – когда большинство сотрудников и студентов приходит в Университет, и начинают обращаться к информационным ресурсам;
- В середине дня – в послеобеденное время, когда сотрудники и студенты прерывают сессии подключения, и восстанавливают их закончив обед.

Исследуемые характеристики системы

По итогам проведенного эксперимента с использованием разрабатываемой модели необходимо ответить на вопрос: каков требуемый набор серверов аутентификации и служб каталогов позволит поддерживать работоспособность системы при ожидаемом уровне нагрузки?

Критериями работоспособности системы следует оценивать по следующим показателям:

- Выполнение не менее 500 запросов на аутентификацию в секунду;
- Максимальное время выполнения аутентификации не более 3 секунд;
- Предельная нагрузка серверов аутентификации не должна превышать 70%;
- Загруженность служб каталогов не должна превышать 70%;
- Допустимый процент отказов (из-за истечения время ожидания, переполнения доступных ресурсов) запросов на аутентификацию должен быть в пределах 3-5%.

Инструмент для создания модели и проведения эксперимента

Благодаря развитию графических средств разработки имитационное моделирование стало более доступно для массового применения. Разработчики стараются предоставить максимальный выбор блоков для визуального построения модели. Одним из таких графических средств разработки является инструмент *AnyLogic*[6].

Уникальность, гибкость и мощность языка моделирования, предоставляемого *AnyLogic*, позволяет учесть любой аспект моделируемой системы с любым уровнем детализации[6]. Графический интерфейс программы, инструменты и библиотеки позволяют быстро создавать модели для широко спектра задач от моделирования производства, логистики, бизнес-процессов до стратегических моделей развития компании и рынков.

Также для описания моделей в среде *AnyLogic* могут применяться низкоуровневые конструкции моделирования (переменные, уравнения, параметры, события и т.п.), различные анимационные фигуры (линии, квадраты, овалы и т.п.), элементы анализа (базы данных, гистограммы, графики), стандартные картинки и шаблоны экспериментов.

