Network Working Group
Request for Comments: 1912
Obsoletes: 1537
Category: Informational

D. Barr
The Pennsylvania State University

February 1996

Common DNS Operational and Configuration Errors

Распространённые ошибки при настройке и использовании DNS

Статус документа

Этот документ содержит информацию для сообщества Internet. Документ не задаёт каких-либо стандартов Internet. Распространение этого документа ничем не ограничивается.

Аннотация

В этом документе описаны ошибки, часто встречающиеся в работе серверов системы доменных имён (Domain Name System или DNS) и данных, содержащихся на этих серверах. Предпринята попытка обобщить текущие требования Internet, а также общепринятую практику настройки и эксплуатации DNS. Документ также пытается обобщить и расширить рассмотрение вопросов, поднятых в [RFC 1537].

1. Введение

Запуск сервера имён - задача не из простых. Многое может пойти не так и нужно принимать множество решений о том, какие данные помещать в DNS и как настраивать серверы. В этом документе предпринята попытка рассмотреть многие распространённые ошибки и подводные камни, связанные с данными DNS, а также с работой серверов имён. Обсуждаются также другие относящиеся в этой теме вопросы, такие как ошибки в серверах и распознавателях и некоторые политические вопросы, связанные с работой DNS в Internet.

2. Данные DNS

В этом разделе обсуждаются проблемы, которые обычно возникают при работе с данными DNS, которые размещаются в файлах, загружаемых в память серверами имён.

2.1 Несогласованность, отсутствие или некорректность данных

Каждому хосту, доступному в Internet, следует иметь имя. Последствия этого становятся всё более очевидными. Многие службы, доступные в Internet, не будут взаимодействовать с вами без корректной регистрации в DNS.

Следует убедиться в согласованности записей PTR и A. Для каждого адреса IP следует включать соответствующую запись PTR в домене in-addr.arpa. Если хост является многодомным (более одного адреса IP) нужно убедиться в наличии для всех адресов IP (а не только для первого) соответствующих записей PTR. Несоответствие записей PTR и A может приводить к потере услуг Internet как при отсутствии регистрации в DNS. Записи PTR должны указывать действительные записи A, а не псевдонимы, заданные CNAME. Настоятельно рекомендуется использовать программы, которые автоматически проверяют соответствие или генерируют данные DNS на основе базы данных, автоматически обеспечивающей согласованность.

Доменные имена в DNS состоят из меток (label), разделённых символом точки (.). В системе DNS применяются очень мягкие правила в части допустимых в доменных именах символов. Однако при использовании доменного имени для хоста следует соблюдать правила, заданные для имён хостов. Если же имя используется для электронной почты, оно должно следовать правилам для почтовых адресов.

В метках разрешается использовать любые буквы, цифры и символ дефиса (-) в кодировке ASCII. Метка не может включать только цифры, но может начинаться с цифр (например, 3com.com). Метки должны начинаться и завершаться буквой или цифрой (см. [RFC 1035] и [RFC 1123]). В [RFC 1035] указано, что метки могут начинаться только с буквы, и имеются сведения, что некоторые старые хосты продолжают следовать этому ограничению, хотя оно было смягчено в [RFC 1123]. Отметим, что в Internet имеются хосты, нарушающие это правило (411.org, 1776.com). Включение в метки символов подчёркивания (_) разрешено в [RFC 1033], но [RFC 1033] имеет статус информационного документа и не задаёт стандарта. Имеется по меньшей мере одна популярная реализация TCP/IP, отказывающаяся взаимодействовать с хостами, содержащими в имени символ подчёркивания. Следует отметить, что формулировка [RFC 1035] указывает эти правила как добровольные, которые введены лишь для тех, кто хочет минимизировать проблемы. Правила для имён хостов Internet применяются и к хостам и адресам, используемым в SMTP (см. RFC 821).

Если доменное имя применяется для почты (не SMTP), оно должно следовать правилам для почты [RFC 822], которые мягче указанных выше правил. Метки для почты могут содержать любые символы ASCII за исключением «специальных», управляющих и пробельных. Специальными считаются символы, используемые при синтаксическом анализе адресов, - «()<>@,;:\".[]». Символ ! Не был указан в [RFC 822], однако его не следует применять в адресах во избежание конфликтов с почтой UUCP, определённой в RFC 976. Поскольку сегодня почти все имена, используемые для почты в Internet, являются также именами хостов, редко можно встретить адреса, использующие смягчённые стандарты, но почтовым программам следует быть либеральными и отказоустойчивыми, чтобы воспринимать адреса.

