

Путевые заметки: IETF80

Пленарное заседание: достаточно ли Интернету одного протокола - http?

Некоторые еще помнят времена, когда веб-приложения требовали полной перезагрузки страниц при выборе пунктов меню или при пользовательском вводе. Учитывая, что и средняя пропускная способность тогда была значительно ниже, чем сегодня, можно только удивляться терпению пользователей!

Веб 2.0 (http://ru.wikipedia.org/wiki/Веб_2.0) положил этим мучениям конец. Веб 2.0 – это не новая версия протокола, с технологической точки зрения - это концепция веба как платформы для построения приложений, в отличие от приложений для десктопа или операционной системы. В социальном плане, веб 2.0 - это система, позволяющая пользователям интерактивно общаться и обмениваться друг с другом текстом, изображениями и видео. Концепция, широко используемая в социальных сетях.

С технической точки зрения, Веб 2.0 базируется на технологиях Ajax (<http://ru.wikipedia.org/wiki/Ajax>) и Adobe Flex (adobe Flash), которые позволяют генерировать веб контент динамически, причем на стороне клиента. В этом заключается его существенное отличие от своего предшественника - Веб 1.0, - когда по существу загружались статические "скриншоты".

В результате веб-приложения сегодня выглядят и работают ничуть не хуже традиционных. Преимущества же их очевидны - новая функциональность и апгрейды доступны пользователю непосредственно в момент работы с приложением, когда удаленный код загружается и выполняется в его браузере. Для разработчиков это означает существенное сокращение времени на «доставку» новой функциональности пользователю, стимулируя поистине неограниченный инновационный цикл.

С другой стороны, цикл обновления традиционных приложений, как правило занимает гораздо больше времени. Это, в частности, связано с необходимостью поддержки приложения на нескольких платформах и операционных системах. С распространением же многообразных мобильных устройств, становящихся одним из основных средств доступа к Интернету, задача это становится почти непосильной.

Правда, многие веб-приложения также требуют плагинов, например Adobe Flash или Microsoft Silverlight, но поддержка ограниченного числа базовых технологий не представляет существенной проблемы. Более того, в новой версии языка HTML - HTML5 (<http://ru.wikipedia.org/wiki/HTML5>), эта функциональность станет частью стандарта и будет поддерживаться самими браузерами.

Превращается ли веб-браузер в операционную систему нового поколения? Какова роль стандартов и, в частности, IETF в области веб-приложений? И должны ли все строительные блоки быть одного цвета? Этим вопросам было посвящено техническое пленарное заседание в первый день работы IETF80 в Праге.

Джонатан Розенберг (Jonathan Rosenberg), Главный Стратег по технологиям компании Скайп, в своем выступлении выбрал SIP (<http://ru.wikipedia.org/wiki/SIP>) в качестве иллюстрации недостатков старой парадигмы разработки протоколов и приложений.

Работа на протоколе SIP началась в 1995 году, первая спецификация была выпущена в 1999, обновленная и скорректированная версия - в 2002. Протокол SIP используется для установления и управление соединением между абонентами, находящимися в различных сетях. Для собственно передачи данных - голоса и видео, используется другой протокол RTP, разработанный еще в 1996 году. Модель SIP была основана на стандартной архитектуре Интернета - прозрачная сеть и "умные" конечные устройства.

С одной стороны, SIP можно смело назвать успешной технологией. Сотни устройств сегодня поддерживают SIP, начиная от телефонных аппаратов (Avaya, Cisco, Nortel, Siemens), смартфонов и заканчивая офисными телефонными станциями и шлюзами в традиционные телефонные сети. Эта технология также используется многими сервис-провайдерами, включая Скайп, который применяет SIP для шлюзования звонков в традиционные телефонные сети.

С другой стороны, замечает Джонатан, с этим протоколом связаны и существенные неудачи.

