Internet Engineering Task Force (IETF)
Request for Comments: 6891
STD: 75
Obsoletes: 2671, 2673
Category: Standards Track
ISSN: 2070-1721

J. Damas
Bond Internet Systems
M. Graff

P. Vixie
Internet Systems Consortium

April 2013

Extension Mechanisms for DNS (EDNS(0))

Механизмы расширения для DNS (EDNS(0))

Аннотация

Протокол DNS[1] включает множество фиксированных полей, для которых диапазоны значений уже заполнены или будут заполнены скоро, что не позволяет запрашивающим анонсировать свои возможности отвечающей стороне. Этот документ описывает совместимые с имеющимися реализациями механизмы, обеспечивающие возможность расширения протокола.

Документ обновляет спецификацию механизмов расширения для DNS (EDNS(0)) и отменяет действие RFC 2671 с учётом опыта использования некоторых реализаций. Документ также отменяет RFC 2673 (Binary Labels in the Domain Name System) и добавляет рассмотрение использования расширенных меток в DNS.

Статус документа

Этот документ относится к категории проектов стандартов (Internet Standards Track).

Документ является результатом работы IETF[2] и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG[3]. Дополнительная информация о документах BCP представлена в разделе 2 документа RFC 5741.

Информация о текущем статусе этого документа, обнаруженных ошибках и способах обратной связи может быть найдена по ссылке http://www.rfc-editor.org/info/rfc6891.

Оглавление

[1] Domain Name System - система доменных имён.
[2] Internet Engineering Task Force - комиссия по исследованиям Internet.
[3] Internet Engineering Steering Group - комиссия по решению инженерных задач Internet.

1. Введение

Протокол DNS [RFC1035] задаёт формат сообщений, а также стандартные форматы для опций кодирования, ошибок и сжатия имён в таких сообщениях. Максимальный размер сообщения DNS для передачи по протоколу UDP без использования предложенных здесь расширений составляет 512 байтов. Многие из предельных значений для протокола DNS, такие как максимальный размер при передаче по протоколу UDP), слишком малы для эффективной поддержки дополнительной информации, которая может передаваться DNS (например, несколько адресов IPv6 или подписи DNSSEC[1]). Кроме того, RFC 1035 не определяет каких-либо способов анонсирования возможностей другим участникам протокола.

В [RFC2671] добавлены механизмы расширения для DNS. Эти механизмы получили широкую поддержку и множество новых приложений DNS и протокольных расширения зависят от наличия этого расширения. Данный документ уточняет определения и отменяет действие [RFC2671].

Не поддерживающие расширений агенты не будут знать, как интерпретировать протокольные расширения, определённые в [RFC2671] и переопределенные здесь. Расширенным агентам нужно быть готовыми к обработке взаимодействий с не поддерживающими расширения клиентами перед лицом появления новых протокольных элементов и аккуратно переходить в режим работы без расширений DNS.

EDNS является поэтапным (hop-by-hop) расширением DNS. Это означает, что использование EDNS согласуется между каждой парой хостов в процессе распознавания (resolution) DNS, например, оконечным распознавателем (stub resolver), взаимодействующим с рекурсивным распознавателем, или рекурсивным распознавателем, взаимодействующим с полномочным сервером.

В [RFC2671] заданы расширенные типы меток. Единственным таким типом был предложенный в [RFC2673] тип «Bit-String Label» или «Binary Labels» (второй вариант стал общепринятым). По разным причинам добавление новых типов меток оказалось слишком сложным и документ [RFC2673] получил статус экспериментального (Experimental). Данный документ отменяет [RFC2673] и двоичные метки (Binary Labels). Расширенные метки сохраняются, но их применение не рекомендуется из-за практических сложностей при развёртывании. Их использование в будущем **следует** выбирать лишь после тщательной оценки препятствий к развёртыванию.

2. Терминология

Запрашивающей (Requestor) называется сторона, передающая запрос, а ответчиком (Responder) - полномочный рекурсивный распознаватель или другой элемент DNS, отвечающий на вопросы. Остальные термины определены в RFC, упоминаемых в этом документе.

Ключевые слова **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не нужно** (SHALL NOT), **следует** (SHOULD), **не следует** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе должны интерпретироваться в соответствии с RFC 2119 [RFC2119].

