Internet Engineering Task Force (IETF)
Request for Comments: 8109
BCP: 209
Category: Best Current Practice
ISSN: 2070-1721

P. Koch
DENIC eG
M. Larson
P. Hoffman
ICANN

March 2017

Initializing a DNS Resolver with Priming Queries

Инициализация распознавателя DNS с подготовительными запросами

Аннотация

В этом документе описаны запросы, которые распознавателю DNS следует выдавать (emit) для инициализации своего кэша. В результате этого распознаватель получает как текущий набор записей ресурсов NS (Resource Record Set или RRset) для корневой зоны, так и требуемые адресные сведения для достижения корневых серверов.

Статус документа

Документ относится к категории Internet Best Current Practice.

Документ является результатом работы IETF[1] и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG[2]. Дополнительную информацию о документах BCP можно найти в разделе 2 в RFC 7841.

Информацию о текущем статусе документа, ошибках и способах обратной связи можно найти по ссылке http://www.rfc-editor.org/info/rfc8109.

Оглавление

1. Введение

Рекурсивным распознавателям DNS нужна отправная точка для распознавания по запросам. В [RFC1034] описан базовый сценарий для рекурсивных распознавателей - они начинают с пустого кэша и некой конфигурации для нахождения имён и адресов корневых серверов DNS. Эта конфигурация описана в [RFC1034] как список серверов, которые будут давать полномочные ответы на запросы о корне. Это стало распространённым вариантом реализации рекурсивных распознавателей и является темой этого документа. В этом документе описаны шаги, требуемые для этого варианта реализации. Отметим, что это не единственный способ запуска рекурсивного сервера имён с пустым кэшем, но в [RFC1034] описан только он. Некоторые разработчики выбрали иные варианты, часть которых работает хорошо, а другие сталкиваются с отказами (иногда катастрофическими) при некоторых условиях. Например, реализация, получающая адреса корневых серверов только из конфигурации, а не от DNS (как описано в этом документе) будет иметь устаревшие данные, что может замедлить распознавание.

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

В этом документе рассматриваются только рекурсивные серверы имён (рекурсивные распознаватели, распознаватели) для класса IN.

[1]Internet Engineering Task Force - комиссия по решению инженерных задач Internet.
[2]Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

2. Описание подготовки

Подготовка (priming) - это поиск списка корневых серверов по конфигурации, где указаны все или некоторые предполагаемые адреса IP корневых серверов. Рекурсивный распознаватель начинает с отсутствия сведений о корневых серверах, а заканчивает списком их имён и адресов. Подготовка описана в параграфах 5.3.2 и 5.3.3 [RFC1034]. В описании использован сценарий, где рекурсивный сервер является также полномочным, что не очень распространено.

Настроенный список IP-адресов корневых серверов обычно поступает от производителя или дистрибьютора программ рекурсивного сервера. Список обычно является корректным и полным на момент поставки, но может устареть со временем.

Список операторов корневых серверов и доменных имён, связанных с каждым из них, не меняется с 1997 г. Однако в настоящее некоторые адреса (IPv4 и IPv6) для доменных имён корневых серверов изменились. Тем не менее, исследования показывают, что после такой замены некоторые распознаватели не получили новых адресов. Поэтому важно, чтобы распознаватели могли справиться с изменениями даже без смены конфигурации их операторами. Смена корневых серверов является основной причиной, по которой распознаватели для получения полного списка корневых серверов должны выполнять подготовку, а не просто обращаться к настроенному списку.

3. Подготовительные запросы

Подготовительным считается запрос DNS, служащий для получения сведений о корневом сервере в распознавателе. Запрос содержит QNAME «.», QTYPE NS и передаётся по одному из адресов, заданных в конфигурации рекурсивного распознавателя. Подготовительный запрос может передаваться по протоколу UDP или TCP. При передаче через UDP порт отправителя **следует** выбирать случайно (см. [RFC5452]). Бит желательности рекурсии (Recursion Desired или RD) **может** иметь значение 0 или 1, хотя значение 1 не определено для подготовительных запросов.

Рекурсивному распознавателю **следует** использовать EDNS(0) [RFC6891] для подготовительных запросов, а также **следует** анонсировать и обслуживать размер сборки не менее 1024 октетов [RFC3226]. Это разрешает отклики величиной до полного размера подготовительного отклика (см. параграф 4.2) для текущего набора корневых серверов. Установка бита DNSSEC OK (DO), определённого в [RFC4033], обсуждается в параграфе 3.3.

3.1. Повторение подготовительных запросов

Рекурсивному распознавателю **следует** передавать подготовительный запрос лишь тогда, когда это необходимо, например при старте распознавателя с пустым кэшем или при завершении срока действия NS RRset для корневой зоны. Поскольку записи NS для корня не являются особыми, срок их действия у рекурсивного распознавателя истекает в соответствии со значениями TTL в них (отметим, что рекурсивный распознаватель **может** заранее извлечь NS RRset до завершения срока действия прежнего набора.)

Если на подготовительный запрос не получено отклика, рекурсивный распознаватель должен повторить запрос с другим целевым адресом, взятым из конфигурации.

3.2. Выбор цели

Для распределения нагрузки между всеми доменными именами корневых серверов рекурсивным распознавателям **следует** случайным образом выбирать цель для подготовительного запроса из списка адресов. Рекурсивный распознаватель может выбирать адреса IPv4 или IPv6 основываясь на своих сведениях о поддержке системой адекватной связности с использованием конкретного типа адреса. Отметим, что этот рекомендуемый метод не является единственным способом выбора из списка в конфигурации рекурсивного распознавателя. Два других распространённых метода включают выбор первого адреса из списка и запоминание адреса, предоставлявшего ранее самый быстрый отклик, для последующего использования. Однако для подготовительных запросов **следует** использовать случайный выбор.

3.3. DNSSEC с подготовительными запросами

Распознаватель **может** установить бит DNSSEC OK (DO). На момент публикации не было большого смысла проверять DNSSEC для подготовительного запроса. В настоящее время все имена корневых серверов заканчиваются на root-servers.net, а RRset AAAA и A размещаются в зоне root-servers.net. Все корневые серверы являются полномочными для этой зоны, что позволяет включать в подготовительные отклики RRset AAAA и A подходящего корневого сервера имён. Однако зона root-servers.net в настоящее время не подписывается, поэтому RRset невозможно проверить.

Атака с участием человека (man-in-the-middle) на подготовительный запрос может направить распознаватель на поддельный (rogue) корневой сервер. Однако проверяющий распознаватель не будет воспринимать отклики от поддельных серверов имён, если они отличаются от реальных откликов, поскольку у распознавателя имеется привязка доверия для корня и отклики от корня подписываются. Таким образом, такая атака на подготовительный запрос будет приводить лишь к отказу в обслуживании, но не к восприятию распознавателем поддельных откликов.

Если зона root-servers.net впоследствии будет подписываться или корневые серверы будут указываться в иной зоне, которая подписана, проверка DNSSEC для подготовительных запросов может стать полезной.

4. Отклики при подготовке

Подготовительный запрос является обычным запросом DNS и корневой сервер не может отличить его от других запросов для корневого NS RRset. Таким образом, отклик корневого сервера тоже является обычным откликом DNS.

4.1. Ожидаемые свойства откликов

Предполагается, что подготовительный отклик будет иметь RCODE NOERROR, а бит полномочного отклика (Authoritative Answer или AA) будет установлен (1). Также предполагается наличие NS RRset в разделе Answer (поскольку NS RRset исходит из корневой зоны) и пустой раздел Authority (NS RRset уже имеется в разделе Answer). В отклике будет раздел Additional с RRset A и/или AAAA для корневых серверов имён, указанных NS RRset.

Программам распознавателей **следует** считать подготовительный отклик обычным откликом DNS и использовать как любые другие данные, приходящие в его кэш. Программам распознавателя **не следует** ожидать в точности 13 записей NS RR, поскольку некоторые корневые серверы возвращают меньшее число записей.

4.2. Полнота откликов

В настоящее время имеется 13 корневых серверов, каждый из которых имеет 1 адрес IPv4 и 1 адрес IPv6. Даже без учета NS RRset, суммарный размер записей A и AAAA превышает 512-октетный предел, заданный в [RFC1035].

При получении отклика, где в разделе Additional отсутствуют адреса некоторых корневых серверов, повторный подготовительный запрос не поможет, если отвечающие корневые серверы используют фиксированный порядок адресов в разделе Additional. Вместо этого рекурсивному распознавателю нужно делать прямые запросы для RRset A и AAAA оставшихся имён. В настоящее время эти RRset полномочно представляются корневыми серверами имён.

5. Вопросы безопасности

Подделку откликов на подготовительные запросы можно использовать для перенаправления всех запросов от жертвы (рекурсивный распознаватель) на один или несколько серверов злоумышленника. Пока отклики на подготовительные запросы не защищены с помощью DNSSEC, нет надёжного способа защиты от таких перенаправлений.

Злоумышленник на пути, который видит подготовительные запросы от распознавателя, может внедрить ложные отклики до того, как корневой сервер сможет передать верный ответ. Если отклик атакующего будет воспринят, это может открыть возможность давать ложные отклики на последующие запросы к распознавателю. Ложные отклики для корневых серверов боле опасны, чем, например, ложные отклики для доменов верхнего уровня Top-Level Domain или TLD), поскольку корень является верхним уровнем DNS (см. параграф 3.3).

В обоих приведённых выше сценариях проверяющий распознаватель сможет обнаружить атаку, если его цепочка запросов ведёт к подписанной зоне.

6. Взаимодействие с IANA

Этот документ не требует действий IANA.

7. Нормативные документы

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <http://www.rfc-editor.org/info/rfc1034>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <http://www.rfc-editor.org/info/rfc1035>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3226] Gudmundsson, O., "DNSSEC and IPv6 A6 aware server/resolver message size requirements", RFC 3226, DOI 10.17487/RFC3226, December 2001, <http://www.rfc-editor.org/info/rfc3226>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <http://www.rfc-editor.org/info/rfc4033>.

[RFC5452] Hubert, A. and R. van Mook, "Measures for Making DNS More Resilient against Forged Answers", RFC 5452, DOI 10.17487/RFC5452, January 2009, <http://www.rfc-editor.org/info/rfc5452>.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <http://www.rfc-editor.org/info/rfc6891>.

Благодарности

Этот документ является результатом работы группы DNSOP и создан на основе выполненных ей обзоров.

Адреса авторов

Peter Koch
DENIC eG
Kaiserstrasse 75-77
Frankfurt 60329
Germany
Phone: +49 69 27235 0
Email: pk@DENIC.DE

Matt Larson
ICANN
Email: matt.larson@icann.org

Paul Hoffman
ICANN
Email: paul.hoffman@icann.org

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru