Network Working Group
Request for Comments: 2181
Updates: 1034, 1035, 1123
Category: Standards Track

R. Elz
University of Melbourne
R. Bush
RGnet, Inc.

July 1997

Пояснения к спецификации DNS

Clarifications to the DNS Specification

Статус документа

Этот документ содержит спецификацию стандартного протокола, предложенного сообществу Internet, и служит приглашением к дискуссии в целях развития и совершенствования протокола. Текущее состояние стандартизации протокола можно узнать из документа Internet Official Protocol Standards (STD 1). Документ может распространяться свободно.

1. Тезисы

В этом документе рассматриваются некоторые проблемные вопросы, связанные со спецификацией DNS[1], и предлагаются корректировки для обнаруженных недостатков. Рассмотрены 8 проблемных вопросов:

- использование адреса из заголовка IP для многодомных серверов;
- значения TTL[2] в наборах записей с одинаковыми именем, классом и типом;
- корректная обработка срезов зон (zone cut);
- три второстепенных проблемы, относящиеся к записям SOA и их использованию;
- точное определение времени жизни (Time to Live или TTL);
- использование в заголовке бита TC (truncated – усечено);
- вопрос т о том, что такое аутентичное (authoritative) или каноническое (canonical) имя;
- вопрос о том, что делает корректная метка DNS.

Для первых шести вопросов корректное поведение не очевидно и мы пытаемся дать разъяснения. Два оставшихся вопроса уже включены в спецификации, однако представляется, что выполняются эти спецификации не всегда. Мы пытаемся подкрепить существующие спецификации.

Оглавление

[1]Domain Name System - система доменных имен.
[2]Time To Live – время жизни.

2. Введение

За годы использования спецификации DNS [RFC1034, RFC1035] в ней были отмечены некоторые проблемы [RFC1123]. Данный документ также отмечает некоторые дополнительные проблемные области. Эти проблемы независимы. К проблемным относятся вопросы использования многодомными серверами DNS адреса отправителя в ответах на запросы, различие значений TTL для записей DNS с одной меткой, классом и типом, а также проблема канонических имен (что это такое, как они связаны с записями CNAME, какие имена корректны для тех или иных частей DNS, корректный синтаксис имен DNS).

В этом документе даются пояснения к спецификации DNS для предотвращения перечисленных проблем. Исправлена также некоторая неоднозначность в RFC1034 относительно записей SOA, определение TTL, а также путаница с битом TC.

3. Терминология

В этом документе не используются термины MUST (**должно**), SHOULD (**следует**), MAY (**возможно**) или их отрицания в принятом для RFC значении[1]. В некоторых случаях форма выражения спецификации может показаться слишком мягкой и, следовательно, некоторые читатели могут решить, что следовать спецификации необязательно. Это неверно. Во всех случаях, когда данная спецификация указывает, что следует или необходимо выполнить некие действия или описано допустимое/недопустимое поведение, это следует рассматривать как фундаментальный аспект данной спецификации, независимо от использованных слов. Если то или иное поведение или действие действительно является необязательным, об этом явно сказано в тексте.

4. Выбор адреса отправителя в откликах сервера

Большинство клиентов DNS (если не все) ожидают получить в отклике тот же адрес, по которому был передан запрос. Это верно для серверов, выступающих в качестве клиента при рекурсивном выполнении запроса, а также для простых клиентов-преобразователей. Адрес и идентификатор (ID) в отклике используются для неоднозначных откликов и фильтрации ложных ответов. Это могло быть сделано неумышленно при разработке спецификации DNS, но сейчас это жизненный факт.

Некоторые многодомные хосты, на которых работают серверы DNS, генерируют отклики, используя в качестве адреса отправителя не тот адрес, на который клиент направил запрос. Такие отклики будут отбрасываться клиентом, поскольку адрес отправителя отклика не совпадет с адресом, по которому клиент передал исходный запрос. Т. е., отклик будет трактоваться как незапрошенный.

