УДК 681.3:003.26

В.К. Задирака, А.М. Кудин,
П.В. Селюх, И.В. Швидченко

ОБЛАЧНЫЕ ТЕХНОЛОГИИ: НОВЫЕ ВОЗМОЖНОСТИ ВЫЧИСЛИТЕЛЬНОГО КРИПТОАНАЛИЗА

Введение

Ежегодные отчеты Cisco Global Cloud Index (2014–2019) [1] отмечают определенную тенденцию дальнейшего увеличения доли трафика, обрабатываемого центрами обработки данных (ЦОД), построенными на базе облачных технологий над трафиком, обрабатываемым ЦОД, построенными на базе традиционных технологий. По прогнозу на 2017 г. доля «облачных» систем в общем трафике превысит 2/3. Это повышает актуальность проблем оценки эффективности применения облачных технологий для решения традиционно сложных и важных вычислительных задач, например задачи криптоанализа. В статье обсуждаются перспективы применения облачных технологий для широкого класса вычислительных задач в области криптоанализа.

Характеристика облачных вычислений как нового способа организации вычислений и новой модели вычислений

Согласно определению, предложенному Национальным институтом стандартов и технологий США [2], облачные вычисления (cloud computing) — модель (концепция) реализации возможности повсеместного и удобного сетевого доступа по требованию к пулу разделяемых конфигурируемых вычислительных ресурсов (например, сетям, серверам, средствам хранения, приложениям и сервисам), которые могут оперативно предоставляться и освобождаться при минимальном усилии управления или взаимодействии с провайдером (поставщиком). Эта модель облака описана пятью основными характеристиками, тремя сервисными моделями и четырьмя моделями развертывания.

Основные характеристики облачных вычислений.

— Самообслуживание по требованию. Потребитель по мере необходимости автоматически, без взаимодействия с каждым поставщиком услуг, может самостоятельно определять и изменять вычислительные мощности, такие как серверное время, объем хранилища данных.

— Широкий (универсальный) сетевой доступ. Вычислительные возможности доступны на большие расстояния по сети через стандартные механизмы, что способствует широкому использованию разнородных (тонких или толстых) платформ клиента (терминальных устройств).

— Объединение ресурсов. Конфигурируемые вычислительные ресурсы поставщика объединены в единый пул для совместного использования распределенных ресурсов большим количеством потребителей.

 ISSN 0572-2691

— Мгновенная эластичность ресурсов (мгновенная масштабируемость). Облачные услуги могут быстро предоставляться, расширяться, сжиматься и освобождаться исходя из потребностей потребителя.

— Измеряемый сервис (учет потребляемого сервиса и возможность оплаты реально использованных услуг). Облачные системы автоматически управляют и оптимизируют использование ресурсов за счет измерений на некотором уровне абстракции, соответствующей типу сервиса.

Если модель (концепция) предоставления распределенных и разделяемых конфигурируемых вычислительных ресурсов соответствует вышеизложенным характеристикам, то это облачные вычисления.

Сервисные модели облачных вычислений.

— Software as a Service (SaaS) — программное обеспечение как услуга. В этой модели предоставления облачных вычислений потребитель использует приложения поставщика, запущенные в облачной инфраструктуре, которые доступны клиенту через интерфейс (web-браузер) или интерфейс программы. Потребители не могут управлять и контролировать лежащую в основе облака инфраструктуру, включая сеть, серверы, операционные системы, хранилища данных или даже изменять параметры настройки конкретного приложения.

— Platform as a Service (PaaS) — платформа как услуга. Модель предоставления облачных вычислений, при которой потребитель получает доступ к использованию программной платформы: операционных систем, СУБД, прикладного ПО, средств разработки и тестирования ПО. Фактически потребитель получает в аренду компьютерную платформу с установленной операционной системой и специализированными средствами для разработки, размещения и управления веб-приложениями. Потребитель не управляет основной инфраструктурой облака, включая сеть, серверы, операционные системы или хранилища данных, но управляет развернутыми приложениями и, возможно, параметрами настройки конфигурации среды окружения.

— Infrastructure as a Service (IaaS) — инфраструктура как услуга. Модель предоставления облачных вычислений, при которой потребитель получает возможность управлять средствами обработки и хранения, а также и другими фундаментальными вычислительными ресурсами (виртуальными серверами и сетевой инфраструктурой), на которых он может самостоятельно устанавливать операционные системы и прикладные программы под собственные цели. По сути, потребитель арендует абстрактные вычислительные мощности (серверное время, дисковое пространство и пропускную способность сетевых каналов) или использует услуги аутсорсинга ИТ-инфраструктуры. Он не управляет основной инфраструктурой облака, но управляет операционными системами, хранилищем и развернутыми им приложениями.

Модели развертывания облачных вычислений.

— Частное облако (Private cloud) — инфраструктура, предназначенная для использования облачных вычислений в масштабе одной организации.

— Облако сообщества (Community cloud) — облачная инфраструктура, предназначенная для исключительного использования облачных вычислений определенным сообществом потребителей от организаций, которые решают общие проблемы.

— Публичное облако (Public cloud) — инфраструктура, предназначенная для свободного использования облачных вычислений широкой публикой;

— Гибридное облако (Hybrid cloud) — комбинация различных облачных инфраструктур (частных, публичных или сообществ), остающихся уникальными объектами, но связанных между собой стандартизованными или частными технологиями, которые обеспечивают возможность обмена данными и приложениями.

Модель облачных вычислений состоит из внешней (front end) и внутренней (back end) частей. Эти два элемента соединены по сети, в большинстве случаев через Интернет. Посредством внешней части потребитель взаимодействует с системой; внутренняя часть — собственно само облако. Внешняя часть состоит из клиентского компьютера или сети компьютеров предприятия и приложений, используемых для доступа к облаку. Внутренняя часть предоставляет приложения, компьютеры, серверы и хранилища данных, создающие облако сервисов [1–6].

Облако предоставляет следующие уровни.

— Уровень инфраструктуры — это основа облака. Он состоит из физических активов — серверов, сетевых устройств, дисков и т.д. Существуют поставщики инфраструктуры как IaaS, например IBM® Cloud. При взаимодействии с IaaS потребитель в действительности не управляет базовой инфраструктурой, однако управляет операционными системами, хранилищами данных, развертываемыми приложениями и, до определенной степени, выбранными сетевыми компонентами. Примером организаций, которые могут получить выгоды от IaaS, являются сервисы печати по требованию (Print On Demand — POD). Модель POD основана на продаже товаров, дизайн которых задается в соответствии с требованиями клиента. POD позволяет физическим лицам открывать магазины и продавать дизайны товаров. Владельцы магазинов могут загрузить столько дизайнов, сколько в состоянии создать. Многие загружают тысячи дизайнов. Благодаря возможностям облачной системы хранения POD может предоставлять неограниченный объем дискового пространства.

— Промежуточным уровнем является платформа — инфраструктура приложений. Платформа как PaaS предоставляет доступ к операционным системам и соответствующим сервисам, способствует развертыванию приложений в облаке с помощью языков программирования и инструментальных средств, поддерживаемых поставщиком. Пользователю не нужно управлять используемой инфраструктурой или контролировать ее, но у него есть возможность управлять развернутыми приложениями и, до определенной степени, конфигурациями среды хостинга приложений. Существуют поставщики PaaS, например Elastic Compute Cloud (EC2) от Amazon. Идеальный потребитель PaaS — это небольшая частная фирма по созданию программного обеспечения. Имея в своем распоряжении такую платформу, можно создавать продукты мирового класса без накладных расходов, свойственных разработке на собственных ресурсах.

— Верхний уровень — уровень приложений, который обычно изображают в виде облака. Приложения, выполняющиеся в нем, предоставляются пользователям по требованию. Существуют поставщики программного обеспечения как сервиса (SaaS), например, Google Pack. Google Pack содержит доступные через Интернет приложения — Calendar, Gmail, Google Talk, Docs и многие другие.

Исходя из определения, можно выделить основные особенности облачных вычислительных технологий, определяющие новые постановки задач криптоанализа.

— Для облачных вычислительных систем (ОВС) характерно наличие асимметричных вычислений — мощное «облако» с практически неограниченными вычислительными возможностями и множество терминальных устройств (в том числе мобильных), которые могут служить элементами управления процессами криптоанализа. Реализация процесса управления задачей криптоанализа должна осуществляться по агентной парадигме с использованием XML-шаблонов.

