Организация безопасных вычислений на распределенных мобильных устройствах

Н.А. Беляев

Новосибирский государственный технический университет, Новосибирск

Представлен проект и прототип реализации системы распределенных вычислений на мобильных устройствах. Обеспечивается предотвращение зловредных действий со стороны вычислительных программ на мобильных устройствах пользователей.

Введение

Все больше растут потребности человечества в вычислениях для решения множества фундаментальных и прикладных задач. Ряд задач успешно решается в волонтерских сетях распределенных вычислений, таких как World Community Grid, работающих на основе программного обеспечения BOINC[1]. Пользователи персональных компьютеров устанавливают у себя клиентский компонент системы BOINC, который обращается на сервер World Community Grid, скачивает вычислительное задание, выполняет его и отдает серверу результат. Сеть широко используется для решения задач анализа медицинских изображений, структуры молекул белков, подбора лекарств, проектирования новых материалов.

Быстрый рост вычислительной производительности мобильных устройств и количества таких устройств в мире делает целесообразным вовлечение ресурсов мобильных устройств в волонтерские вычислительные сети. В 2013 г. в рамках BOINC было разработано клиентское программное обеспечение для платформы Android [1].

Участники вычислений в таких проектах доверяют организаторам вычислений, и вопрос о проверке безопасности вычислений не ставится. Подобные сети могли бы использоваться для решения задач, поставляемых произвольными заказчиками, и участники расчетов могли бы получать некоторое вознаграждение за предоставление ресурсов. При этом возникает необходимость

[1]BOINC: http://boinc.berkeley.edu

обеспечения безопасности вычислительных устройств и их пользователей.

В настоящей работе предлагается проект системы безопасных отказоустойчивых вычислений на устройствах, работающих на платформе Android, и представлен прототип системы. При организации такой системы распределенных вычислений требуется решение многих проблем, основными из которых являются ограниченность ресурсов мобильных устройств, необходимость обеспечения безопасности вычислений для предотвращения попадания на устройства вредоносных программ, а также обеспечение равномерного планирования вычислительной нагрузки по узлам сети. Также важной проблемой является обеспечение отказоустойчивости в вычислительной сети.

Архитектура системы

Модель организации вычислений. Система состоит из сервера, осуществляющего планирование задач, и клиентских устройств, выполняющих вычисления. На клиентские устройства устанавливается программное обеспечение, которое позволяет пользователю устройства выбрать задачи, в решении которых он хочет участвовать, загружает исполняемые файлы программ, осуществляющих вычисления, и загружает порции данных для обработки этими программами. По мере готовности результаты вычислений отправляются на сервер.

Для выполнения расчетов конкретной прикладной задачи разработчик программы загружает исполняемый файл и исходные данные, разбитые на части, на сервер распределенных вычислений, который производит планирование нагрузки, отправку исходных данных на клиентские устройства, а также сбор результатов.

Принцип обеспечения безопасности. Для обеспечения безопасности предлагается следующий подход. Исполняемый файл, попадая на сервер, проходит ряд модификаций. Сначала производится поиск в файле специальной секции, описывающей импортируемые функции. Она предназначена для того, чтобы у программ была возможность коммуникаций с операционной системой, и представляет собой по сути таблицу, в которой каждому символьному имени функции или переменной соответствует некоторое относительное смещение, из которого в процессе загрузки операционная система вычисляет виртуальный адрес данной функции или переменной. Собственно модификации