4.1. Выбор адреса отправителя UDP

Для предотвращения этой проблемы серверы, отвечающие на запросы по протоколу UDP, должны передавать отклики, в которых поле адреса отправителя в заголовке IP совпадает с адресом получателя в заголовке IP пакета, содержавшего запрос, связанный с этим откликом. Если это ведет к передаче отклика с адреса IP, который не разрешено использовать для таких целей, отклик может быть передан с любого корректного адреса IP, выделенного серверу. Этот адрес следует выбирать так, чтобы максимально упростить для клиента возможность использования адреса в последующих запросах. Серверы, настроенные так, что не все интерфейсы имеют равную доступность для каждого потенциального клиента, должны принимать особые меры предосторожности при передаче откликов на запросы, переданные по адресам типа anycast (любой), multicast (групповой) и т. п.

4.2. Выбор номера порта

Отклики на все запросы должны направляться в порт, из которого был передан запрос. При откликах по протоколу TCP это наследуется из свойств транспортного протокола. Для запросов, полученных по протоколу UDP, сервер должен принимать во внимание номер порта отправителя и использовать его в качестве порта назначения для отклика. Отклики всегда следует передавать из порта, в который был направлен запрос. За исключением экстраординарных ситуаций это общеизвестный номер порта, выделенный для запросов DNS [RFC1700].

5. Наборы RR

Каждая запись о ресурсе DNS (RR[2]) имеет метку[3], класс, тип и данные. Бессмысленно держать две записи, в которых совпадает все - метка, тип, класс и данные - серверам следует подавлять такие дубликаты, если они встречаются. Однако для большинства типов записей возможно существование множества RR с совпадающими метками, типом и классом, но различными данными. Такие группы записей определяются здесь как наборы RR - RRSet[4].

[1] См. RFC 2119, перевод которого имеется на сайт www.protocols.ru. *Прим. перев.*
[2] Resource Record
[3] Имя. *Прим. перев.*
[4] Resource Record Set – набор записей RR

5.1. Передача записей RR из RRSet

Запрос для указанной (или неуказанной) метки, класса и типа всегда будет возвращать все записи из связанного набора RRSet, независимо от количества RR. Отклик должен быть помечен как «усеченный», если в него не включается весь набор RRSet.

5.2. Время жизни (TTL) для записей RR в RRSet

Записи RR имеют определенный срок жизни (TTL). Можно задавать для записей RR в наборе RRSet разные значения TTL. Не известно вариантов использования этого подхода, которые невозможно было бы реализовать лучше иными путями. Кроме того, такой вариант может приводить к частичным откликам (без маркера truncated) от кэширующего сервера, на котором истекло время жизни для некоторых (но не для всех) записей RR из RRSet.

Следовательно, использование разных значений TTL в наборе RRSet настоящим документом осуждается - все значения TTL для записей RR в наборе RRSet должны быть одинаковы.

Клиенту, получившему отклик с записями RR в одном наборе RRSet, имеющими различное время жизни, следует трактовать это как ошибку. Если этот набор RRSet получен из неуполномоченного источника, клиенту следует просто игнорировать RRSet, а если значения требуются - найти способ получения ответа из уполномоченного источника. Клиентам, настроенным на отправку всех запросов одному или множеству заданных серверов, следует трактовать в данном случае эти серверы, как уполномоченные. Если уполномоченный источник передает такой некорректно сформированный набор RRSet, клиенту следует трактовать время жизни для всех RR из набора RRSet в соответствии с минимальным значением TTL в данном наборе RRSet. Сервер ни в коем случае не может передавать набор RRSet в котором значения TTL различаются.

5.3. Специальные случаи DNSSEC

Два типа записей, добавленные в DNS Security (DNSSEC) [RFC2065], требуют специального внимания при формировании наборов RRSet. К этому типу относятся записи SIG и NXT. Следует отметить, что спецификация DNS Security слишком нова и опыта ее использования еще мало. Читателям следует быть готовыми к тому, что содержащаяся в этом документе информация в части DNSSEC может устареть, когда будет принята окончательная спецификация DNS Security.