Следует также соблюдать осторожность, чтобы у хостов не было адресов, которые разрешены альтернативным синтаксисом вызовов библиотеки inet_ntoa(). Например, имя 0xe является действительным, однако при вводе команды telnet 0xe будет предприниматься попытка соединения с адресом IP 0.0.0.14. Имеются сведения, что существуют процедуры inet_ntoa(), воспринимающие адреса, подобные x400, как адреса IP.

В некоторых операционных системах имеются ограничения на размер собственного (без домена) имени хоста. Хотя для DNS это не имеет особого значения, следует помнить об ограничениях операционной системы при выборе имени хоста.

Следует помнить, что у многих записей о ресурсах (RR) имеется один или несколько аргументов. HINFO требует двух аргументов, как и RP. Если нужно число аргументов не предоставлено, серверы могут иногда возвращать «мусор» для опущенных полей. Если в данные нужно включить пробельные символы, соответствующая строка должна помещаться в кавычки.

2.2 Записи SOA

В записи SOA каждой зоны нужно указать адрес электронной почты ответственного за поддержку зоны DNS на сайте (обычно его называют hostmaster) с заменой символа @ в адресе точкой (.). Не следует указывать символ @ в этом адресе. Если в локальной части адреса уже имеется точка (например, John.Smith@widget.xx), нужно поместить перед этой точкой символ обратной дробной черты (\), например, John\.Smith.widget.xx. Можно (и предпочтительно) указать базовое имя hostmaster и использовать псевдоним для перенаправления почты в нужный ящик. Некоторые программы используют это имя для автоматического создания адреса при контактах с зоной и будут сталкиваться с проблемами, если поле имеет некорректный формат. Очень важно, чтобы почта с таким адресом попадала одному или нескольким людям, поскольку адрес часто используется для разных целей - от уведомлений о некорректных данных в зоне DNS до информации об инцидентах безопасности.

Хотя некоторые версии BIND позволяют использовать в порядковом номере десятичные дроби, делать так не следует. Дробный порядковый номер всё равно преобразуется в 32-битовое целое число (n.m преобразуется в $n*10^{(3+int(0.9+log10(m)))} + m$), что приводит к неожиданным результатам. Например, дробный порядковый номер (возможно созданный автоматически SCCS) может быть инкрементирован так, что об будет численно больше, но после указанного выше преобразования значение станет **меньше** прежнего. Дробные порядковые номера были официально отменены в недавних версиях BIND. Рекомендуется использовать номера вида YYYYMMDDnn (YYYY=год, MM=месяц, DD=число, nn=номер выпуска), которые не приведут к переполнению до 4294 г.

Временные параметры в записи SOA указываются в секундах, как описано ниже.

Refresh - обновление

Указывает, как часто вторичный сервер будет опрашивать вторичный на предмет увеличения порядкового номера зоны (чтобы знать о необходимости запросить новую копию данных зоны). Значение определяет, как долго вторичные серверы могут сохранять устаревшие данные. Можно установить небольшое значение (от 20 минут до 2 часов), если некоторый рост расхода пропускной способности не беспокоит, или более долгое время (от 2 до 12 часов), если соединение с Internet медленное или организуется по запросу. В свежих версиях BIND (4.9.3) имеется дополнительный код для уведомления вторичных серверов об изменении данных, что позволяет установить для этого TTL достаточно большое значение (сутки и более).

Retry - интервал повтора

Если вторичный сервер не смог связаться с первичным при последнем обновлении, он будет ждать в течение указанного этим значением времени перед повторением попытки. Это значение не так важно, как другие, если только вторичный сервер не подключён к значительно удалённой от первичного сети или первичный сервер не подвержен перебоям в работе. Обычно это значение составляет какую то часть интервала обновления (refresh).