Во-первых, неудача инновации. С SIP связывали новое поколение коммуникаций между людьми, открывающее возможность обмена многообразной информацией, немислимой в традиционной телефонии. В результате основной функциональностью оказались все те же функции инициации, установления и завершения соединения, все та же телефония, но в новой обертке. Причиной этому традиционный цикл разработки технологии, включающий начальную потребность, заинтересованность производителей оборудования, процесс стандартизации, внедрения в оборудовании и, наконец, использования сервис-провайдерами. Цикл, занимающий годы.

Во-вторых, несмотря на стандартизацию, взаимодействие между устройствами различных производителей оставляет желать лучшего. Опять же, причиной здесь является длительный цикл разработки, нечувствительный к новым требованиям и меняющимся условиям. Но также и сложность системы, более ста различных стандартов и спецификаций, играет существенную роль. К тому же, многие производители взяли часть функциональности SIP за основу и разработали собственную нестандартную функциональность.

Традиционный подход разработки системы, взятый на основу для SIP, уходит корнями в прошлое, когда основная стоимость новой системы приходилась на специализированное оборудование, и сервис-провайдеры, поставщики оборудования и пользователи являлись независимыми группами. В современном Интернете тенденции совсем другие.

Одна из тенденций заключается в том, что поставщик услуг сегодня является и производителем клиентского и серверного "оборудования", роль которого, впрочем, теперь выполняет программное обеспечение. Новая услуга - новое приложение, которое легко можно скачать из Интернета. Поставщик услуг сам разрабатывает приложения-клиенты, поэтому потребность в стандартизации не возникает. Посмотрите внимательно на приложения вашего iPhone или Android, более половины используют собственный протокол, в большинстве случаев основанный на HTTP.

Широкое распространение веб-технологий только способствует развитию этой тенденции. Приложения скачиваются и исполняются в браузере по требованию и в момент пользования услугами сервис-провайдера. Взаимодействие с другими провайдерами осуществляется путем "публикации" программного интерфейса с использованием технологии REST (<http://ru.wikipedia.org/wiki/REST>).

В результате весь цикл разработки находится в руках поставщика услуг и занимает минимум времени. Как следствие, мы наблюдаем невиданную доселе волну инноваций в Интернете. Сравнительная диаграмма традиционного и инновационного циклов разработки представлена на рисунке 1.