5.3.1. Записи SIG и наборы RRSet

Запись SIG обеспечивает подпись, подтверждающую корректность данных для другого набора RRSet в DNS. Когда зона подписана, каждый набор RRSet в этой зоне имеет связанную с ним запись SIG. Тип данных набора RRSet включается в данные SIG RR для индикации конкретного набора RRSet, с которым связана запись SIG. Когда перечисленные правила используются на практике, всякий раз, при включении записи SIG в отклик для подтверждения его корректности требуется также включать записи SIG для всех других наборов RRSets, связанных с соответствующим узлом. В некоторых случаях это может приводить к очень большому числу записей.

В частности, для раздела authority можно ограничиться включением лишь тех записей, для которых поле «покрытия типов» совпадает с полем типа в возвращаемом ответе. Однако при возврате записей SIG в разделе ответов в отклике на запрос записи SIG или запрос всех записей, связанных с именем (type=ANY), должен включаться весь набор SIG RRSet, как для всех остальных типов RR.

Серверы, получающие отклики с записью SIG в разделе authority или (возможно некорректно) в виде дополнительных данных, должны понимать, что весь набор RRSet почти наверняка не будет включен. Таким образом, для серверов недопустимо кэшировать эти записи SIG, чтобы они могли быть возвращены при запросах для записей SIG, получаемых этим сервером. RFC2065 требует направлять запросы SIG только уполномоченным серверам, чтобы избежать проблем и такая необходимость будет сохраняться, пока существуют серверы, не понимающие специальных свойств записей SIG. Однако аккуратная обработка записей SIG в новых реализациях должна привести в будущем к смягчению этого ограничения и от преобразователей не потребуется специальная трактовка записей SIG.

Время от времени приходится слышать, что полученный запрос для записи SIG следует пересылать уполномоченному серверу, а не отвечать на него с использованием данных из кэша. Это не обязательно - сервер, который применяет специальную обработку записей SIG, знает, в каких случаях можно корректно кэшировать записи SIG, принимая во внимание их характеристики. Тогда сервер может определить в каких случаях безопасно взять отклик из кэша, а в каких ответ недоступен и запрос нужно пересылать дальше.

5.3.2. Записи NXT

Записи NXT отличаются еще большей эксцентричностью. Для конкретного имени в зоне всегда существует только одна запись NXT и проблема RRSet может представляться тривиальной. Однако на срезе зоны обе зоны - родительская и дочерняя - (супервона и субзона в терминологии RFC2065) будут иметь записи NXT для одного и того же имени. Эти две записи NXT не формируют RRSet даже в тех случаях, когда родительская и дочерняя зоны размещаются на одном сервере. Наборы NXT RRSet всегда содержат единственную RR. В тех случаях, когда видимы обе записи NXT, существует два набора RRSet. Однако серверам не требуется трактовать этот случай, как специальный, при обработке записей NXT в отклике. Они могут принять существование двух разных наборов NXT RRSet и поступать с ними, как с двумя разными RRSet любого другого типа (т. е., кэшировать одну и игнорировать другую). Тем не менее, от защищенных серверов требуется корректная обработка записи NXT.

5.4. Прием наборов RRSet

Серверы никогда не должны объединять записи RR из отклика с RR из своего кэша для формирования набора RRSet. Если отклик содержит данные, которые будут формировать RRSet с данными из серверного кэша, сервер должен игнорировать записи RR в отклике или отбросить полностью набор RRSet, находящийся в кэше (если это приемлемо). Следовательно, не возникает проблемы с различием значений TTL между кэшем и откликом, поскольку те или другие будут игнорироваться. Т. е., один из наборов данных всегда будет некорректен, если данные в отклике и кэше различаются. Задачей сервера является определение одного корректного из этих двух наборов данных и игнорирование другого. Отметим, что при получении сервером отклика, содержащего набор RRSet, который идентичен

набору данных из его кэша, за исключением возможных различий в TTL, он может по своему разумению обновить значение TTL в кэше, заменив его TTL из полученного ответа. Это следует делать, если полученный ответ будет рассматриваться как более полномочный (см. следующий параграф), нежели ранее кэшированный ответ.

5.4.1. Ранжирование данных

При решении вопроса о восприятии RRSet из отклика или сохранении находящегося в кэше набора RRSet, серверу следует принимать во внимание уровень доверия к различным данным. Полномочный ответ из отклика следует помещать вместо кэшированных данных, которые были получены из дополнительной информации в более раннем отклике. Однако дополнительная информация из отклика будет игнорироваться, если в кэше хранятся данные из полномочного ответа или файла зоны.

Точность данных зависит от их источника. Ниже приводится градация уровней доверия для различных источников данных (уровень снижается от начала списка к его концу).

- данные из первичного файла зоны, отличные от приклеенных (glue data);
- данные из переноса зоны, отличные от приклеенных;
- полномочные данные из раздела answer полномочного отклика;
- данные из раздела authority полномочного ответа;
- приклеенные данные (Glue) из первичной зоны или переноса зоны;
- данные из раздела answer неуполномоченного ответа и неполномочные данные из раздела answer полномочных ответов;
- дополнительная информация из полномочного ответа; данные из раздела authority неполномочного ответа; дополнительная информация из неполномочных ответов.

Отметим, что раздел answer в полномочном ответе обычно содержит только полномочные данные. Однако, когда имя ищется как псевдоним (alias, см. параграф 10.1.1), полномочной должна быть только запись, описывающая этот псевдоним. Клиентам следует предполагать, что другие записи могут быть взяты из кэша сервера. Когда требуются полномочные ответы, клиенту следует повторить запрос, используя связанное с псевдонимом классическое имя.

Непроверенные RR полученные и кэшированные из наименее доверенных источников, к которым относятся данные из дополнительного раздела и данные из раздела authority неполномочного ответа, не следует кэшировать таким образом, чтобы они могли возвращаться как ответы на полученные запросы. Они могут возвращаться как дополнительная информация, когда это приемлемо. Игнорирование этого может привести к беспричинному росту доверия к сравнительно недостоверным данным.

Если используется защита DNS [RFC2065] и аутентифицированный отклик был получен и проверен, аутентифицированные таким способом данные следует рассматривать, как заслуживающие большего доверия, нежели неаутентифицированные данные того же типа. Отметим, что хотя в этом документе термин «полномочный» обозначает отклик с установленным флагом AA, DNSSEC использует доверенные цепочки записей SIG и KEY для аутентификации данных и бит AA почти не имеет значения. Однако поддерживающие DNSSEC серверы должны по-прежнему корректно устанавливать бит AA в откликах, чтобы обеспечить корректную работу обычных серверов (в настоящий момент они составляют абсолютное большинство).

Отметим, что за исключением приклеенных данных, невозможно возникновение противоречий между информацией из пары корректных первичных файлов зоны, пары корректно настроенных вторичных зон (данные из переноса зоны) или данными из корректно настроенных первичной и вторичной зоны. Когда приклеенные данные для одного имени существуют в нескольких зонах и различаются между собой, серверу имен следует предпочитать данные из первичного файла зоны данным из вторичной зоны, а в остальных случаях может выбираться любой из таких наборов данных. В тех случаях, когда возможно определение, имеет смысл выбор более полномочного источника данных. Предпочтение первичных данных (по отношению к вторичным) упрощает определение некорректных приклеенных данных в тех случаях, когда такая проблема существует. Если сервер видит, что или оба файла зоны имеет некорректную конфигурацию, порождающую конфликты, ему следует отказаться от загрузки зон с ошибками и провести подходящую диагностику.

Термин «приклеенные» выше обозначает любые записи в файле зоны, которые не являются в полной мере частью данной зоны, включая имена сервером DNS делегированных субзон (записи NS), адресные записи, сопровождающие эти записи NS (A, AAAA и т. п.), а также любые другие «приблудившиеся» данные.