Expire - срок действия данных зоны

Как долго вторичный сервер будет считать свою копию данных зоны действительной при невозможности контакта с первичным сервером. Это значение следует делать больше, чем продолжительность типичного серьёзного сбоя в работе и больше значений minimum и retry, чтобы предотвратить завершение срока действия данных зоны до появления возможности обновить их. По завершении срока действия вторичный сервер будет продолжать попытки связаться с основным, но прекратит обслуживание имён для зоны. Рекомендуется значение от 2 до 4 недель.

Minimum

Принятый по умолчанию срок действия (time-to-live или TTL) записей о ресурсах - время, в течение которого они будут сохраняться в кэше других серверов имён (в [RFC 1035] это значение определено как минимальное, но серверы, похоже, используют его в качестве заданного по умолчанию). Это, безусловно, важный параметр и для него следует устанавливать значение, соответствующее частоте обновлений на сервере имён. Если планируются серьёзные изменения на сервере, имеет смысл временно установить небольшое значение, затем дождаться истечения прежнего значения minimum, внести изменения, проверить их корректность и вернуть прежнее значение минимума. Обычно устанавливается значение от 1 до 5 суток. Следует помнить, что этот параметр может переопределяться для отдельных записей о ресурсах.

Из приведённого описания видно, что значения таймеров могут меняться в широких пределах. В популярных документах, например, [RFC 1033], рекомендуется указывать минимальное значение TTL в 1 сутки, но сейчас это значение считается слишком малым, если только зона не обновляется достаточно часто. После стабилизации DNS рекомендуется устанавливать значение не менее 3 суток. Рекомендуется также индивидуально устанавливать значения TTL (1-2 недели) для RR, на которые часто ссылаются, но при этом записи меняются редко. Хорошим примером являются записи MX, A и PTR для почтовых хостов, записи NS для зоны и записи A для серверов имён.

2.3 Склеивающие записи A

Склеивающими называют записи A, связанные с записями NS, для начальной загрузки (bootstrapping) информации в сервер имён. Например,

```
podunk.xx.      in      ns      ns1.podunk.xx.
                in      ns      ns2.podunk.xx.
ns1.podunk.xx.  in      a       1.2.3.4
ns2.podunk.xx.  in      a       1.2.3.5
```

Указанные в примере записи A называются склеивающими (Glue record).

Склеивающие записи требуются только в файлах прямых зон для серверов имён, размещённых в субдоменах текущей зоны, которая делегируется. В файл зоны in-addr.arpa не следует включать какие-либо записи A (если не применяется представление масок подсетей в стиле RFC 1101).

Если сервер имён является многодомным (более одного адреса IP), в склеивающих записях требуется указать все его адреса, чтобы избежать несогласованности кэшей из-за разных значений TTL, в результате чего некоторые поисковые запросы будут находить не все адреса сервера имён.

Некоторые люди имеют дурную привычку добавлять склеивающую запись для каждой записи NS «просто для уверенности». Наличие дублирующихся склеивающих записей в зоне лишь усложнит задачу при смене или удалении IP-адреса сервера имён. В результате придётся потратить время, чтобы понять, почему кто-то видит старый адрес IP из-за того, что не удалена склеивающая запись в каком-то другом файле. Новые версии BIND игнорируют лишние склеивающие записи в локальных файлах зон.

В старых версиях BIND (4.8.3 и ниже) имеется проблема вставки лишних склеивающих записей при передаче данных зоны вторичным серверам. Если одна из таких записей ошибочна, ошибка может распространиться на другие серверы. Если два сервера являются вторичными для других зон друг друга, один из них может постоянно передавать другому старые склеивающие записи. Единственным способом избавиться от старых данных является отключение обоих серверов, удаление старых резервных копий файлов и повторный запуск серверов. В этих же версиях сервера имеется тенденция заражения фиктивными данными, найденными на других (не вторичных) серверах имён (например, данными корневой зоны).

2.4 Записи CNAME

Записям CNAME не разрешается сосуществовать с какими-либо иными данными. Иными словами, если suzy.podunk.xx является псевдонимом sue.podunk.xx, нельзя включать для suzy.podunk.xx[1] запись MX или A и даже TXT. Ниже приведён пример недопустимого сочетания записей CNAME и NS.

```
podunk.xx.      IN      NS      ns1
                IN      NS      ns2
                IN      CNAME   mary
mary            IN      A       1.2.3.4
```

Неопытные администраторы часто пытаются использовать CNAME как очевидный способ разрешить доменному имени быть хостом. Однако серверы DNS, такие как BIND, будут видеть запись CNAME и отвергать добавление любых других ресурсов для этого имени. Поскольку сосуществование других записей с CNAME не допускается, записи NS будут проигнорированы и все хосты домена podunk.xx также будут игнорироваться!