исполняемого файла, производимые сервером распределенных вычислений, представляют собой удаление информации о всех импортируемых данных исполняемым файлом функций операционной системы, что предотвращает попытку их прямого вызова и, как следствие, попытку выполнить на клиентском устройстве какие-либо вредоносные действия. Важно отметить, что данный метод не гарантирует полную безопасность клиентских устройств, он основан на том факте, что в момент запуска исполняемого файла задачи на клиентском устройстве неизвестны фактические адреса API операционной системы. Таким образом, при наличии уязвимостей в операционной системе существует возможность обхода данного механизма. Также следует отметить тот факт, что модификация исполняемого файла сама по себе не обеспечивает безопасность вычислений. Существует необходимость обеспечения правильного исполнения данных файлов на клиентских устройствах. В настоящем проекте оно обеспечивается благодаря своеобразной «песочнице». Механизм исполнения заданий проиллюстрирован на рис.1. Программа, полученная с сервера распределенных вычислений, не имеет возможности напрямую обращаться к операционной системе. Вместо этого она обращается к API песочницы, которая, в свою очередь, проведя необходимые проверки входных параметров, обращается к соответствующим API операционной системы.

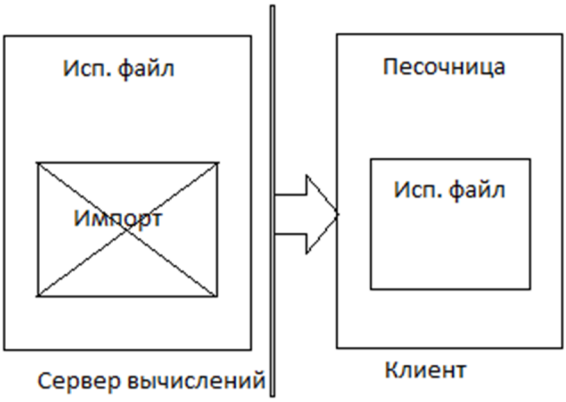


Рис. 1

Механизм планирования

Помимо проблемы обеспечения безопасности, существует необходимость планирования вычислительной нагрузки, распределяемой по узлам. Стоит отметить, что в таких крупных проектах, как BOINC, порции данных, отправляемые для обработки устройствами, имеют одинаковый размер и распределяются по вычислительным узлам вне зависимости от производительности этих ресурсов. Предлагается механизм двухстадийного планирования вычислительной нагрузки. Идея его заключается в том, что изначально известен вычислительный вес каждого фрагмента исходных данных. На первой стадии планирования все множество фрагментов разбивается на подмножества таким образом, что в каждом из подмножеств представлены фрагменты данных, пригодные к вычислению за разумное время на соответствующем наборе устройств, имеющих примерно одинаковую производительность. Так, например, при подобном делении можно учесть различные классы устройств – от высокопроизводительных до бюджетных. Вторая же стадия планирования осуществляется уже непосредственно в момент отправки задания на клиентское устройство и представляет собой, по сути, выбор первого фрагмента входных данных из подмножества, соответствующего данному устройству, который еще не был обработан. Такой подход обеспечивает малые затраты на планирование и в то же время исключает возможность попадания фрагмента, имеющего большой вес, на устройство с низкой производительностью.

Заключение

Разработан проект и реализован прототип системы безопасных распределенных вычислений на мобильных устройствах, работающих на платформе Android. Постановку вычислительных задач осуществляют заказчики, предоставляя исполняемый файл и наборы данных для обработки. Пользователи мобильных устройств, устанавливая специальное программное обеспечение, становятся участниками распределенной вычислительной сети и принимают расчетные задания на свои мобильные устройства. Система предотвращает возможность вредоносных действий со стороны программ заказчиков на мобильных устройствах участниках за счет модификации программ и исполнения их в специальной

виртуальной среде – «песочнице». Обеспечивается планирование распределения вычислений в соответствии с производительностью вычислительных устройств.

Дальнейшие планы включают обеспечение возможности параллельных расчетов на мобильных устройствах и разработку системы отказоустойчивости таких вычислений.

Литература

1. *Fritzsch J.* BOINC on Android: state & outlook // The 9th BOINC Workshop. Grenoble, Inria, 25-27 September 2013 http://boinc.berkeley.edu/trac/raw-attachment/wiki/WorkShop13/boa-state_and_outlook.pdf