5.5. Передача наборов RRSet (reprise)

Набор RRSet в любой отклик DNS следует включать лишь однократно. Он может присутствовать в любом из разделов Answer, Authority или Additional Information[1] в соответствии с реальной потребностью. Однако набор не следует повторять в той же или какой-то иной секции за исключением тех случаев, когда это явно требует спецификация. Например, отклик AXFR требует, чтобы запись SOA (это всегда набор RRSet, содержащий одну запись RR) была первой и последней в отклике. Когда требуется передача таких дубликатов, указываемые в каждом из них значения TTL должны совпадать.

6. Срезы зон

Дерево DNS делится на зоны, которые являются наборами доменов, трактуемых, как элементы для некоторых задач управления. Зоны ограничены «срезами». Каждый срез отделяет «дочернюю» зону (ниже среза) от «родительской» (выше среза). Доменное имя, появляющееся наверху зоны (сразу под срезом, отделяющим зону от ее родителя) называется «источником зоны». Имя зоны совпадает с именем домена в источнике зоны. Каждая зона включает то подмножество дерева DNS, которое расположено ниже источника зоны и выше среза, который отделяет зону от ее

[1]Ответ, полномочия или дополнительная информация.

потомков (если они имеются). Наличие среза зоны показывается в родительской зоне существованием записей NS, указывающих источник дочерней зоны. Зоны-потомки не содержат каких-либо явных ссылок на своего родителя.

6.1. Полномочия для зоны

Уполномоченные серверы для зоны перечисляются в записях NS источника информации для зоны, который, вместе с записью SOA[1], является обязательным элементом для каждой зоны. Такие серверы являются уполномоченными для всех записей в зоне, которые не относятся к другой зоне. Запись NS, которая указывает «срез» зоны, относится к создаваемой дочерней зоне, как и все остальные записи для источника этой зоны и все ее субдомены. Серверу для зоны не следует возвращать полномочные отклики на запросы, относящиеся к именам другой зоны, которая включена в записи NS (и возможно записи A) на срезе зоны, если данный сервер не обслуживает также и эту зону.

Кроме случаев, связанных с DNSSEC и рассмотренных ниже, серверам следует игнорировать все данные, кроме записей NS и необходимых записей A для поиска серверов, указанных в записях NS, которые могут быть указаны в зоне на срезе.

6.2. Проблемы DNSSEC

Механизм защиты DNS [RFC2065] вносит некоторое усложнение, поскольку введенные в нем новые записи RR весьма необычны по сравнению с другими DNS RR. В частности, запись NXT[2] RR содержит информацию о существующих и несуществующих в зоне имена и, таким образом, эта запись должна относиться к зоне, в которой она существует. Одно доменное имя может иметь разные записи NXT в родительской и дочерней зоне - обе эти записи будут корректны и не будут образовывать RRSet (см. параграф 5.3.2).

Поскольку записи NXT рассчитаны на автоматическую их генерацию, а не задание операторами DNS, серверы могут, но не обязаны, сохранять все различающиеся записи NXT, которые они могут получать, с учетом правил, указанных в параграфе 5.4.

Чтобы защищенная родительская зона могла с обеспечением защиты показать, что субзона не защищена, DNSSEC требует наличия в родительской зоне записи KEY RR, показывающей, что субзона не защищена, и аутенцифицирующей записи (записей) SIG RR, которые по определению не могут находиться в субзоне. Если субзона защищена, записи KEY и SIG будут присутствовать в этой зоне и являться полномочными, но их все равно следует включать в родительскую зону (если та защищена).

Отметим, что ни в одном из этих случаев серверу родительской зоны, который не является сервером субзоны, не следует устанавливать бит AA в каких-либо откликах для имен на срезе зоны.

7. Записи SOA

Требуется разъяснение для трех незначительных проблем, связанных с записью SOA.

7.1. Размещение записей SOA в уполномоченных откликах

Параграф 3.7 RFC 1034 указывает, что секция authority полномочного отклика может содержать запись SOA для зоны, из которой передается ответ. При обсуждении негативного кэширования в параграфе 4.3.4 RFC 1034 упоминается этот метод, но со ссылкой на дополнительный раздел отклика. Первый вариант корректен, что косвенно проиллюстрировано примером в параграфе 6.2.5 RFC 1034. Записи SOA, если они добавляются, размещаются в разделе полномочий (authority).