Если нужно связать домен с хостом, это можно сделать, как показано ниже.

```
podunk.xx.      IN      NS      ns1
                IN      NS      ns2
                IN      A       1.2.3.4
mary            IN      A       1.2.3.4
```

Не следует перебарщивать с CNAME. Эти записи пригодны для переименования хостов, но от них следует избавляться, информируя об этом пользователей. Однако записи CNAME полезны (и поощряются) для обобщённых имён серверов - ftp для сервера FTP, www для Web-сервера, gopher для сервера Gopher, news для новостей Usenet и т. п.

Следует не удалять записи CNAME связанные с удаляемым хостом, для которого они служат псевдонимом. Забытые записи CNAME ведут к ненужному расходу ресурсов.

Не следует применять CNAME в сочетании с иными RR, указывающими другие имена, такими как MX, CNAME, PTR, NS (PTR является исключением в случае бесклассового делегирования in-addr). Ниже приведён пример недопустимого сочетания.

```
podunk.xx.      IN      MX      mailhost
mailhost        IN      CNAME   mary
mary            IN      A       1.2.3.4
```

В параграфе 3.6.2 [RFC 1034] сказано, что не следует так делать, а в [RFC 974] явно указано, что записи MX не должны указывать на псевдоним, заданный CNAME. Такое сочетание приводит к излишней опосредованности при доступе к данным, а серверам и распознавателям DNS потребуются дополнительные действия для получения ответа. Желаемого результата можно с использованием препроцессора (например, m4) в файле хостов.

Наличие цепочек, таких как запись CNAME, указывающая CNAME, может упростить администрирование, однако некоторые распознаватели не могут корректно распознавать петли. В результате некоторые хосты не смогут распознать такие имена.

Наличие записей NS, указывающих на CNAME, является дурным тоном и может вызывать конфликты с современными серверами BIND. Фактически, текущие реализации BIND будут игнорировать такие записи, что может приводить к неработающему делегированию. В BIND имеются некоторые проверки безопасности для предотвращения ложных записей DNS NS. Известно, что старые версии BIND попадают в петлю запросов, пытаясь получить адрес сервера имён по псевдониму, что ведёт к передаче непрерывного потока запросов DNS.

2.5 Записи MX

Хорошей идеей является предоставление каждому хосту записи MX, даже если она указывает сам этот хост! Некоторые почтовые программы (mailer) кэшируют записи MX, но перед отправкой почты MX всегда следует проверять. Если у сайта нет MX, любая часть почты будет приводить к дополнительному запросу распознавателя, поскольку отклик на запрос MX часто будет включать IP-адреса хостов MX. От почтовых программ Internet SMTP [RFC 1123] требует поддерживать механизм MX.

Следует размещать записи MX даже на хостах, не предназначенных для примем или передачи электронной почты. Если на таком хосте возникнут проблемы безопасности, некоторые люди могут направлять сообщения по адресу postmaster или root на сайте, проверяя сначала, является ли он «реальным» хостом или просто терминалом или ПК, не предназначенным для восприятия электронной почты. Наличие записи MX позволит перенаправлять почту реальному человеку. Иначе почта может просто часами находиться в очереди, пока отправитель не отменит попытку передать её.

[1] В оригинале ошибочно указано suzy.podunk.edu, см. https://www.rfc-editor.org/errata/eid5381. *Прим. перев.*

Не следует забывать, что при добавлении записи MX нужно указать целевой почтовой программе, должна ли она считать первый хост «локальным» (например, флаг Cw в sendmail).

При добавлении записи MX, указывающей внешний хост (например, для резервирования маршрутизации почты), нужно сначала получить разрешение от соответствующего сайта. Иначе этот сайт может принять меры (например, отбрасывать почту или пожаловаться администратору DNS родительского домена или провайдеру сетевых услуг).

2.6 Другие записи о ресурсах

2.6.1 WKS

Записи WKS отменены в [RFC 1123]. Они не выполняют каких-либо полезных функций, кроме внутренних функций для машин LISP. Не следует использовать эти записи.

2.6.2 HINFO

Некоторые считают записи HINFO вызывающими проблемы безопасности, поскольку широковещательная передача сведений об оборудовании и операционной системе может приводить к систематическим атакам на известные бреши в защите. При использовании таких записей следует внимательно отслеживать сведения о безопасности оборудования. Тем не менее, эти записи могут быть полезны. Следует помнить, что HINFO требует двух аргументов, указывающих тип оборудования и информационную систему.

Иногда записи HINFO ошибочно применяют для предоставления других сведений. Эти записи предназначены для указания информации о самой машине, а для иных сведений о хосте в системе DNS следует использовать записи TXT.

2.6.3 TXT

Записи TXT не имеют конкретного определения и в них можно помещать практически любые сведения. Некоторые используют такие записи для базового описания хоста, другие - для определённых сведений, таких как местоположение, основной пользователь или даже номер телефона.

2.6.4 RP

Записи RP введены сравнительно недавно и применяются для указания адреса электронной почты (см. параграф 2.2) ответственного лица (Responsible Person) на хосте и имени записи TXT, содержащей дополнительные сведения [RFC 1183].

2.7 Шаблонные записи

Шаблоны MX полезны в основном для сайтов, подключённых не по протоколу IP. Распространённой ошибкой является представление, что шаблон MX применяется ко всем хостам зоны. На деле этот шаблон применяется лишь к именам в этой зоны, которые совсем не указаны в DNS. Например,

```
podunk.xx.      IN      NS      ns1
                IN      NS      ns2
mary            IN      A       1.2.3.4
*.podunk.xx.    IN      MX      5 sue
```

Почта для mary.podunk.xx будет отправляться напрямую этому хосту, а почта для jane.podunk.xx и других хостов, не указанных в примере, будет передаваться по записи MX. Для большинства сайтов Internet шаблонные записи MX бесполезны. Нужно устанавливать явные записи MX на каждом хосте.

Шаблонные записи MX могут приводить к успешному выполнению некоторых операций, которые на деле следует отклонять. Рассмотрим случай, когда кто-то в домене widget.com пытается отправить почту по адресу joe@larry. Если хоста в реальности нет, почту следует незамедлительно возвратить. Но при поиске адреса домен распознаётся как larry.widget.com и при наличии шаблонной записи MX адрес будет действительным в DNS. Возможны просто опечатки в хостовой части почтового адреса и сообщение будет маршрутизироваться на почтовый хост, который возвратит странное сообщение об ошибке, например, I refuse to talk to myself (Я не буду разговаривать сам с собой) или Local configuration error (Ошибка в локальной конфигурации).

Шаблонные записи MX полезны при наличии большого числа хостов, не подключённых напрямую к Internet (например, размещённых за межсетевым экраном), и административных или политических сложностей с указанием отдельных записей MX для каждого хоста или для «сокрытия» всех почтовых адресов за одним или несколькими доменными именами. В этом случае нужно разделить систему DNS на две части - внутреннюю и наружную. Внешняя часть будет включать лишь несколько хостов и явные записи MX, а также одну или несколько шаблонных записей MX для внутренних доменов. Внутренняя часть DNS будет полной с явными записями MX и без шаблонов.

Возможны шаблонные записи A и CNAME, но это запутывает пользователей и может стать кошмаром при необдуманном применении. Это может приводить (опять же в результате поиска) к тому, что любые обращения по протоколу telnet или ftp из домена к неизвестному хосту будут направляться на один адрес. Одна шаблонная запись CNAME (*.edu.com) вызвала нарушение обслуживания в масштабе всей сети Internet и возможные проблемы безопасности из-за неожиданных взаимодействий с поиском. Это было быстро исправлено и даже был выпущен специальный документ [RFC 1535] с описанием проблемы.