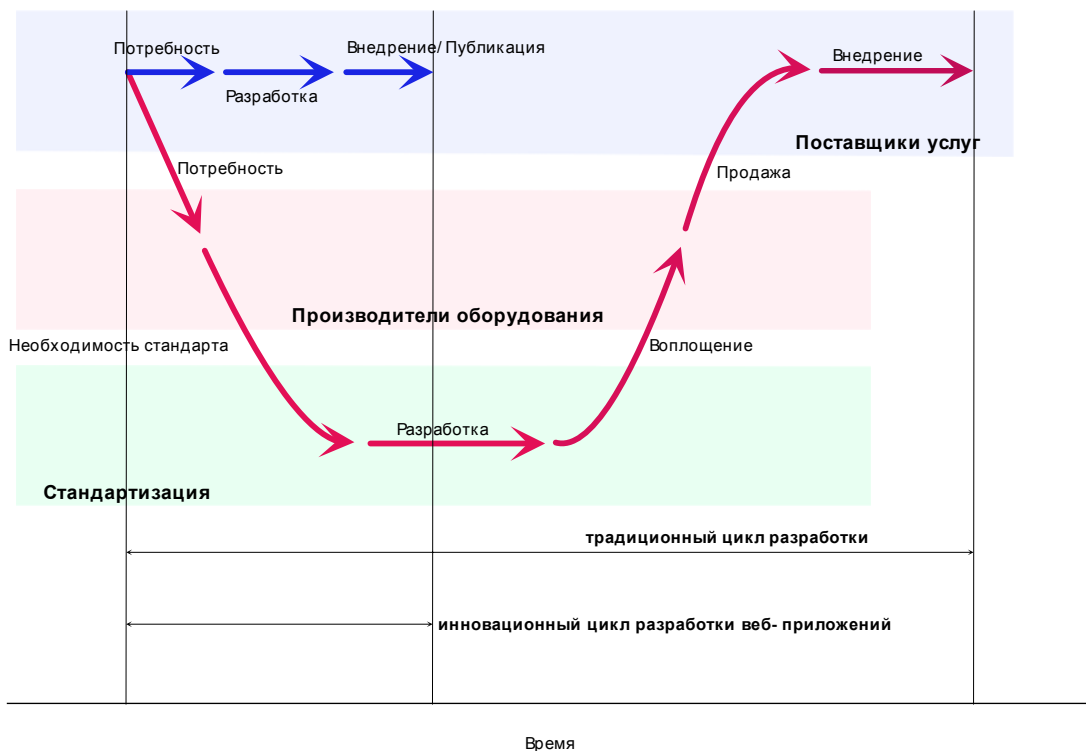


Рисунок 1. Традиционный и инновационный циклы разработки систем (из презентации Jonathan Rosenberg Chief Technology Strategist, Skype)

Однако необходимость в разработке новых протоколов и их стандартизации не отпадает, и даже, возможно, возрастает. Это отметили все выступавшие.

Например, стандартизация необходима при наличии множества провайдеров, предлагающих сходные услуги. По мнению большинства докладчиков в фокус стандартизации в первую очередь попадают протоколы междоменной коммуникации. Взаимодействие между клиентом и сервером во многих случаях стандартизации не требует, поскольку эта функциональность может быть легко "установлена" клиентом в виде скачанного кода JavaScript.

Меняется также характер стандартизации - не разработка системы, а отдельных элементов, являющихся строительными материалами будущими многообразными системами. Одним из таких блоков, например, являются технологии безопасности, требующие тщательной разработки и экспертизы.

Очевидно также, что плоды стандартизации должны быть понятны и доступны разработчикам. Другими словами, разработки IETF и W3C должны способствовать инновации, а не сдерживать ее. В противном случае - стандарты будут жить своей жизнью, а веб-приложения - своей.

Безопасность маршрутизации - новый виток

Рабочая группа SIDR (<http://datatracker.ietf.org/wg/sidr/charter/>) заканчивает работу над первой фазой повышения безопасности маршрутизации. В результате - будет опубликована серия стандартов для технологий, позволяющих криптографически удостовериться в авторизации отправителя анонса маршрута BGP.

Другими словами, будучи воплощенной, система позволит определить является ли префикс, полученный в сообщении BGP, правомочным и является ли автономная система-отправитель сообщения BGP, правомочным источником (origin) префикса. Об этой инфраструктуре, получившей название RPKI (Resource PKI), и ее применении я более подробно писал в статье "Безопасность системы маршрутизации Интернета".

Однако, хотя внедрение RPKI и, главное, применение сопутствующих технологий сервис-провайдерами, позволит ограничить вред, наносимый маршрутизационными атаками, на данном этапе полностью исключить их невозможно.

Например, злоумышленник по-прежнему может сфальсифицировать сообщения BGP и представить, что автономная система, авторизованная анонсировать некий маршрут, является его клиентом, тем самым захватив этот маршрут. Другими словами, атаки типа Pilsosov и YouTube, хотя и в ограниченном масштабе, все же возможны.

Поэтому рабочая группа начала работу над следующей фазой: возможности проверки криптографической подлинности анонса маршрута и достоверности пути, по которому этот анонс был передан.

Например, представим анонс маршрута 192.168/16 сетью AS 1 сети AS 2, и затем AS 3. По получению этого анонса сеть AS 4 сможет удостовериться, что AS 1 является правомочным "владельцем" маршрута 192.168/16, который она анонсировала. Сеть AS 2 получила этот маршрут от сети AS 1 и передала его сети AS 3. Сеть AS 3 получила этот маршрут от сети AS 2 и передала его сети AS 4.

На заседании рабочей группы в Праге был представлен проект спецификации нового расширения протокола BGP, которое получило название BGPsec (<http://datatracker.ietf.org/doc/draft-lepinski-bgpsec-protocol/>). Это расширение реализуется с помощью нового атрибута BGP BGPSEC_Path_Signatures. Атрибут этот содержит последовательность цифровых подписей для каждой сети (точнее - автономной системы), через которую было передан данный анонс маршрута.

Для лучшего понимания, как это работает, представим три сети AS1, AS2 и AS3. Допустим, что AS1 анонсирует маршрут 192.168/16 сети AS2. При использовании BGPsec этот анонс будет содержать атрибут BGPSEC_Path_Signatures состоящий из префикса 192.168/16, сети-источника AS1 и сети-пира, которой этот маршрут передан - AS2. Вся эта информация заверена подписью AS1. В свою очередь, когда сеть AS2 передаст этот анонс сети AS3 она также заверит своей подписью информацию, полученную от AS1 плюс номер автономной системы-пира, которой передается анонс, - AS3. Схематично это показано на рисунке 2.

Заметим, что с точки зрения передачи анонсов, от граничных маршрутизаторов требуется только наличия секретного ключа своей автономной системы для подписания атрибута BGPSEC_Path_Signatures.

Если AS3 захочет удостовериться в подлинности полученного анонса, ей придется проделать несколько проверок. Сначала, она должна будет убедиться, что AS1 действительно авторизована держателем соответствующего адресного пространства анонсировать этот маршрут. Для этого AS3 проверит наличие и достоверность соответствующего объекта ROA системы RPKI (см. статью "Безопасность системы маршрутизации Интернета"). Затем она должна будет последовательно проверить подписи, содержащиеся в атрибуте BGPSEC_Path_Signatures полученного анонса, чтобы убедиться в их подлинности и соответствии пути прохождения анонса.

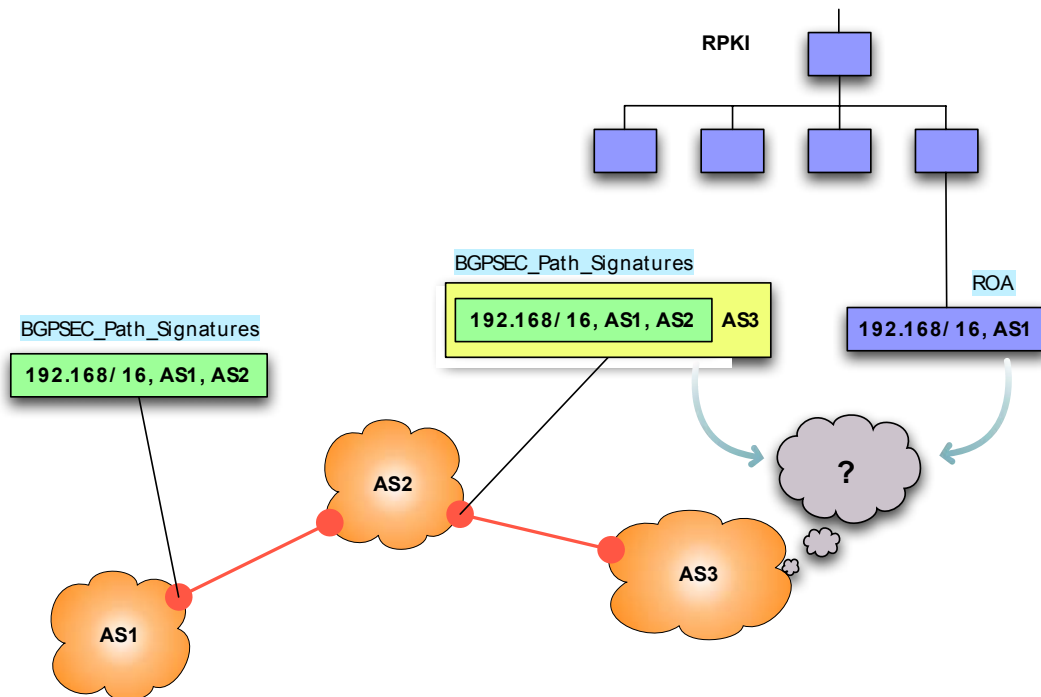


Рисунок 2. Схема работы BGPsec

Для возможности проверки подписей граничных маршрутизаторов, BGPsec определяет дополнительный тип сертификата, связывающего открытый ключ маршрутизатора с номером его автономной системы. Этот сертификат является дочерним по отношению к сертификату RPKI данной автономной системы. Таким образом при проверке анонса, проверяющая сторона сможет создать цепочку доверия, используя глобальную систему RPKI.