7.2. Значения TTL в записях SOA RR

Можно заметить, что в параграфе 3.2.1 RFC 1035, который определяет формат Resource Record, в определении поля TTL пропущена строка, которая говорит, что значение TTL записи SOA всегда следует устанавливать нулевым для предотвращения кэширования. Больше об этом нигде не упоминается и в общем случае это требование не применяется. Реализациям не следует предполагать, что запись SOA имеет значение TTL = 0 и требовать передачи записей SOA с нулевым значением TTL.

7.3. Поле SOA.MNAME

В спецификации достаточно ясно указано, что в поле MNAME записи SOA следует указывать имя первичного (master) сервера для зоны, идентифицируемой записью SOA и тем не менее это требование практически игнорируется. В это поле не следует включать имя самой зоны. Такая информация будет бесполезна, поскольку при ее получении требуется начать обработку с доменного имени в записи SOA, а это поле содержит имя зоны.

8. Время жизни (TTL)

Определение приемлемых значений поля TTL в STD 13 недостаточно четко указывает число значимых битов и тип значения (со знаком или без знака). Настоящий документ указывает, что поле TTL представляет собой целое число без знака с минимальным значением 0 и максимальным 2147483647 (т. е., $2^{31} – 1$). При передаче это значение следует помещать в младшие биты (31) 32-битового поля TTL, устанавливая для старшего бита (знак) нулевое значение.

Реализациям следует трактовать полученные значения TTL с установленным старшим битом, как нулевые (все биты имеют значение 0).

Реализации могут по своему разумению задавать верхнюю границу для получаемых значений TTL и трактовать превышающие этот порог значения, как установленный для времени жизни максимум (верхняя граница). Поле TTL задает максимальное время жизни, а не обязательный срок существования.

[1]Start of Zone of Authority - начало полномочий для зоны.
[2]Next - следующий.

9. Бит заголовка TC

Флаг TC (усечено) следует устанавливать в откликах лишь в том случае, когда RRSet требуется, как часть отклика, но набор не может быть включен в отклик целиком. Бит TC не следует устанавливать лишь на основании того, что не может быть включена какая-либо дополнительная информация, хотя свободное пространство имеется. Это правило относится и к обработке дополнительного раздела. В таких случаях весь не помещающийся в отклик набор RRSet следует опустить и передать отклик, как есть, сбросив флаг TC. Если получателю нужны опущенные данные, он может создать запрос для получения этих данных отдельно.

При установленном флаге TC часть набора RRSet, не помещающегося в отклик целиком, может оставаться в передаваемом отклике. Когда клиент DNS получает отклик с установленным флагом TC, ему следует игнорировать этот отклик и повторить запрос, используя иной механизм (например, соединение TCP), который позволит получить отклик большего размера.

10. Проблемы именования

Из некоторых разделов спецификации DNS [RFC1034, RFC1035] иногда делают вывод, что хост или, возможно, интерфейс хоста может иметь только одно полномочное (официальное) имя, называемое каноническим. В DNS нет такого требования.

10.1. Записи CNAME

Запись DNS CNAME[1] (каноническое имя) используется для обеспечения канонического имени, связанного с псевдонимом. Для любого псевдонима существует только одно каноническое имя. Это имя, в общем случае, должно присутствовать где-то в DNS, хотя имеется незначительное число приложений для псевдонимов, чьи канонические имена не определены в DNS. Псевдоним (метка записи CNAME) может иметь, если используется DNSSEC, записи SIG, NXT и KEY RR, но может не иметь других данных. Т. е., для любого имени в DNS (любого доменного имени) верно в точности одно из перечисленных ниже условий:

- существует одна запись CNAME, которая может сопровождаться записями SIG, NXT, KEY RR;
- существует одна или множество записей, но ни одна из них не является CNAME;
- существует имя без какой-либо записи RR, связанной с ним;
- имени не существует.