— Новые возможности использования облачных вычислений для криптоанализа ограничиваются проблемой раскрытия целей и методик криптоанализа перед провайдером услуг. Поэтому возникают новые постановки задач, связанные с определением перечня вычислительных задач, существенных для криптоанализа,

изучение которых в совокупности не позволяло бы восстанавливать задачу криптоанализа целиком. Фактически речь идет о частном случае задачи разделения секрета.

Несмотря на то, что облачные технологии — это просто реализация распределенных вычислений, впервые практический аналог полной распределенности или конструктора из которого можно построить любые модели вычислений, кроме тех, которые используют новые физические принципы (например, квантовые).

Оценки сложности атак тотального и целенаправленного перебора ключей симметричных криптосистем

Trade-off атаки — атаки компромисса между временем, которое тратится на поиск ключа и объемом памяти для хранения предвычислений. Исторически первой атакой компромисса времени и памяти была атака Хеллмана [7]. В оригинальной работе рассматривалось построение атаки на симметричный алгоритм шифрования DES на основе открытого текста. Кратко рассмотрим суть атаки Хеллмана на примере ее применения к алгоритмам хеширования [8].

Пусть имеем функцию хеширования $F(x)$, которая принимает значение на множестве **Φ**, и множество возможных сообщений $X = \{x_i\}$, которые могут подаваться на вход алгоритма хеширования. Наша задача — построение таблицы, содержащей пары значений сообщение/хеш-код $\{< x_i, F(x_i)\}$. О сообщении известно только то, что это битовая последовательность фиксированной длины n. При тривиальном построении полной таблицы значений $\{< x_i, F(x_i)\}$ расходуется 2^n элементов памяти $n+c$ бит каждый (здесь c — длина хэш-кода в битах). Для уменьшения размера таблицы Хеллман предложил выбрать m стартовых точек $x_0,..., x_{m-1}$ из пространства возможных значений сообщений X. Над каждой точкой нужно выполнить цепочку преобразований:

$$x_i^0 \to R(F(x_i^0)) \to x_i^1 \to R(F(x_i^1)) \to ... \to x_i^{t-1} \to F(x_i^{t-1}),$$

где R — функция «редукции» (произвольное сюръективное отображение $R : F \to X$, например хеш-функция). Длина цепочки t. В таблицу вносится лишь значение $\{< x_i^0, F(x_i^{t-1})\}$. Для построения цепочки прообраза некоторой хеш-последовательности S_0 выполняются такие преобразования:

$$S_0 \to F(R(S_0)) \to S_1 \to F(R(S_1)) \to ... \to S_{t-1}.$$

Если окажется, что $\exists i : S_i = F(X_j^{t-1})$, то выполнив цепочку преобразований, получим $x_j^k : S_0 F(x_j^k)$. При этом значение m и t должны удовлетворять правилу $mt^2 = |X|$. Тогда вероятность удачного поиска по таблице равна [7]

$$P \geq \frac{1}{N} \sum_{i=1}^{m} \sum_{j=1}^{t} \left(1 - \frac{it}{N}\right)^j,$$

где $N = |X|$, а покрытое таким образом количество входных сообщений будет:

$$N' \approx \frac{mt}{N} = \frac{1}{t}.$$

Для атаки используется t таблиц с разными функциями редукции, поскольку в случае обычных таблиц поиска гарантировано можно найти прообраз, а в дан-

ной атаке лишь с определенной вероятностью. Но количество памяти, которое затрачивается на построение таблицы, будет всего лишь $M = mt$, что значительно меньше, чем полные таблицы прообразов, а время перебора $T = t^2$ значительно меньше времени полного перебора.

Недостаток такого построения таблицы заключается в возможности возникновения слияния цепочек, т.е. если некоторые две цепочки содержат одинаковый элемент, то следующие элементы цепочек тоже будут совпадать. Это, в свою очередь, приводит к уменьшению вероятности успеха в нахождении прообраза.

Для того чтобы бороться с проблемой слияния цепочек, предложены так называемые радужные таблицы [9]. Отличие радужных таблиц от классических в том, что вместо одинаковой функции редукции для всех элементов цепочки используется не одна, а набор функций редукции, т.е. цепочка будет иметь вид

$$x_i^0 \to R_0(F(x_i^0)) \to x_i^1 \to R_1(F(x_i^1)) \to ... \to x_i^{t-1} \to F(x_i^{t-1}).$$

В таком случае слияние цепочек возможно только тогда, когда одинаковые элементы находятся на одной позиции. Тогда слитые цепочки легко распознаются, поскольку имеют одинаковые конечные точки. В другом случае цепочки могут пересекаться, но не сливаться.

Если сравнивать с таблицами для атак Хеллмана, то вероятность успеха поиска в t таблицах с длиной цепочки t и количеством стартовых точек m будет приблизительно эквивалентна вероятности поиска с помощью радужных таблиц с длиной цепочки t и количеством стартовых точек mt.

Для поиска прообраза данной хеш-последовательности S_0 по таблице выполняется такая цепочка преобразований:

$$S_0 \to F(R_0(S_0)) \to S_1 \to F(R_1(S_1)) \to F(R_2(S_2)) \to ... \to S_{t-1}.$$

Если значение S_{t-1} не является конечной точкой, выполняется такая цепочка:

$$S_0 \to F(R_1(S_0)) \to S_1 \to F(R_2(S_1)) \to ... \to S_{t-1}.$$

Вычисления проводят, пока значение S_{t-1} не станет равным какой-то конечной точке. Очевидно, что нужно выполнить не больше $t(t-1)/2$ операций для построения цепочек и еще $t\log(mt)$ операций с памятью для поиска конечных точек.

Основные преимущества радужных таблиц перед таблицами в trade-off атаке Хеллмана:

— вероятность слияния двух цепочек при их пересечнии $p = 1/t$, в случае слияния такие цепочки легко распознать по идентичности их конечных точек;

— радужные цепочки не образуют петель;

— количество операций для построения цепочек вдвое меньше, чем в классической атаке;

— количество обращений к памяти приблизительно в t раз меньше, это весьма существенное преимущество, поскольку обращение к памяти, как правило, — очень медленная операция, а вычисление цепочек происходит значительно быстрее.

Авторы провели расчеты стоимости trade-off атак с использованием облачных технологий, которые показывают, что стоимость криптоанализа сопоставима со стоимостью криптоанализа с применением кластерных систем. Это объясняется хорошим распараллеливанием рассмотренных выше алгоритмов направленного перебора [10] при существенном времени вычисления каждой цепочки.

Оценки сложности атак на асимметричные криптосистемы

Основным из наиболее распространенных и, к сожалению, подверженным атакам на реализацию, является механизм Диффи–Хеллмана формирования ключей, которые используются для шифрования канала связи. Самая распространенная атака — понижение стойкости параметров алгоритма, которые позволяют успешно проводить атаки человек-по-середине. Стойкость упомянутого алгоритма Диффи–Хеллмана основана на задаче дискретного логарифмирования, сложной теоретико-числовой задаче с субэкспоненциальной сложностью.

Новейший рекорд дискретного логарифма — 596 бит [11], причем число p, для которого было найдено решение, выбрано таким, что $(p-1)/2$ также является простым. Затраты на этап просеивания оценивались примерно в 50 ядро-лет, этап фильтрации — примерно в один ядро-год и этап расчетов линейной алгебры — 80 ядро-лет. Задача факторизации 768-битного модуля RSA оценивается в 900 ядро-лет, а дискретного логарифмирования для числа такой же длины — в 36500 ядро-лет. Сравнивая задачи факторизации 768 бит и 1024 бит временные затраты возрастают в 1220 раз, в то время как затраты памяти всего в 35 раз. Сложность алгоритма NFS оценивается выражением $\exp(k+o(1))(\log N)^{1/3}(\log\log N)^{2/3}$, где N — факторизуемый модуль либо простое число для дискретного логарифмирования. Параметр k зависит от алгоритма. При $k = 1{,}923$ выражение равно верхней границе для обоих алгоритмов.

Бесспорно, задачи факторизации и дискретного логарифма по своей природе имеют специфику при параллелизации вычислений и хранения промежуточных значений. Существенно, что известны эффективные применения аппаратуры специального назначения для решения некоторых этапов этих задач [12]. Грубо оценивая стоимости такого рода аппаратуры, которая будет решать задачу просеивания, получим результирующую сумму примерно в 2 млн долларов. Задача оценки стоимости этапа решения задач линейной алгебры намного сложнее, так как наработок по специализированным устройствам, решающим эту задачу, мало. Прибегая к оценке этого этапа, используя процессоры общего назначения для суперкомпьютера Титан с 300 тыс. процессоров этапа линейной алгебры для числа размерности 1024 бита, получим результат в 117 лет. С применением аппаратуры специального назначения задачи факторизации [13, 14] можно улучшить в 80 раз. При этом цена решения этапа линейной алгебры для факторизации за годовой срок суперкомпьютером Титан оценивается в 11 млрд долларов. Однозначно резонансна соизмеримая годовая сумма бюджета Национального Агентства Безопасности США.

Несмотря на то, что у истоков развития облачных технологий ставились именно задачи получения необходимой вычислительной мощности, которой не хватало персональным станциям, именно такой вид услуги сейчас получить крайне сложно. Чаще всего вместе с процессорным временем идет речь об ограничениях в оперативной памяти, размерах кеша, объемах жесткого диска. Имея оценки решения задач в ядро-годах, возникает задача в терминах облачных услуг — определить набор необходимых вычислительных узлов и рассчитать их стоимость. Например, облако Windows Azure предоставляет виртуальные машины с заданными характеристиками процессора, а именно, количества ядер, операционно-запоминающего устройства, дискового пространства. Для определения стоимости предоставляется он-лайн калькулятор [15]. С учетом специфики вычислений, необходимых для задач факторизации и дискретного логарифмирования, авторы

статьи оценили ожидаемые финансовые затраты при использовании в облаке Azure машины с 112 Gb ОЗУ, 800 Gb дискового пространства, 16 ядрами процессора в 1943 доллара в месяц. При этом конфигурация вычислительного ресурса избыточна по всем характеристикам, кроме количества ядер. Подобная услуга от Google обойдется примерно 690 долларов в месяц. Облако HP предоставит четыре виртуальных ядра, 60 Gb ОЗУ, 540 Gb дискового пространства за 985,5 долларов в месяц. Такие ресурсы неоптимальны. Эффективнее использовать много облегченных вычислительных узлов — станций с минимальными характеристиками. Для облака Google цена одного такого узла составляет примерно 40 долларов в месяц. Результирующая стоимость факторизации 1024-битного числа оценивается в сотню миллиардов долларов, что выходит дороже построения суперкомпьютера.

Заключение

Несмотря на стремительное развитие облачных технологий, применение которых должно сокращать затраты на компьютерную технику, оптимизировать использование процессорного времени, исключить простой оборудования и расход энергетических ресурсов, сегодня «облака» ориентированны на задачи массового потребителя, не связанные с проведением большого объема вычислений. Применение облачных технологий для решения задач криптоанализа, как и задач криптографии и стеганографии, в настоящее время — предмет активных исследований [16]. Состояние рынка услуг облачных вычислений позволяет сделать следующие выводы:

— услугам не хватает гибкости при выборе необходимых ресурсов, что делает применение этой технологии более дорогим по сравнению с ценой построения вычислительного комплекса с нуля;

— традиционные алгоритмы и схемы организации вычислений для решения задач криптоанализа (которые во многом сводятся к задачам управляемого перебора и просеивания) для облачных технологий должны быть пересмотрены в целях оптимизации затрат аренды «облаков».

В.К. Задірака, А.М. Кудін, П.В. Селюх, І.В. Швідченко

ХМАРНІ ТЕХНОЛОГІЇ: НОВІ МОЖЛИВОСТІ ОБЧИСЛЮВАЛЬНОГО КРИПТОАНАЛІЗУ

Хмарні обчислення як нова модель доступу до обчислювальних ресурсів видозмінюють постановки класичних задач криптографії та криптоаналізу. Особливий інтерес представляють оцінки важливого для хмарних технологій параметра — вартості наданих обчислювальних послуг при розв'язанні задач криптоаналізу симетричних і асиметричних криптосистем. Розглянуто оцінки вартості хмарних обчислень для етапів спрямованого перебору деяких загальних методів криптоаналізу симетричних і асиметричних криптосистем, а також перспективи використання хмарних обчислень для розв'язання задач криптоаналізу.

V.K. Zadiraka, A.M. Kudin, P.V. Seliukh, I.V. Shvidchenko

CLOUD TECHNOLOGIES: NEW FACILITIES OF COMPUTING CRYPTANALYSIS

Cloud computing as a new model of access to computing resources modifies classical problems of cryptography and cryptanalysis. Of special interest is an important evaluation parameter for cloud computing – the cost of computing services offered in

solving cryptanalysis problems of symmetric and asymmetric cryptosystems. The article deals with cloud computing cost evaluation for the stages of directed enumeration of some common methods of cryptanalysis of symmetric and asymmetric cryptosystems. The perspectives of using cloud computing for cryptanalysis problems solution are considered.

1. *Cisco* Global Cloud Index: Forecast and Methodology, 2014–2019 White Paper. — http://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/Cloud_Index_White_Paper.html.
2. *Mell P., Grance T.* The NIST definition of cloud computing. — http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf.
3. *Таненбаум Э., ван Стеен М.* Распределенные системы. Принципы и парадигмы. — СПб. : Питер, 2003. — 877 с.
4. *Bhaskar Prasad Rimal, Eunmi Choi, Ian Lumb.* A taxonomy and survey of cloud computing systems // 2009 Fifth International Joint Conference on INC, IMS and IDC. — 2009. — P. 44–51.
5. *Петренко А.І.* Хмарні і грід-обчислення для *E*-науки // Міжнар. конф. «Кластерні обчислення» (Київ,12–14 червня 2012 р.). — Київ : Ін-т кібернетики ім. В.М. Глушкова НАНУ, 2012. — С. 36–40.
6. *Уокер Г.* Основы облачных вычислений. Новый способ предоставления вычислительных ресурсов. — https://www.ibm.com/developerworks/ru/library/cl-cloudintro/.
7. *Hellman M.E.* A cryptanalytic time-memory trade-off // IEEE Transactions on Information Theory. — 1980. — **IT-26**, N 4. — P. 401–406.
8. *Кудін А.М., Коваленко Б.А.* Алгоритмічні аспекти пошуку прообразів хеш-функцій на прикладі MD5 // Захист інформації. — 2015. — **17**, № 3. — С. 205–210.
9. *Oechslin P.* Making a faster cryptanalytic time-memory trade-off // EUROCRYPT 2003, LNCS. — 2003. — **2729**. — P. 617–630.
10. *Sedeeq Hassn Albana Ali Al-Khazraji* Using parallel computing to implement security attack // International Journal of Computer Science and Information Security. — 2015. — **13**, N 8. — P. 35–40.
11. *New* record for discrete logarithm in a prime finite field of 180 decimal digits / C. Bouvier, P. Gaudry, L. Imbert, H. Jeljeli, E. Thomé — http://caramel.loria.fr/p180.txt.
12. *Geiselmann W., Steinwandt R.* Non-wafer-scale sieving hardware for the NFS: another attempt to cope with 1024-bit // EUROCRYPT 2007, LNCS. — 2007. — **4515**. — P. 466–481.
13. *Geiselmann W., Köpfer H., Steinwandt R., Tromer E.* Improved routing-based linear algebra for the number field sieve // Information Technology: Coding and Computing. — 2005. — **1**. — P. 636–641.
14. *Задирака В.К., Кудин А.М., Олексюк А.С.* Адаптивные алгоритмы получения простых чисел и их применение в криптографии // Компьютерная математика. — 2007. — № 1. — С. 54–61.
15. *Microsoft* Azure. Cloud services pricing. — http://azure.microsoft.com/en-us/pricing/details/cloud-services.
16. *Задирака В.К., Кудин А.М.* Облачные вычисления в криптографии и стеганографии // Кибернетика и системный анализ. — 2013. — **49**, № 4. — С. 113–119.

Получено 10.12.2015