3. Требование поддержки EDNS

EDNS обеспечивает механизм улучшения расширяемости DNS, позволяя сделать применение DNS в Internet более разнообразным. Это делается путём обеспечения возможности использовать для DNS транспорт UDP с сообщениями, размер которых превышает заданный в RFC 1035 предел, а также предоставления дополнительного пространства для флагов и кодов возврата (RCODE). Однако опыт развёртывания показывает, что добавления новых значений RCODE следует избегать по причине сложности обновления установленных систем. Флаги **следует** применять только в случае их необходимости для преобразования DNS.

Для многих приложений может оказаться предпочтительным EDNS Option Code.

С течением времени для некоторых приложений DNS расширения EDNS стали обязательными при развёртывании. Например, DNSSEC использует дополнительное пространство флагов, привнесённое EDNS, для указания запросов, в отклики на которые нужно включать данные DNSSEC.

С учётом роста размера откликов DNS при включении крупных элементов данных, таких как записи AAAA, информация DNSSEC (например, RRSIG или DNSKEY) и большие записи TXT, дополнительное пространство данных UDP, обеспечиваемое EDNS, может помочь в плане расширяемости DNS без широкого использования TCP в качестве транспорта DNS.

4. Изменение сообщений DNS

4.1. Заголовок сообщения

Второе полное 16-битовое слово заголовка сообщения DNS делится на 4-битовые поля OPCODE и RCODE, а также множество 1-битовых флагов (см. параграф 4.1.1 в [RFC1035]). Некоторые из флагов были зарезервированы на

[1]DNS Security - защита DNS.

будущее и сейчас уже почти все флаги распределены. Большинство значений RCODE также занято. Псевдо-RR OPT, описанные ниже, содержат расширения для поля RCODE, а также дополнительные биты флагов.

4.2. Типы меток

Первые 2 бита в формате передачи (wire format) метки домена используются для обозначения типа метки. В [RFC1035] выделено 2 из 4 возможных типов, а 2 оставшихся зарезервированы. В [RFC2671] были определены дополнительные типы меток. Использование 2-битовой комбинации, определённой в [RFC2671], для идентификации расширенных типов меток остаётся в силе. Однако опыт показал, что развёртывание новых типов меток оказалось значительно сложнее и поэтому оно рекомендуется лишь после внимательной оценки вариантов и требований разворачиваемой системы.

4.3. Размер сообщений UDP

Традиционные сообщения DNS ограничены размером 512 байтов при передаче по протоколу UDP [RFC1035]. Рост объёма данных, которые могут передаваться в сообщениях DNS, сделал ограничение в 512 байтов неудобным. Например, включение записей DNSSEC зачастую создаёт отклики, размер которых значительно больше 512 байтов.

EDNS(0) указывает способ анонсирования дополнительных возможностей (включая увеличенный размер сообщений), которые предназначены для предотвращения отсечки откликов UDP, приводившей к необходимости использовать транспорт TCP. Расширение обеспечивает возможность доставки более крупных сообщений без перехода на TCP.

5. Расширенные типы меток

Первый октет представления метки DNS при передаче (on-the-wire) указывает тип метки. Базовая спецификация DNS [RFC1035] выделяет для этого 2 старших бита данного октета.

В [RFC2671] определён тип 0b01 для индикации расширенных меток DNS. Конкретный расширенный тип метки указывается 6 младшими битами первого октета. Таким образом, типы расширенных меток указываются значениями 64 - 127 (0b01xxxxxx) в первом октете метки.

Расширенные типы меток оказались очень сложны для развёртывания по причине отсутствия поддержки в клиентах и промежуточных шлюзах, как описано в [RFC3363], который перевёл [RFC2673] в статус экспериментального (Experimental), и [RFC3364], где описаны «про и контра» для этого. Таким образом, приложениям, планирующим применять расширенные метки, **следует** взвесить стоимость развёртывания и возможность реализации другими способами.

В заключение отметим, что реализациям **недопустимо** генерировать двоичные метки (Binary Labels), поскольку они признаны устаревшими.

6. Псевдо-RR OPT

6.1. Определение записи OPT

6.1.1. Базовые элементы

Псевдо-RR OPT (иногда говорят мета-RR) **может** быть добавлена в раздел дополнительных данных запроса.

OPT RR имеет тип 41.

При наличии записи OPT в запросе соответствующие спецификации распознаватели **должны** включать запись OPT в свой отклик.

Запись OPT не содержит каких-либо данных DNS и применяется лишь для передачи управляющей информации, которая относится к последовательности запрос-отклик в конкретной транзакции. Записи OPT RR **недопустимо** кэшировать, пересылать, сохранять в локальных первичных (master) файлах или загружать из них.

OPT RR **может** размещаться в любом месте раздела дополнительных данных. При наличии OPT RR в любом сообщении DNS, она **должна** быть единственной такой записью в сообщении. При получении запроса с множеством OPT RR **должна** возвращаться ошибка FORMERR (RCODE=1). Гибкость размещения OPT RR не отменяет необходимости для записи TSIG или SIG(0) RR быть последней в дополнительном разделе, если такая запись имеется.

6.1.2. Формат передачи

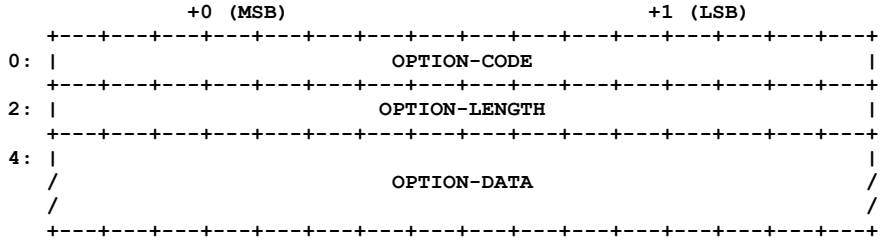
OPT RR имеет фиксированную часть и набор опций переменного размера в форме пар {attribute, value}. Фиксированная часть содержит некоторые метаданные DNS, а также небольшой набор базовых элементов расширений, которые предполагаются достаточно популярными и будут расходовать излишнее пространство при кодировании в форме {attribute, value}.

Структура фиксированной части OPT RR показана в таблице ниже.

Формат OPT RR.

Имя поля	*Тип поля*	*Описание*
NAME	domain name	Должно быть 0 (корневой домен)
TYPE	u_int16_t	OPT (41)
CLASS	u_int16_t	Размер данных UDP запрашивающей стороны
TTL	u_int32_t	Расширенный код RCODE и флаги
RDLEN	u_int16_t	Размер всех RDATA
RDATA	octet stream	Пары {attribute,value}

Переменная часть OPT RR может содержать множество (возможно пустое) опций в RDATA. Каждая опция **должна** трактоваться как битовое поле. Кодирование опций показано ниже.



OPTION-CODE

Назначается по процедуре Expert Review, как определено рабочей группой DNSEXT и IESG.

OPTION-LENGTH

Размер OPTION-DATA в октетах.

OPTION-DATA

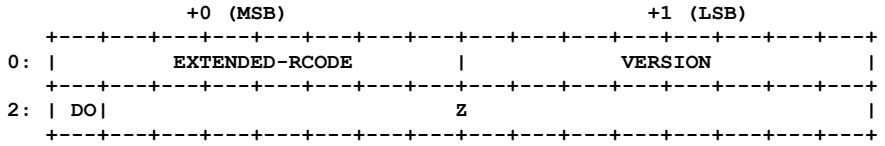
Зависит от OPTION-CODE. **Должно** рассматриваться как битовое поле.

Порядок размещения пар для отдельных опций не задаётся. Если одна опция меняет поведение другой или множество опций так или иначе связаны между собой, результат не зависит от порядка при кодировании в RDATA.

Любые значения OPTION-CODE, не понятные ответчику или запрашивающей стороне, **должны** игнорироваться. Спецификации опций могут пожелать включения сигналов подтверждения. Например, спецификация опции может указывать, что при распознании и поддержке ответчиком опции XYZ, он **должен** включать опцию XYZ в свой отклик.

6.1.3. Использование поля OPT Record TTL

Расширенные значения RCODE и флаги, которые OPT хранит в поле RR TTL[1], имеют показанную ниже структуру.



EXTENDED-RCODE

Формирует старшие 8 битов расширенного 12-битового значения RCODE (вместе с 4 битами, определёнными в [RFC1035]). Отметим, что EXTENDED-RCODE = 0 указывает применение обычных RCODE (от 0 до 15).