10.1.1. Терминология CNAME

По традиции метку записи CNAME называют просто CNAME. Это неудачная традиция, поскольку CNAME является сокращением «canonical name», а метка записи CNAME чаще всего не является каноническим именем. Однако эта терминология уже укоренилась. Поэтому следует быть осторожными при обсуждении, чтобы не путать метку и значение (каноническое имя) записей CNAME. В этом документе метка записи CNAME всегда рассматривается, как псевдоним.

10.2. Записи PTR

Путаница с каноническими имена привела к верованию в то, что в RRSet для указателя следует включать в точности одну PTR RR. Это некорректно и соответствующий раздел RFC1034 (параграф 3.6.2) указывает, что значение записи PTR должно быть каноническим именем. Т. е., оно не должно быть псевдонимом. Однако в этом разделе ничего не сказано о том, что запись PTR должна быть единственной. Не следует предполагать такого ограничения.

Отметим, что несмотря на недопустимость использования псевдонимов в качестве значения записи PTR, не предъявляется требований при обработке записей PTR невозможности преобразования в какие-либо псевдонимы. Метка, которая ищется для значения PTR, может иметь запись CNAME, т. е. это может быть псевдоним. Значение этой CNAME RR (если это не другой псевдоним, которому не следует быть) будет указывать место, где может быть найдена запись PTR. Эта запись дает результат поиска для типа PTR. Этим окончательным результатом (значение PTR RR) является метка, которая не должна быть псевдонимом.

10.3. Записи MX и NS

Доменное имя, используемое в качестве значения записи NS или части значения записи MX не должно быть псевдонимом. Это не просто разъяснение - использование псевдонима в любой из указанных позиций не обеспечит корректной работы и не приведет к ожидаемым результатам. Это доменное имя должно иметь в качестве значения по крайней мере одну адресную запись. В настоящее время в качестве таких значений могут использоваться записи типа A, однако в будущем могут появиться другие типы, дающие адресную информацию. Это также может быть RR другого типа, но ни в коем случае не CNAME RR.

Поиск записей NS или MX вызывает «обработку дополнительной секции», в которой адресные записи, связанные со значением искомой записи, указываются в конце отклика. Это позволяет избежать необходимости дополнительных запросов, которых можно было бы ожидать после первого запроса.

Обработка дополнительной секции не включает записей CNAME, не говоря уже об адресных записях, которые могут быть связаны с каноническими именами, полученными для этого псевдонима. Т. е., при использовании псевдонима в качестве значения записи NS или MX со значением NS или MX не будет возвращено адресной информации. Это может приводить к дополнительным запросам и связанной с ними избыточной нагрузке на сеть. Администраторы DNS могут легко предотвратить эту проблему, преобразуя псевдонимы и указывая каноническое имя в соответствующей записи сразу после создания или изменения такого имени. В некоторых особо сложных случаях отсутствие адресных записей в дополнительной секции результатов поиска записи NS может приводить к отказу от обработки запроса.

[1] Canonical name.

11. Синтаксис имен

Иногда считают, что серверы DNS предназначены только для отображения имен хостов Internet на адреса и обратно го преобразования. Это не соответствует действительности - DNS представляет собой иерархическую базу общего (иногда ограниченно) пользования и позволяет хранить практически любые данные с разными целями.

Сама система DNS вносит только одно ограничение на конкретные метки, которые могут использоваться для идентификации записей о ресурсах. Это ограничение связано с размером меток, который может составлять от 1 до 63 октетов, включительно. Размер полного доменного имени ограничен 255 октетами (включая разделители). Полное имя нулевого размера определено для представления корня дерева DNS, его обычно указывают точкой (.). Кроме указанных ограничений в именах меток записей о ресурсах могут использоваться произвольные двоичные строки. Подобно этому, произвольные двоичные строки могут использоваться в качестве значений любой записи, которая включает доменное имя, как часть или все свое значение (записи SOA, NS, MX, PTR, CNAME и любые другие, которые могут появиться в будущем). Реализации протокола DNS не должны вносить каких-либо ограничений на использование меток. В частности, для серверов DNS недопустимо отвергать обслуживание зон на основании того, что некоторые клиентские программы DNS не способны воспринять такие метки. Сервер DNS может быть настроен на выдачу предупреждений и даже на отказ от загрузки первичной зоны, содержащей метки, которые могут рассматриваться, как спорные, однако по умолчанию этого делать не следует.

Отметим, однако, что приложения, использующие данные DNS, могут иметь ограничения, связанные с допустимостью значений конкретных меток в рабочей среде приложений. Например, возможность использования произвольных двоичных меток в качестве значений записей MX вовсе не означает, что в задающей имя хоста части адреса электронной почты могут использоваться любые символы. Клиент DNS может вносить те или иные ограничения, связанные с обстоятельствами использования значений, служащих ключами в запросах DNS, и значений, возвращаемых DNS. Если клиент имеет такие ограничения, он принимает на себя ответственность за проверку данных, полученных от DNS, чтобы обеспечить их соответствие имеющимся ограничениям до начала использования данных.

См. также параграф 6.1.3.5 [RFC1123].

12. Вопросы безопасности

Этот документ не рассматривает вопросы безопасности.

В частности, ни какая часть раздела 4 не связана с вопросами защиты и не является полезной для решения задач обеспечения безопасности.

Параграф 5.4.1 также не связан с безопасностью. Защита данных DNS может быть обеспечена с помощью расширения Secure DNS [RFC2065], которое практически не пересекается с этим документом.

Не предполагается, что какая-либо часть данного документа что-либо добавляет к проблемам безопасности, которые могут присутствовать в DNS, или делает что-либо для решения этих проблем. Корректная реализация содержащихся в этом документе разъяснений может играть некоторую роль в ограничении распространения некорректных (но безвредных) данных в DNS, но только DNSSEC может помочь в предотвращении попыток нарушения работы системы DNS.

13. Литература

[RFC1034] Mockapetris, P., "Domain Names - Concepts and Facilities", STD 13, RFC 1034, November 1987.

[RFC1035] Mockapetris, P., "Domain Names - Implementation and Specification", STD 13, RFC 1035, November 1987.

[RFC1123] Braden, R., "Requirements for Internet Hosts – application and support", STD 3, RFC 1123, January 1989.

[RFC1700] Reynolds, J., Postel, J., "Assigned Numbers", STD 2, RFC 1700[1], October 1994.

[RFC2065] Eastlake, D., Kaufman, C., "Domain Name System Security Extensions", RFC 2065[2], January 1997.

14. Благодарности

Этот документ является результатом дискуссий в рабочей группе IETF DNSIND в течение 1995 — 1996 гг. и члены группы несут основную ответственность за высказанные здесь идеи. Особо следует отметить Donald E. Eastlake, 3rd, и Olafur Gudmundsson за их помощь при подготовке связанных с DNSSEC вопросов, а также John Gilmore за указание вопросов, не требующих прояснения. Bob Halley предложил разъяснить местоположение записей SOA в уполномоченных откликах и предоставил информацию. Michael Patton, как обычно, Mark Andrews, Alan Barrett и Stan Barber оказали существенную помощь по многим вопросам. Josh Littlefield помог убедиться, что разъяснения не вызывают проблем в некоторых вызывающих раздражение случаях.

15. Адреса авторов

Robert Elz

Computer Science

University of Melbourne

Parkville, Victoria, 3052

Australia.

EMail: kre@munnari.OZ.AU

[1] В соответствии с RFC 3232 этот документ признан устаревшим и заменен базой данных, доступной по ссылке http://www.iana.org/numbers/. *Прим. перев.*

[2] Этот документ признан устаревшим и заменен RFC 2535, который, в свою очередь, был заменен документами RFC 4033, RFC 4034, RFC 4035. *Прим. перев.*

Randy Bush
RGnet, Inc.
5147 Crystal Springs Drive NE
Bainbridge Island, Washington, 98110
United States.
EMail: randy@psg.com

Перевод на русский язык

Николай Малых
nmalykh@protocols.ru