2.8 Ошибки полномочий и делегирования (записи NS)

Для каждого домена требуется не менее двух серверов имён, хотя предпочтительно иметь больше. Вторичные серверы следует размещать вне своей сети. Если вторичный сервер не находится под вашим контролем, следует периодически проверять его, убеждаясь в получении текущих данных вашей зоны. Запросы к «чужому» серверу имён для ваших хостов всегда должны возвращать «полномочные» (authoritative) отклики. Если это не так, возникает «неудачное делегирование» (lame delegation). Неудачное делегирование имеет место, когда серверу имён переданы полномочия предоставлять службу имён для зоны (через записи NS), но тот не обслуживает имена для этой зоны (обычно из-за того, что он не настроен как первичный или вторичный сервер этой зоны). «Классический» пример неудачного делегирования показан ниже.

```
          podunk.xx.      IN      NS      ns1.podunk.xx.
                          IN      NS      ns0.widget.com.
```

Домен podunk.xx создан недавно и для зоны организован сервер имён ns1.podunk.xx. Сделано ещё не все и не проверена настройка ns0.widget.com в качестве вторичного сервера, на котором ещё нет сведений о домене podunk.xx, хотя DNS говорит о том, что домен имеется. В зависимости от используемого сервера имён могут возникать различные ситуации и в худшем случае имена хостов не будут распознаваться, а электронная почта будет возвращена.

Иногда сервер имён может переноситься на другой хост или исключаться из списка вторичных. К сожалению, из-за кэширования записей NS, многие сайты будут считать хост вторичным сервером после того, как он перестанет поддерживать службу имён. Для предотвращения неработающей передачи полномочий, пока кэш ещё не устарел, следует продолжать поддержку службы имён на прежнем сервере в течение срока, равного большему из значений «минимум плюс время обновления» для данной зоны и её родительской зоны (см. параграф 2.2)

При удалении или изменении первичного или вторичного сервера требуются достаточно большие человеческие усилия для координации всех вовлечённых сторон (самого сайта, его родительского домена и сайта, где размещён вторичный сервер). При переносе первичного сервера нужно убедиться, что на всех вторичных серверах обновлён файл named.boot и сервер имён перезапущен. При переносе вторичного сервера нужно убедиться, что адресные записи на уровне первичного сервера и родительской зоны изменены.

Отмечено, что некоторые сайты выбирают популярные отделённые серверы имён (например, ns.uu.net) и просто добавляют их в список своих записей NS, полагая, что эти серверы будут волшебным образом выполнять для них дополнительное распознавание имён. Это ещё хуже, чем ущербное делегирование, поскольку добавляет трафик для загруженного и без того сервера имён. При обнаружении таких ситуаций следует обращаться к администраторам таких серверов. Для обнаружения и активного поиска таких сервером имеются различные инструменты. Список таких программ приводится в дистрибутиве BIND.

Следует убедиться, что в родительском домене указаны те же записи NS, что и в зоне, не забывая и зоны in-addr.arpa. Не следует указывать слишком много записей (рекомендуется использовать не более 7), поскольку это усложняет поддержку и реально нужно лишь для очень популярных серверов верхнего уровня и корневых зон. Кроме того, слишком большое число записей может привести к превышению 512-байтового ограничения для размера пакетов UDP с откликами на запросы NS. Такое превышение приведёт к использованию распознавателями протокола TCP, что существенно увеличит нагрузку на сервер имён.

При выборе географического местоположения для вторичных серверов имён важно минимизировать задержки и повысить надёжность. При этом следует учитывать топологию сетей. Например, если сайт подключён к медленному международному или местному каналу, следует рассмотреть возможность размещения вторичного сервера на другой стороне такого канала для снижения средней задержки. Для получения дополнительных сведений о доступных вторичных серверах имён следует обращаться к провайдеру Internet или администратору родительского домена.

3. Работа BIND

В этом разделе рассматриваются типичные проблемы, возникающие при работе серверов имён (в частности, BIND). Для постоянной доступности данных требуется не только их корректность, как описано выше, но и правильная работа сервера имён.

3.1 Порядковые номера

С каждой зоной связан порядковый номер, служащий для отслеживания владельца самых актуальных данных. Если номер в первичной зоне больше, чем во вторичной, последняя запрашивает копию новых данных зоны (особый случай описан ниже).

При изменении данных следует менять порядковый номер зоны. Если этого не сделать, вторичные серверы не будут забирать обновлённые данные зоны. Хорошей идеей является автоматическое увеличение порядкового номера программой.