VERSION

Указывает уровень реализации установщика. Полное соответствие данной спецификации указывается версией 0. Запрашивающим настоятельно рекомендуется указывать в этом поле наименьший уровень реализации, способный выразить транзакцию, для минимизации нагрузки на ответчик и сеть при определении наибольшего общего уровня реализации, поддерживаемого запрашивающим и ответчиком. Стратегия нумерации версий запрашивающей стороны в идеале **может** быть настраиваемой в процессе работы. Если ответчик не реализует запрошенный уровень VERSION, он **должен** отвечать с RCODE=BADVERS. Все ответчики **должны** быть ограничены по формату уровнем VERSION в запросе, но в поле VERSION каждого отклика **следует** указывать высший уровень реализации ответчика. В результате вызывающая сторона будет узнавать уровень реализации ответчика как дополнительный результат каждого отклика, включая отклики с ошибками и RCODE=BADVERS.

6.1.4. Флаги

DO

Бит DNSSEC OK, как определено в [RFC3225].

Z

Устанавливается в 0 отправителями и игнорируется получателями, пока не будет изменено последующей спецификацией.

6.2. Поведение

6.2.1. Поведение кэша

Записи OPT **недопустимо** кэшировать.

6.2.2. Откат к старой версии

Если запрашивающий видит, что удалённая сторона не поддерживает EDNS(0), он **может** передавать запросы без записи OPT. Эта информация может кэшироваться на короткое время для предотвращения запоздалого отката в будущем. Однако если нужно расширение DNSSEC или какая-либо будущая опция, которой требуется EDNS, откат к старой версии выполнять не следует, поскольку такие опции сигнализируются лишь с помощью EDNS. Если реализация видит, что некоторые серверы для зоны поддерживают EDNS(0), а другие требуют использования TCP для получения всех данных, предпочтение **может** быть отдано серверам EDNS(0). Разработчикам **следует** анализировать этот выбор и его влияние на обеих сторонах.

[1]Time to Live - время жизни.

6.2.3. Размер данных запрашивающей стороны

Размер данных UDP запрашивающей стороны (представляется полем RR CLASS) - это число октетов наибольшего поля данных UDP, который может быть собран и доставлен сетевым стеком запрашивающего. Отметим, что это значение может превышать path MTU (с фрагментацией или без неё).

Значения меньше 512 **должны** трактоваться как 512.

Запрашивающей стороне **следует** указывать в этом поле размер, который она реально может воспринимать. Например, если запрашивающий размещается за межсетевым экраном, который блокирует фрагменты IP, ему **не следует** выбирать значение, которое вызовет фрагментацию. Это будет препятствовать получению больших откликов и может вызвать откат к старой версии. Информация может автоматически определяться реализацией или задаваться администратором сети.

Отметим, что при 512 октетах данных UDP для сборки IP требуется буфер в 576 октетов. Выбор значений от 1280 до 1410 байтов для IP (v4 или v6) в сети Ethernet будет разумным.

Если фрагментация не вызывает проблем, разработчикам **следует** предусматривать более крупные значения. Реализациям **следует** использовать их максимальные настроенные или реализованные значения в качестве стартовой точки транзакций EDNS при отсутствии информации об отвечающем сервере.

Выбор очень большого значения гарантированно приведёт к фрагментации на уровне IP и может препятствовать получению откликов в результате потери одного фрагмента или ошибок в настройке межсетевого экрана.

Максимальный размер данных запрашивающей стороны может меняться с течением времени. Его **недопустимо** кэшировать для использования за пределами транзакции, в которой этот размер анонсирован.

6.2.4. Размер данных ответчика

Максимальный размер данных ответчика может изменяться с течением времени, но разумно считать, что он не будет меняться для двух близко расположенных транзакций (например, произвольная операция QUERY для определения максимального размера данных UDP у ответчика, за которой следует операция UPDATE, использующая этот размер). Это считается предпочтительным по сравнению с использованием TCP для запросов большого размера, если есть основания предполагать, что ответчик реализует EDNS, а запрос не будет помещаться в используемый по умолчанию размер данных 512 байтов.

6.2.5. Выбор размера данных

В связи с издержками транзакций не рекомендуется анонсировать архитектурное ограничение в качестве максимального размера данных UDP. Даже в системных стеках, способных собирать дейтаграммы размером 64 Кбайт, использование памяти на нижних уровнях системы будет играть роль. Хорошим компромиссом может быть использование максимального размера данных EDNS 4096 октетов в качестве стартовой точки.

Запрашивающая сторона может реализовать откат к меньшим анонсируемым размерам для работы в среде с межсетевыми экранами или при наличии иных сетевых ограничений. Запрашивающему **следует** выбирать применение механизма отката, начиная с большого размера, такого как 4096. Если это вызовет отказ, **следует** попытаться использовать значения 1280-1410 байтов, поскольку они имеют высокий шанс передачи в одном кадре Ethernet. Если отказ произойдёт и в этом случае, запрашивающая сторона **может** выбрать размер 512 октетов, который в большинстве случаев может потребовать использования транспорта TCP.

Значения меньше 512 **должны** трактоваться как 512.

6.2.6. Поддержка на промежуточных устройствах

В сети, передающей трафик DNS, может быть активное оборудование, не относящееся к элементам, непосредственно участвующим в процессе преобразования DNS (оконечные и кэширующие распознаватели, полномочные серверы), но влияющее на передачу сообщений DNS (например, межсетевые экраны, балансировщики, прокси и т. п.), которые называют промежуточными устройствами (middlebox).

Соответствующим спецификации промежуточным устройствам **недопустимо** ограничивать размер сообщений DNS по протоколу UDP до 512 байтов.

Промежуточным устройствам, которые просто пересылают запросы рекурсивным распознавателям, **недопустимо** менять или удалять содержимое записи OPT в любом направлении.

Промежуточным устройствам с дополнительной функциональностью, таким как отвечающие на запросы или выполняющие интеллектуальную пересылку, **следует** поддерживать способность обрабатывать записи OPT и действовать на основе их содержимого. Такие устройства **должны** считать входящие запросы и любые исходящие запросы отдельными транзакциями, если характеристики сообщений различаются.

Более глубокое рассмотрение такого оборудования и вопросов его взаимодействия с трафиком DNS приведено в [RFC5625].

7. Транспорт

Наличие псевдо-RR OPT в запросе следует считать индикацией запрашивающей стороной полной реализации данной версии EDNS и возможности корректного понимания любых откликов, которые соответствуют спецификации.

Отсутствие записи OPT в запросе **должно** считаться указанием того, что запрашивающая сторона не реализует данную спецификацию совсем, а ответчику **недопустимо** включать запись OPT в свои отклики.

Расширенные агенты **должны** быть готовы обрабатывать взаимодействие и обычными клиентами перед лицом новых элементов протокола и при необходимости аккуратно откатываться к DNS без расширения.

Ответчики, отказавшиеся от реализации определённых здесь расширений, **должны** возвращать код (RCODE) FORMERR для сообщений, содержащих запись OPT в дополнительном разделе. Включение записи OPT в такие отклики **недопустимо**.

При возникновении проблем с обработкой самой записи OPT, таких как опция с некорректным форматом или выходящими за допустимые пределы значениями, **должна** возвращаться ошибка FORMERR. В таких случаях отклик **должен** включать запись OPT. Это предназначено для того, чтобы запрашивающая сторона могла отличать не поддерживающие EDNS серверы от случаев ошибок формата в EDNS.

Минимальный отклик **должен** включать заголовок DNS, раздел вопроса (question) и запись OPT. Такой вариант **должен** применяться также при возврате усечённого отклика (с битом TC в заголовке DNS).

8. Вопросы безопасности

Задание размера буфера на запрашивающей стороне может открывать возможность DoS[1]-атак на DNS, если ответчики станут передавать слишком большие сообщения, которые промежуточные узлы не смогут переслать, что может приводить к ICMP-штормам между запрашивающей и отвечающей стороной.

Анонсирование слишком большого буфера UDP может приводить к отбрасыванию сообщений DNS промежуточными устройствами (параграф 6.2.6). Это вызовет повторы передачи без надежды на успех. Известно, что некоторые устройства отбрасывают фрагментированные пакеты UDP.

Анонсирование слишком малых буферов UDP может вызывать откат к протоколу TCP с соответствующим влиянием на работу серверов DNS. Это особенно важно для DNSSEC, где ответы могут быть большими.

9. Взаимодействие с IANA

Агентство IANA выделило код типа RR 41 для записей OPT.

В [RFC2671] задано множество субреестров IANA в реестре «DOMAIN NAME SYSTEM PARAMETERS»:

- DNS EDNS(0) Options;
- EDNS Version Number;
- EDNS Header Flags.

В дополнение к этому добавлены записи в имеющиеся реестры:

- EDNS Extended Label Type в реестр DNS Label Types;
- Bad OPT Version в реестр DNS RCODES.

Агентство IANA заменило в этих реестрах и субреестрах ссылки на [RFC2671] ссылками на этот документ.

В [RFC2671] был создан реестр DNS Label Types, который сохраняется открытым.

Для реестра DNS Label Types используется регистрационная процедура Standards Action.

Этот документ выделяет код опции 65535 в реестре DNS EDNS0 Options как резервный (Reserved for future expansion).

Текущее состояние реестра IANA для кодов опций EDNS на момент публикации документа имеет вид:

- 0-4 выделены с указанными в реестре ссылками;
- 5-65000 доступны для распределения;
- 65001-65534 для локального применения и экспериментов (Local/Experimental);
- 65535 зарезервирован для будущего расширения.

В [RFC2671] размер значений RCODE увеличен с 4 до 12 битов. Это позволяет использовать более 16 разных значений RCODE, разрешённых в [RFC1035]. Для добавления кодов применяется процедура IETF Review.

Этот документ выделяет значение EDNS Extended RCODE 16 для кода «BADVERS» в реестре DNS RCODES.

В [RFC2671] предложено записывать назначение расширенных типов меток 0bxx111111 как «резерв для будущих расширенных типов меток» (Reserved for future extended label types», в настоящее время реестр IANA содержит запись «резерв для будущих расширений» (Reserved for future expansion). Это предложение предполагало запрос на создание нового реестра расширенных типов меток, но по причине возможной путаницы вместо этого новые регистрации вносились в общий реестр DNS Label Types, содержащий также изначально определённые в [RFC1035] типы. В результате реестр Extended Label Types не был создан и все типы регистрируются в реестре DNS Label Types.

Этот документ отменяет Binary Labels. Поэтому статус регистрации DNS Label Types для «Binary Labels» меняется на «Historic».

Для назначения новых флагов EDNS(0) требуется процедура IETF Standards Action. Флаги **следует** применять только в тех случаях, когда они требуются для преобразования (resolution) DNS. Во многих случаях лучше использовать EDNS Option Code.

Для создания новых записей в реестре EDNS Version Number требуется процедура IETF Standards Action. Выделение значения EDNS Option Code выполняется по процедуре Expert Review. В соответствии с этим документом IANA поддерживает реестр для EDNS Option Code.

[1]Denial-of-service - отказ в обслуживании.

9.1. Изменение названия реестра DNS EDNS0 Option Code[1]

Этот документ меняет название имеющегося реестра DNS EDNS0 Options на DNS EDNS0 Option Codes (OPT) и процедуру регистрации новых значений на Expert Review.

Коды опций следует распределять великодушно, избегая дублирования функциональности.

10. Литература

10.1. Нормативные документы

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2671] Vixie, P., "Extension Mechanisms for DNS (EDNS0)", RFC 2671, August 1999.

[RFC3225] Conrad, D., "Indicating Resolver Support of DNSSEC", RFC 3225, December 2001.

10.2. Дополнительная литература

[RFC2673] Crawford, M., "Binary Labels in the Domain Name System", RFC 2673, August 1999.

[RFC3363] Bush, R., Durand, A., Fink, B., Gudmundsson, O., and T. Hain, "Representing Internet Protocol version 6 (IPv6) Addresses in the Domain Name System (DNS)", RFC 3363, August 2002.

[RFC3364] Austein, R., "Tradeoffs in Domain Name System (DNS) Support for Internet Protocol version 6 (IPv6)", RFC 3364, August 2002.

[RFC5625] Bellis, R., "DNS Proxy Implementation Guidelines", BCP 152, RFC 5625, August 2009.

Приложение A. Отличия от RFC 2671 и 2673

Ниже приведён список существенных отличий от RFC 2671 и RFC 2673.

- Поддержка записей OPT стала обязательной.
- Типы расширенных меток остались, но их применение не рекомендуется в качестве общего решения, поскольку наблюдались значительные трудности с их развёртыванием Internet, как показала работа с типом «Binary Labels».
- RFC 2673 с определением «Binary Labels» переведён в статус Experimental и запрошен перевод в «Historic».
- Внесены изменения в выбор размера буферов EDNS и приведены рекомендации по таком выбору.

Адреса авторов

Joao Damas
Bond Internet Systems
Av Albufera 14
S.S. Reyes, Madrid 28701
ES
Phone: +1 650.423.1312
EMail: joao@bondis.org

Michael Graff
EMail: explorer@flame.org

Paul Vixie
Internet Systems Consortium
950 Charter Street
Redwood City, California 94063
US
Phone: +1 650.423.1301
EMail: vixie@isc.org

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru

[1] В исходном документе название и содержимое параграфа были иными. См. https://www.rfc-editor.org/errata/eid3604. *Прим. перев.*