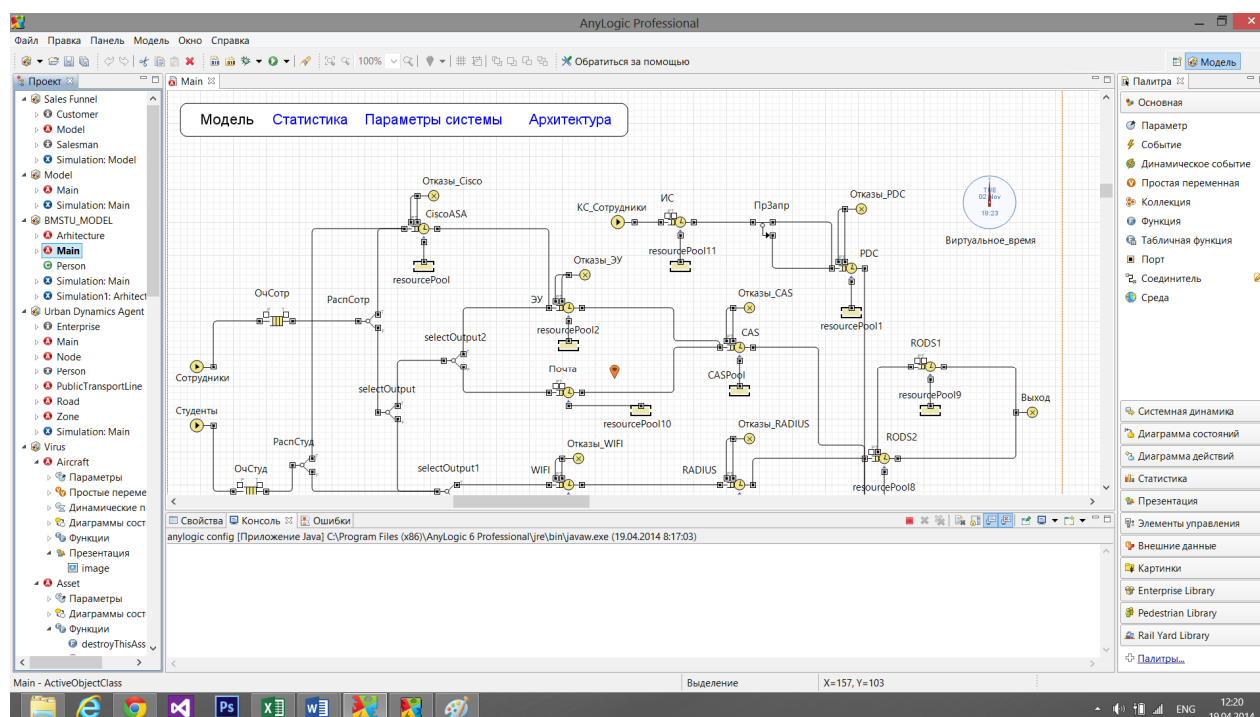


Рис. 2. Интерфейс визуального редактора AnyLogic

Особенностью данного инструмента моделирования является возможность создания интерактивной анимации. Запуск анимированной модели дает наглядное представление о состоянии и поведении сложной системы любого вида.

Учитывая перечисленные преимущества, а также удобство создания моделей и их анимированных представлений, для разработки модели и проведения эксперимента использовался инструмент *AnyLogic*.

Модель единой системы аутентификации пользователей

На основе архитектуры распределенной системы, учитывая все описанные ранее ограничения, входные характеристики серверов аутентификации и служб каталогов, а также распределение внутренних и внешних пользователей в течении дня, была построена следующая модель единой системы аутентификации пользователей:

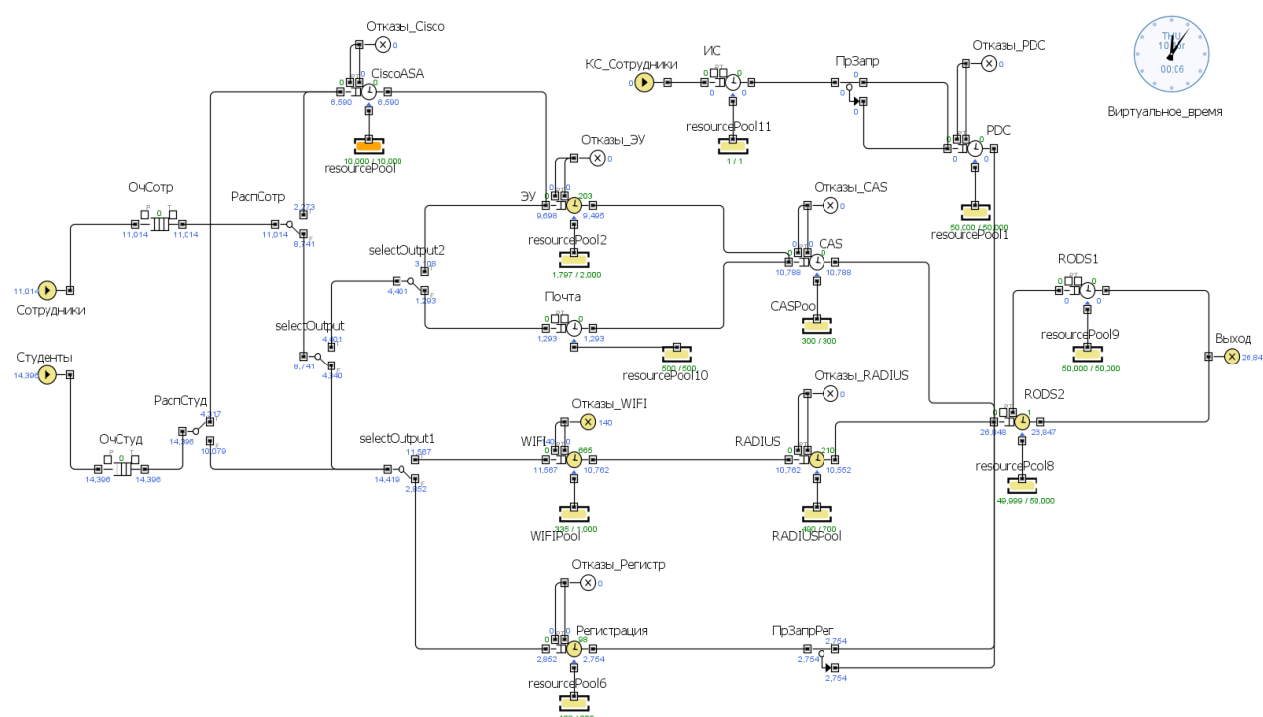


Рис. 3. Разработанная имитационная модель Единой системы аутентификации, авторизации и учета пользователей Университета