Если по ошибке порядковый номер увеличен слишком сильно и нужно уменьшить значение, следует соблюдать указанные ниже процедуры.

К «ошибочному» порядковому номеру следует прибавить 2147483647 и, если результат превышает 4294967296, вычесть из него 4294967296. Полученное значение следует указать в зоне и выждать 2 интервала обновления (refresh), чтобы зона обновилась на всех серверах.

Эти действия следует повторять, пока полученный в результате порядковый номер не станет меньше целевого.

Порядковый номер следует увеличить до целевого значения.

Эта процедура не сработает, если на каком-либо из вторичных серверов используется старая версия BIND (4.8.3 или ниже). В этом случае следует обратиться к администратору вторичного сервера для удаления файла зоны и перезапуска сервера. При редактировании порядковых номеров следует соблюдать осторожность - администраторы DNS не любят останавливать и повторно запускать серверы имён, поскольку при этом теряются кэшированные данные.

3.2 Рекомендации для файлов зон

Ниже приведено несколько советов по структурированию файлов зон, которые помогут заметить и предотвратить ошибки. Следует сохранять общий стиль записей в файлах DNS. Например, для $ORIGIN podunk.xx. Не следует пытаться внести записи вида

```
          mary            IN      A       1.2.3.1
          sue.podunk.xx.  IN      A       1.2.3.2
```

или

```
          bobbi           IN      A       1.2.3.2
                          IN      MX      mary.podunk.xx.
```

Следует везде использовать полные доменные имена (Fully Qualified Domain Name или FQDN) или только неполные (относительные), не допуская их смешивания. Возможно также использование, например, только полных доменных имён в правой части и неполных - слева. В любом случае нужна согласованность.

Для разделения полей следует использовать символы табуляции, пытаясь сохранить выравнивание по колонкам. Это позволяет легче заметить пропуски полей (отметим, что некоторые поля, такие как IN, наследуются от предыдущей записи и при определённых обстоятельствах могут быть пропущены). Отметим, что при указании нескольких записей для одного хоста не требуется повторять имя этого хоста. В файле следует сохранять все записи, связанные с хостом. Это упростит работу в будущем, если хост нужно будет переименовать или удалить.

Следует помнить об $ORIGIN. Если в конце FQDN отсутствует точка (.), имя не будет воспринято как полное и сервер имён добавить $ORIGIN в конце этого имени. Следует обращать пристальное внимание на точку в конце имён, особенно в файлах зоны in-addr.arpa, где они нужны больше всего.

Следует внимательно относиться к синтаксису записей SOA и WKS (записи с круглыми скобками), поскольку BIND не обеспечивает достаточной гибкости при анализе таких записей. Дополнительные сведения приведены в документации BIND.

3.3 Проверка данных

После добавления или изменения данных следует проверить их, запросив распознаватель с помощью команды dig (или иного инструмента DNS, многие из которых включены в дистрибутив BIND). Несколько секунд, затраченных на проверку, могут избавить от затраты времени на поиск и устранение проблем, потери почтовых сообщений и т. п. При наличии ошибок в данных DNS или загрузочном файле демон named укажет их через syslog.

Настоятельно рекомендуется автоматизировать проверку с помощью программы контроля файлов данных перед их загрузкой на сервер имён или контроля уже загруженных данных. Некоторые программы для таких проверок включены в дистрибутив BIND.

4. Прочие вопросы

4.1 Настройка загрузочного файла

Указанные ниже зоны следует всегда включать в конфигурацию сервера имён.

```
primary         localhost               localhost
primary         0.0.127.in-addr.arpa    127.0
primary         255.in-addr.arpa        255
primary         0.in-addr.arpa          0
```

Эти зоны создаются для обслуживания «специальных» адресов или для исключения передачи корневым серверам случайных запросов для широковещательных или локальных адресов. Все эти файлы содержат записи NS и SOA, как и другие файлы поддерживаемых зон, но таймеры SOA в них могут быть очень большими, поскольку данные никогда не меняются.

Адрес localhost является «специальным» и всегда служит для указания локального хоста. Он указывается в виде

```
localhost.      IN      A       127.0.0.1
```

В файле 127.0 следует указывать строку

```
1    PTR     localhost.
```

Было много дискуссий по поводу добавления в этот файл локального домена. В итоге пришли к выводу, что лучшим решением будет указание localhost. Причины этого указаны ниже.