Отмечу, что решение, которое примет сеть AS3 по результату проверки, зависит от политики безопасности этой сети. Например, при наличии строгой политики данный путь будет удален из таблицы маршрутизации, а более гибкая политика лишь уменьшит предпочтительность полученного пути.

Новое расширение BGPsec, должно обладать одним важным свойством - а именно иметь возможность постепенного внедрения. Другими словами, необходимо, чтобы BGPsec мог быть изначально внедрен на отдельных сегментах Интернета без необходимости изменения всей глобальной системы маршрутизации в одночасье.

В проекте спецификации BGPsec это достигается следующим образом. Во-первых, данное расширение будет использоваться между маршрутизаторами пиринговых сетей если эти маршрутизаторы поддерживают BGPsec и только по взаимной договоренности. Причем эта договоренность действует отдельно для IPv4 и IPv6, а также независимо для каждого направления. Последнее позволяет конечным клиентским сетям внедрить BGPsec только для анонсирования своих маршрутов и, таким образом, избежать существенных затрат - ведь подписание анонсов гораздо менее трудоемко, чем проверка подлинности.

Пока что работа над протоколом находится в начальной стадии обсуждения. Рабочей группе предстоит разрешить множество вопросов, прежде чем детальная спецификация сможет быть воплощена в оборудовании и затем внедрена сетевыми операторами. А пока что операторы могут сфокусироваться на внедрении результатов первой фазы работы SIDR.

Нужен ли нам PKI? Или что может Symantec, чего не может DNSSEC.

Одним из важных элементов безопасности веб-приложений является использование защищенных протоколов обмена данными между клиентом (веб-браузером) и самим приложением (веб-сервером). Наиболее употребительным является протокол HTTPS, предусматривающий использования цифровых сертификатов.

Эта технология решает две задачи - во-первых, позволяет удостовериться, что имя веб-сайта, указанное в URL, действительно принадлежит владельцу этого домена, а во-вторых позволяет осуществить шифрование данных между браузером и веб-сервером для предотвращения их перехвата или модификации.

Именно поэтому перед тем как ввести данные нашей кредитной карты, мы благоразумно удостоверяемся в наличии правильного сертификата и защищенного соединения, которое некоторые браузеры указывают с помощью значка замка.

Однако каким образом браузер производит проверку сертификата?

Сертификаты, используемые HTTPS являются сертификатами X.509 Инфраструктуры открытых ключей или PKI (http://ru.wikipedia.org/wiki/Инфраструктура_открытых_ключей). Инфраструктура открытых ключей чаще всего представляет собой иерархическую структуру во главе которой стоит головной удостоверяющий центр (УЦ). Система работает таким образом, что с помощью корневого сертификата головного УЦ, можно удостовериться в подлинности всех сертификатов данной инфраструктуры.

Таких инфраструктур в Интернете насчитываются сотни. По крайней мере ваш браузер доверяет нескольким десяткам головных УЦ и соответствующие корневые сертификаты уже сохранены в браузере при установке этого программного обеспечения.

Однако тут возникает некоторый парадокс. Процедуры и качество УЦ неоднородны, причем большинство действуют вне системы присвоения имен DNS. Другими словами, для того чтобы удостовериться, что запрос на сертификат получен от полномочного владельца соответствующего доменного имени, УЦ приходится обращаться к регистрационной информации самого домена. Некоторые не делают даже этого.

Добавлю, что процесс регистрации УЦ и их корневых сертификатов в веб-браузерах также оставляет желать лучшего с точки зрения безопасности.

Возникает вопрос - почему бы не получить удостоверение от самого администратора домена непосредственно? Что, если администратор домена вместо того, чтобы приобретать TLS-сертификат у посредника, коммерческого УЦ, просто впишет самостоятельно сгенерированный сертификат в собственную зону DNS? Разумеется, в этом случае необходимо применение DNSSEC, чтобы исключить возможный подлог данных в процессе передачи. Другими словами, почему бы не использовать DNS для распространения криптографического материала (открытых ключей и сертификатов)? Ведь глобальная система DNS с внедренным DNSSEC ничем не уступает инфраструктуре PKI, и даже имеет ряд преимуществ.

Над этой проблемой начала работать относительно новая рабочая группа IETF – DANE (<http://datatracker.ietf.org/wg/dane/charter/>). Но хотя идея привлекательна, многие вопросы требуют дальнейшей проработки. Например, один из существенных вопросов, требующих разрешения – как система DANE будет сосуществовать с традиционной PKI, используемой веб-приложениями сегодня? Сегодня, при получении сертификата от веб-сервера, браузер пытается определить его подлинность, используя соответствующий корневой сертификат. Что, если сертификат будет получен из системы DNS?

Если полученный сертификат является корневым сертификатом, процедура особенно не меняется – полученные от сервера сертификат TLS должен быть удостоверен, используя цепочку доверия к этому корневому сертификату. Если же сертификат является конечным, т.е. выданным УЦ конечному пользователю, а не другому удостоверяющему центру, здесь существует несколько вариантов, показанных на диаграмме 3.

Необходимым условием является соответствие сертификата DNS, сертификату TLS, полученному от сервера. Однако является ли этой условие достаточным? Если ответ – да, то по сути проверка подлинности сертификата не проводится и не лучше ли в этом случае сравнивать (и хранить) просто криптографические ключи? Если же ответ – нет, то дополнительным условием является удостоверение подлинности сертификата TLS, используя стандартную процедуру PKI – а именно построение цепочки доверия к одному из доверенных корневых сертификатов. Если сертификаты DNS и TLS являются сертификатами некоторого внешнего УЦ, ситуация не отличается от стандартной и DNS-сертификат служит в качестве дополнительной предосторожности. В противном случае наиболее правильным решением является построение микро PKI, состоящей из корневого сертификата и дочернего сертификата, который и используется как в DNS, так и в TLS. Это связано в первую очередь с тем, что самоподписанные (корневые) сертификаты недопустимы в спецификации TLS. В этом случае записи DNS должны содержать оба сертификата – корневой и дочерний.

Другой серьезной проблемой является отсутствие защиты DNSSEC между DNS-резолверами, проверяющими достоверность полученных ответов DNS, и приложениями (веб-браузерами, например).

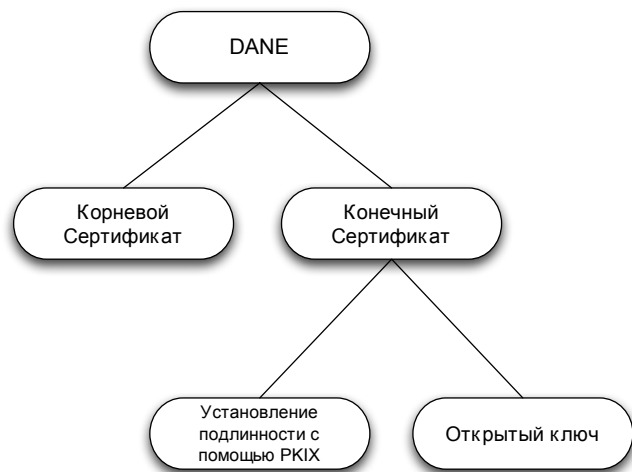


Рисунок 3. Варианты обработки сертификатов DANE

Однако работа в этом направлении активно ведется, и кто знает, может быть в недалеком будущем коммерческие сертификаты TLS канут в лету. Как и некоторые УЦ их выдающие ☺

Андрей Робачевский, Менеджер по программам ISOC

Мнения, представленные в статье, не обязательно отражают официальную позицию ISOC