На входе имеются два источника: Сотрудники и Студенты. Данные источники генерируют заявки, которые фактически являются пользователями, которые обращаются к различным приложениям. Используя блок распределения заявки-пользователи поступают либо в обработчики-приложения (для внутренних пользователей), либо в обработчик имитирующий поведение *Cisco ASA* (внешние пользователи). Внутренние пользователи, проходя обработку в блоке *Cisco ASA* также поступают в блоки-приложения.

Источники генерируют заявки-пользователей используя описанные выше данные о числе пользователей и их распределении в течении дня. Блоки-приложения представляют собой очередь с задержкой, потоки пользователей, поступающие в эти блоки, преобразуются в запросы на аутентификацию. Запросы на аутентификацию поступают в определенные сервера аутентификации, которые также являются очередями с задержкой. Из блоков-серверов аутентификации запросы передаются к службе каталогов, после чего проходят обратный путь через сервера аутентификации возвращаясь к блокам-приложениям.

Таким образом, промоделировав процесс проведения аутентификации, который включает в себя передачу заявки от приложения серверу аутентификации и службе каталогов, можно фиксировать все требуемые параметры системы.

Для наглядного представления работы единой системы аутентификации пользователей на основе модели была разработана анимированная схема проведения аутентификации пользователями:

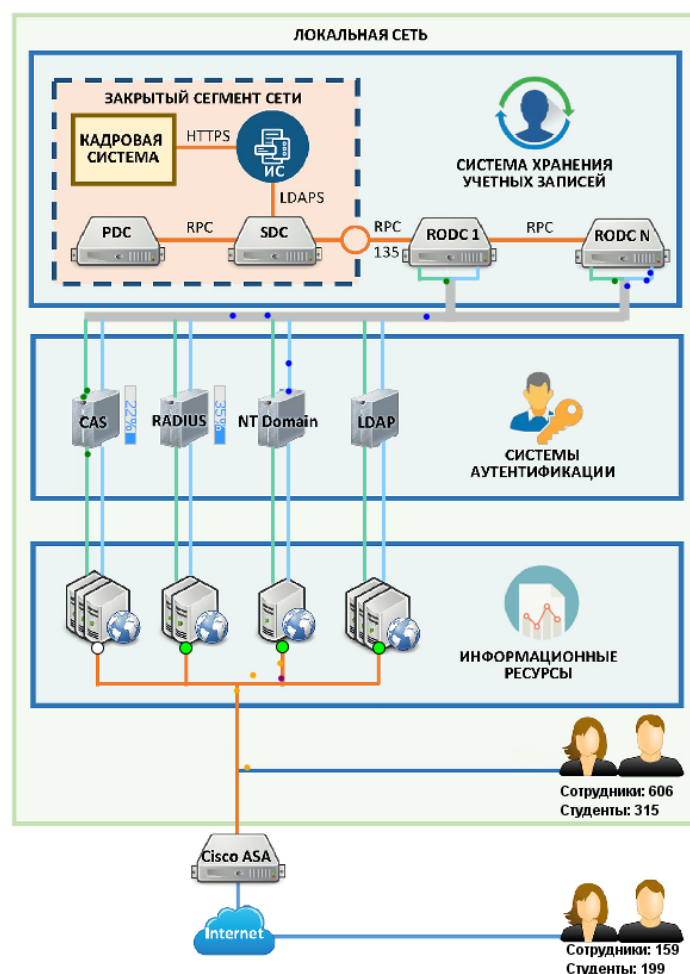


Рис. 4. Анимированная схема выполнения аутентификации в системе

Данная схема наглядно отображает потоки пользователей и миграцию заявки между приложением, сервером аутентификации и службой каталогов.

Возможности по проведению экспериментов, с использованием разработанной модели

Разработанную модель можно использовать для проведения множества различных экспериментов, меняя ее характеристики или конфигурацию. Для ответа на поставленный вопрос, при выполнении эксперимента, можно изменять интенсивность числа обращений в сервера аутентификации, перераспределять потоки пользователей между приложениями, изменять число серверов аутентификации и служб каталогов и т.п.

При выполнении эксперимента возможно получение широкого круга характеристик исследуемой системы, а также получение итоговой статистики работы системы. Некоторые из этих характеристик представлены на рисунке:



Рис. 5. Статистика работы узлов системы во время проведения эксперимента

Используя полученную модель будет произведено дальнейшее исследование разработанной системы:

- Необходимо показать адекватность полученной модели и разработанной системы;
- Требуется получить необходимую конфигурацию и характеристики серверов аутентификации и служб каталогов;
- Будет произведена имитация максимальной ожидаемой нагрузки узлов системы, с целью проверки ее работоспособности.

Выводы

В рамках данной статьи были описаны все этапы разработки модели Единой системы аутентификации, авторизации и учета пользователей Университета. Разрабатываемая система была представлена, как система массового обслуживания, что позволило построить ее дискретно-событийную модель.

Протестировав сервера аутентификации с использованием инструмента *Apache JMeter*, а также проанализировав файлы статистики использования механизмов аутентификации, были получены характеристики выполнения аутентификации *RADIUS*, *CAS* и *LDAP* при ожидаемом уровне нагрузки на предполагаемой конфигурации серверов.

Дальнейшее проведение эксперимента, для определения работоспособности системы с использованием имитационного моделирования, позволит с минимальными затратами, без необходимости развертывания множества серверов, определить различные характеристики как всей системы в целом, так и отдельных ее компонентов, а также оценить работу полученного решения при ожидаемом уровне загруженности.

Использование современного графического средства моделирования *AnyLogic* позволило построить модель с заданным уровнем детализации, получить требуемые характеристики системы, а визуализация модели дала наглядное представление о состоянии и поведении единой системы аутентификации.

Список литературы

1. Борщев А.В., Применение имитационного моделирования в России - состояние на 2007г. // 3-я Всероссийская научно-практическая конференция по имитационному моделированию ИММОД 2007 (Санкт-Петербург, 17-19 октября 2007 г.): тез. докл. Санкт-Петербург, 2007. С. 5-7.

2. Попков Т. В., Многоподходное моделирование, практика использования // 4-я Всероссийская научно-практическая конференция по имитационному моделированию ИММОД 2009. (Санкт-Петербург, 21-23 октября 2009 г.): тез. докл. Санкт-Петербург, 2009. С. 3-4.
3. Строгалев В. П., Толкачева И. О. Имитационное моделирование: МГТУ им. Баумана, 2008, 737 с.
4. Рыбальченко М.А. Архитектура единой службы аутентификации, авторизации и учета пользователей информационных ресурсов университета // Молодежный научно-технический вестник. Электрон. Журн. 2014. № 1. Режим доступа: <http://sntbul.bmstu.ru/doc/681183.html> (дата обращения 20.11.2014).
5. Рыбальченко М.А. Разработка единой системы управления пользователями в корпоративных приложениях Университета // Молодежный научно-технический вестник. МГТУ им. Н.Э. Баумана. Электрон. Журн. 2013. № 4. Режим доступа: <http://sntbul.bmstu.ru/doc/551855.html> (дата обращения 20.11.2014).
6. Мезенцев К.Н., Моделирование систем в среде AnyLogic 6, Режим доступа: [http://www.anylogic.ru/upload/Books_ru/Mezenczev_Systems_Simulation_In_AnyLogic_6_\(Part_1\).pdf](http://www.anylogic.ru/upload/Books_ru/Mezenczev_Systems_Simulation_In_AnyLogic_6_(Part_1).pdf) (дата обращения 12.05.14).