Имя localhost применяется самостоятельно и предполагается его использование в некоторых системах.

Трансляция 127.0.0.1 в localhost.dom.ain может вынудить некоторые программы подключаться к интерфейсу, когда они этого не хотели, поскольку localhost не совпадает с localhost.dom.ain.

В файлы 255 и 0 не следует включать какие-либо данные, кроме записей NS и SOA.

Отметим, что в будущих версиях BIND все или часть указанных выше данных будут включаться автоматически без дополнительной настройки.

4.2 Другие ошибки распознавателей и серверов

В очень старых версиях распознавателя DNS была ошибка, из-за которой запросы для имён, похожих на адреса IP, не выполнялись, поскольку пользователь представлял IP-адрес, а программа не знала, что его не нужно распознавать. Эта проблема была устранена, но иногда ещё проявляется. Это важно, поскольку такие запросы будут направляться напрямую к корневым серверам имён, увеличивая и без того высокую нагрузку DNS на них.

Хотя работа вторичного сервера имён с другим вторичным сервером возможна, делать этого не рекомендуется, если только этого не требует топология сети. Известны случаи, когда это приводило к таким проблемам, как фиктивные значения TTL. Хотя это может быть связано со старыми и ошибочными реализациями DNS, не следует делать цепочки вторичных серверов, поскольку они создают дополнительные зависимости в плане надёжности, а также вносят дополнительную задержку при обновлении данных зон.

4.3 Проблемы серверов

DNS работает в основном через сообщения UDP (User Datagram Protocol). Некоторые операционные системы UNIX в стремлении сэкономить такты CPU отключают проверку контрольных сумм UDP. Сравнительные достоинства такого подхода обсуждаются давно. Однако с ростом скорости CPU важность этого для производительности снижается. Настоятельно рекомендуется включить проверку контрольных сумм UDP, чтобы избежать повреждения данных не только для DNS, но и для других служб на основе UDP (например, NFS). Включение проверки контрольных сумм UDP описано в документации ОС.

Литература

[RFC 974] Partridge, C., "Mail routing and the domain system", STD 14, RFC 974, CSNET CIC BBN Laboratories Inc, January 1986.

[RFC 1033] Lottor, M, "Domain Administrators Operations Guide", RFC 1033, USC/Information Sciences Institute, November 1987.

[RFC 1034] Mockapetris, P., "Domain Names - Concepts and Facilities", STD 13, RFC 1034, USC/Information Sciences Institute, November 1987.

[RFC 1035] Mockapetris, P., "Domain Names - Implementation and Specification", STD 13, RFC 1035, USC/Information Sciences Institute, November 1987.

[RFC 1123] Braden, R., "Requirements for Internet Hosts -- Application and Support", STD 3, RFC 1123, IETF, October 1989.

[RFC 1178] Libes, D., "Choosing a Name for Your Computer", FYI 5, RFC 1178, Integrated Systems Group/NIST, August 1990.

[RFC 1183] Ullman, R., Mockapetris, P., Mamakos, L, and C. Everhart, "New DNS RR Definitions", RFC 1183, October 1990.

[RFC 1535] Gavron, E., "A Security Problem and Proposed Correction With Widely Deployed DNS Software", RFC 1535, ACES Research Inc., October 1993.

[RFC 1536] Kumar, A., Postel, J., Neuman, C., Danzig, P., and S. Miller, "Common DNS Implementation Errors and Suggested Fixes", RFC 1536, USC/Information Sciences Institute, USC, October 1993.

[RFC 1537] Beertema, P., "Common DNS Data File Configuration Errors", RFC 1537, CWI, October 1993.

[RFC 1713] A. Romao, "Tools for DNS debugging", RFC 1713, FCCN, November 1994.

[BOG] Vixie, P, et. al., "Name Server Operations Guide for BIND", Vixie Enterprises, July 1994.

5. Вопросы безопасности

Вопросы безопасности не обсуждаются в этом документе

6. Адрес автора

David Barr
The Pennsylvania State University
Department of Mathematics
334 Whitmore Building
University Park, PA 16802
Voice: +1 814 863 7374
Fax: +1 814 863-8311
EMail: barr@math.psu.edu

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru