**Министерство образования и науки Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ**

**УНИВЕРСИТЕТ ИМ. Р. Е. АЛЕКСЕЕВА»**

**(НГТУ)**

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

**Методические указания**

**по выполнению лабораторных работ**

**по дисциплине «**Администрирование сетевых операционных систем**»**

Направление подготовки

09.03.01(230100) «Информатика и вычислительная техника»

код и наименование направления подготовки

Профиль подготовки

«Автоматизированные системы обработки информации и управления»

наименование профиля подготовки

Уровень высшего образования

Бакалавр

Форма обучения

Очная, очно-заочная

Нижний Новгород

2015

Разработчик(и)/составитель(и) методических указаний по выполнению

лабораторных работ по дисциплине «Администрирование сетевых операционных систем»:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_к.т.н., Ляхманов Д.А.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ученое звание, степень, фамилия, инициалы

Кафедра \_\_«Информатика и системы управления» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

наименование кафедры

Дата, подпись \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Методические указания по выполнению лабораторных работ по дисциплине

«Администрирование сетевых операционных систем» рассмотрены на заседании кафедры «Информатика и системы управления»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

наименование кафедры

Протокол № \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ от « \_\_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_\_ 20 \_\_\_\_ г.

Заведующий кафедрой ИСУ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ д.т.н., профессор Э.С.Соколова

ученое звание, степень фамилия, имя, отчество

Дата, подпись \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Методические указания по выполнению лабораторных работ по дисциплине

«Администрирование сетевых операционных систем» согласованы с председателем методической комиссии по профилю подготовки, специальности (или председателем предметной комиссии) \_\_\_зав. кафедрой ИСУ\_\_\_\_\_\_

должность

\_\_\_\_\_\_\_\_\_ д.т.н., профессор Э.С.Соколова \_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ученое звание, степень, фамилия, инициалы дата, подпись

Методические указания по выполнению лабораторных работ по дисциплине

«Администрирование сетевых операционных систем» утверждены Ученым советом учебно-научного института радиоэлектроники и информационных технологий

Протокол №\_\_ от «\_\_\_»\_\_\_\_\_\_\_2015 г.

**Цель работы**

Цель данной работы – научиться конфигурировать и тестировать службу имен для корпоративной сети на основе сервера DNS bind.

**Теоретические сведения**

Служба доменных имён DNS является важнейшей системной службой в TCP/IP сетях. Назначение DNS – преобразование символьных имен в IP адреса и наоборот, а так же предоставление дополнительной информации о хостах и группах хостов (доменах), такой как адреса почтовых обменников, публичные ключи служб и т.п. В сети Интернет служба DNS оперирует распределённой иерархической базой данных в виде дерева имен, находящейся на  множестве различных хостов. Ответственность за корректность базы данных в каждом узле дерева возлагается на администратора соответствующего домена.

В сети Интернет корнем дерева является домен “.”. Полное - абсолютное или полностью определенное, fully qualified domain name - доменное имя заканчивается точкой, обозначающей корень доменного дерева, но часто эта завершающая точка опускается. Анализ имени производится справа налево.  Самая правая секция имени характеризует страну (для каждой страны мира выделен свой домен с двух символьным именем в соответствии со стандартом ISO, например, ua – Украина, ru – Россия, uk – Англия и т.п.)  или характер организации (образовательная, коммерческая, правительственная и т.п.). Таим образом, домены верхнего уровня (Top Level Domains, TLD) - это хорошо известные всем домены, такие как org (бесприбыльная организация), com (коммерческая организация), gov (государственное учреждение), mil (военное предприятие или организация), edu (учебное заведение), int (международная организация), net (большая сеть) и т.д. Ответственность за TLD лежит на международной бесприбыльной организации Internet Corporation for Assigned Names and Numbers (ICANN, [www.icann.org](http://www.icann.org/)).

Ниже по дереву имен расположены домены второго и последующих уровней. В некоторых странах существуют домены, подобные исторически первым TLD, например, org.ua, gov.ua, edu.ua и т.п. В Украине для каждой области выделены двухбуквенные домены, такие как cn.ua – Чернигов, dp.ua – Днепропетровск, и т.д., хотя исторически сложившиеся домены, такие как kiev.ua, kharkov.ua, chernigov.ua, тоже поддерживаются. Маленький фрагмент интернетовской иерархии имен показан на рисунке 1. Число уровней реально больше, но обычно не превышает 5.

Таким образом, в службе DNS каждый сервер отвечает за определенную зону (зона ответственности) - т.е. свою часть дерева доменных имен, хранит соответствующие базы данных и отвечает на запросы. При этом вышестоящие по дереву серверы имеют информацию об адресах нижестоящих серверов, что обеспечивает связность дерева. Говорят, что вышестоящий сервер делегирует нижестоящему серверу полномочия по обслуживанию определенной зоны. Важно понимать различие между доменом и зоной. Домен - это поддерево дерева доменных имен. Зона - это часть дерева, за которую отвечает тот или иной DNS-сервер.

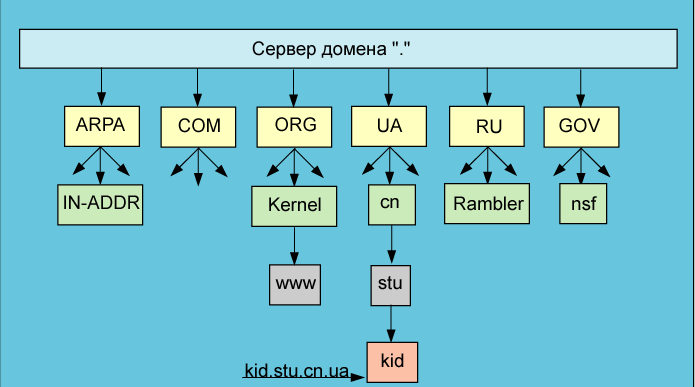


Рисунок 1 – Фрагмент иерархии имен в Интернет

За каждую зону DNS отвечает не менее двух серверов. Один из них является первичным, primary, или, в новой терминологии — master. остальные - вторичными, secondary, или slave. Первичный сервер содержит оригинальные файлы с базой данных DNS для своей зоны. Вторичные серверы получают эти данные по сети от первичного сервера и периодически запрашивают первичный сервер на предмет обновления данных. Признаком обновления данных служит увеличение серийного номера в записи SOA – см ниже. В случае, если данные на первичном сервере обновлены, вторичный сервер запрашивает "передачу зоны" ("zone transfer")- т.е. базы данных требуемой зоны. Передача зоны происходит с помощью протокола TCP, порт 53, в отличие от запросов, которые направляются на UDP/53.

Изменения в базу данных DNS могут быть внесены только на первичном сервере. С точки зрения обслуживания клиентских запросов первичные и вторичные серверы идентичны. Рекомендуется, чтобы первичный и вторичные серверы находились в разных сетях -  для увеличения надежности обработки запросов на случай, если сеть одного из серверов становится недоступной. Серверы DNS не обязаны находиться в том домене, за который они отвечают.

Вторичный сервер необязательно получает данные непосредственно с первичного сервера; источником данных может служить и другой вторичный сервер. В любом случае сервер-источник данных для данного вторичного сервера называется "главным" ("master").

У каждого домена есть свои “писаные правила”, именуемые “полиси”, в соответствии с которыми можно получить имя хоста или субдомена (чаще говорят “зоны”) в данном домене. Самым простым примером таких правил могут служить полиси домена com. Если данное имя не зарегистрировано или не находится в процессе регистрации, то по уплате определенной суммы любое физическое или юридическое лицо может получить это имя от регистратора домена com.

Процесс получения собственного домена называется “делегирование”, что, собственно, и отражает суть действий: субдомен передается в полное управление его администратору. Для получения субдомена в каком-то домене необходимо в соответствии с полиси обратится к регистратору, выполнить формальные процедуры, а так же ряд технических действий по настройке сервера DNS, которые, собственно и являются предметом данной лабораторной работы.

Различают 3 режима работы сервера DNS:

master (primary). Данный режим используется администратором зоны, файлы баз данных ведутся вручную на этом сервере. Данный сервер является абсолютным авторитетным источником информации для данной зоны;

slave (secondary). Данный режим используется по просьбе администратора зоны, которая автоматически регулярно копируется с master сервера. Данный сервер является авторитетным источником информации для данной зоны;

hint (caching). Режим кэширования всех запросов, попадающих в определённую зону, обычно “.”, т.е. кэшируются все запросы. Такой сервер обычно используется для ускорения работы с сетью.

Для каждой зоны, обслуживаемой данным сервером, может быть выбран тот или иной режим. Обычно для зоны “.” все сервера конфигурируются по типу hint, что позволяет кешировать все запросы пользовательских рабочих станций на время жизни конкретной записи DNS. Это значительно ускоряет обработку локальных запросов.

База данных DNS для каждого домена в простейшем случае представляет собой набор текстовых файлов, которые системный администратор ведет на главном сервере имен этого домена. В этих файлах содержатся директивы синтаксического анализатора ($ORIGN, $TTL) и записи о ресурсах.

**Записи ресурсов в базе данных домена**

Файл любой зоны начинается с записи Start Of Authority, SOA. Эта запись является заголовочной и содержит информацию о размещении зоны, о почтовом адресе ответственного лица и о базовых временных параметрах записей данной зоны.

Файл прямой зоны содержит стандартные записи ресурсов базы данных DNS для преобразования доменных имен хостов в данной зоне в IP-адреса, определения авторитарных DNS-серверов данной зоны, определения хостов-обработчиков почты для доменных имен в данной зоне и др.

 Файлы баз данных DNS состоят из стандартных записей ресурсов. В общем виде стандартная запись ресурса связывает данные определенного типа с некоторым именем и формируется по шаблону:

имя  [время\_жизни\_записи] IN тип\_записи данные

Именем является некоторое доменное имя (необязательно имя физически существующих хоста или домена). Если поле "имя" пусто, то значение этого поля берется из предыдущей записи. Данными может быть, например, IP-адрес хоста, если имя относится к хосту, или DNS-сервер домена, если имя относится к домену, и т.п.

Время жизни записи определяет время хранения информации этой записи в кэше запросившего запись сервера в секундах и указывается, только если оно отличается от времени жизни, определенного для всей зоны в записи SOA.

Основные типы записей:

SOA (Start Of Authority) - заголовок зоны;

NS (Name Server) - сервер DNS;

A (Address) - IP-адрес для хоста;

MX (Mail Exchanger) - почтовый обменник;

CNAME (Canonical Name) - каноническое имя, псевдоним хоста;

PTR (Pointer) - указатель по обратной зоне, фактически — имя хоста;

**Прямая зона DNS.**

Рассмотрим примеры файлов базы данных DNS. Первой рассмотрим прямую зону для приватной части корпоративной сети, домен “stu.”, файл db.stu.

$ORIGIN .  
stu 28800 IN SOA ns.stu. dnsmaster.stu. (  
2005033100 ; Serial   
28800 ; Refresh   
7200 ; Retry  
604800 ; Expire  
86400 ; Time To Live)

; authoritative name servers for zone  
28800 IN NS ns.stu.  
28800 IN NS ns1.stu.

; mail exchangers for entire zone  
28800 IN MX 10 stalker.stu.  
28800 IN MX 20 cs.stu.

$ORIGIN stu.  
; name servers glue records  
ns IN A 192.168.0.10  
ns1 IN A 192.168.0.14  
;servers  
dragon IN A 192.168.0.17  
auth IN CNAME dragon.stu.  
cs IN A 192.168.0.14  
stalker IN A 192.168.0.10  
www IN CNAME stalker.stu.  
mail IN CNAME stalker.stu.  
ftp IN CNAME stalker.stu.  
www.docs IN CNAME cs.stu.  
kid IN A 192.168.0.12  
; workstations  
ie-21-7 IN A 192.168.3.40  
ie-21-8 IN A 192.168.3.41  
ie-21-9 IN A 192.168.3.42  
vc-105-1 IN A 192.168.66.2

Первая строка – это макрос, говорящий, что все имена далее следуют непосредственно за доменом “точка”.  Таким образом, для приватной сети мы используем имена в нашем приватном дереве относительно нашего собственного корня “.”. Следует помнить, что для сервера, разрешающего одновременно и имена в корпоративной сети, и имена в интернете, имя зоны следует выбирать из 3-х символов, не совпадающих с именами TLD.

  Первой записью всегда идет SOA (Start of Authority), в которой указывается имя зоны (“stu.”, или макрос @), TTL, т.е. время жизни этой записи, дальше – ключевые слова IN (Internet records) и SOA. Далее идут параметры зоны: имя основного сервера DNS, почтовый адрес администратора зоны, однако вместо символа “@” там стоит точка, поскольку @ - это ссылка на имя зоны. Сразу за открывающейся скобкой находится серийный номер данного файла, обычно в формате ггггммддNN. Серийный номер необходимо увеличивать при каждом изменении файла, что бы ведомые сервера идентифицировали изменения и обновили файлы баз данных с главного сервера. Далее следуют стандартные времена в секундах для данной зоны:

Refresh – время, по истечении которого вторичные сервера должны обновить данные с первичных серверов (zone transfer);

Retry – время, через которое вторичные сервера должны совершить повторную попытку обновления, если предыдущая попытка не удалась;

Expire – время, через которое вторичные сервера должны выбросить запись о зоне и считать ее недоступной, если обновления не удались.

TTL – стандартное время жизни записей из данной зоны для кеширующих серверов.

Следующая группа записей является так же обязательной и указывает на авторитетные сервера имен для данной зоны - записи типа NS. Авторитетным является сервер, на котором информация соответствует реальному состоянию зоны, т.е. регулярно обновляется (см. выше). Крайне желательно, чтобы имена, указанные в этой секции, имели соответствующие адресные IN A записи в этой же базе данных.

Ниже следует секция почтовых обменников, т.е. записи типа MX (Mail Exchanger). Они указывают на сервера электронной почты, способные принимать почту для всего домена по протоколу SMTP. Чем меньше цифра перед именем, тем больший приоритет имеет данный почтовый сервер. Как правило, запись с наивысшим приоритетом относится к серверу, на котором почта заканчивает свой путь, а другие записи относятся к серверам-релеям, на которых почта может сохраняться некоторое время, пока основной почтовый сервер для зоны не доступен. Естественно, записи MX на релеи нельзя расставлять произвольно, поскольку релей обязательно должен быть сконфигурирован для приёма почты данного домена. При отсутствии записи MX для какого-либо доменного имени, почта, адресованная с этим доменным именем, будет доставляться непосредственно на хост, имеющий такое имя. Однако, такого хоста может не быть, в этом случае почта вернется отправителю с сообщением об ошибке.

Ниже, после макроса “$ORIGIN stu.”, задающего суффикс для всех записей ниже, следуют записи типа IN A, предназначенные для задания соответствия между именем хоста в зоне и его IP адресом.

Для задания псевдонимов хостам используется запись CNAME (Canonical Name). Псевдонимы удобны для указания на стандартные сервисы, такие как www, mail, ftp, а так же для задания псевдонимов, используемых для создания виртуальных серверов (см. www, docs).

Рассмотрим теперь файл зоны stu.cn.ua. Данная зона мало чем отличается от предыдущей зоны внешне.

$ORIGIN .  
stu.cn.ua 28800 IN SOA ns.stu.cn.ua. nsmaster.stu.cn.ua(   
2005033100 ; Serial  
28800 ; Refresh   
7200 ; Retry  
604800 ; Expire  
86400 ; Time To Live  
)

; authoritative name servers for zone  
IN NS ns.stu.cn.ua.  
IN NS ns1.stu.cn.ua.  
IN NS ns.cn.ua.

; mail exchangers for entire zone  
IN MX 10 stalker.stu.cn.ua.  
IN MX 15 cs.stu.cn.ua.  
IN MX 20 relay1.cn.ua

; name servers glue records  
ns.cn.ua IN A 212.86.96.10  
$ORIGIN stu.cn.ua.  
ns IN A 195.69.76.130  
ns1 IN A 195.69.76.134  
;servers  
dragon IN A 195.69.76.137  
auth IN CNAME dragon.stu.cn.ua.  
cs IN A 195.69.76.134  
stalker IN A 195.69.76.130  
www IN CNAME stalker.stu.cn.ua.  
mail IN CNAME stalker.stu.cn.ua.  
ftp IN CNAME stalker.stu.cn.ua.  
www.docs IN CNAME cs.stu.cn.ua.  
; workstations  
admin IN A 195.69.76.139

Конфигурация данной зоны практически повторяет предыдущую зону, однако отличие в том, что данная зона является субдоменом домена cn.ua. Значит, она должна быть делегирована в соответствующей зоне cn.ua примерно так, как показано в следующем фрагменте:

$ORIGIN cn.ua.  
stu IN NS ns.stu.cn.ua.  
IN NS ns1.stu.cn.ua.  
IN NS ns.cn.ua.

$ORIGIN .  
ns.stu.cn.ua IN A 195.69.76.130  
ns1.stu.cn.ua IN A 195.69.76.134

Как видно из приведенного фрагмента, в “материнской” зоне cn.ua. находятся только записи о серверах имен для делегированной зоны stu.cn.ua. Управление остальной информационной базой зону передается на сервера  ns.stu.cn.ua и  ns1.stu.cn.ua.

**Обратная зона DNS**

Теперь рассмотрим файлы обратных зон, предназначенные  для проведения обратного DNS-преобразования, т.е. "IP-адрес  в доменное имя".

Для приватных блоков адресов, таких как 10.0.0.0/8, 172.16.0.0/12 и 192.168.0.0/16 никаких проблем с делегированием, в общем-то, нет, поскольку эти адреса не маршрутизируются в Интернете и нужны только внутри приватной сети.

$ORIGIN .

0.168.192.IN-ADDR.ARPA. 86400 IN SOA ns.stu. dnsmaster.stu. (  
2006060200  
86400  
14400  
3600000

345600

)  
86400 IN NS ns.stu.  
86400 IN NS ns1.stu.

$ORIGIN 0.168.192.IN-ADDR.ARPA.

10 IN PTR stalker.stu.  
14 IN PTR cs.stu.  
17 IN PTR dragon.stu.

12 IN PTR kid.stu.

Стоит обратить внимание, что имя зоны состоит из развёрнутых по отношению к записи адреса цифр. Для адресного блока 192.168.0.0/24 имя зоны 0.168.192.IN-ADDR.ARPA. IN­ADDR.ARPA. - это специальный домен верхнего уровня, отведенный для делегирования обратных зон. В файле обратной зоны присутствует, конечно же, запись SOA, как минимум пара записей типа NS об официальных авторитетных серверах и записи типа PTR (Pointer), ставящие в соответствие адреса и имена. Обратная зона для публичных адресов 195.69.76.0/24 приведена ниже:

$ORIGIN .

76.69.195.IN-ADDR.ARPA. 86400 IN SOA ns.stu.cn.ua dnsmaster.stu.cn.ua (2004060200 86400 14400 3600000 345600 )

IN NS ns.stu.cn.ua.  
IN NS ns1.stu.cn.ua.

$ORIGIN 0.168.192.IN-ADDR.ARPA.

10 IN PTR stalker.stu.cn.ua.  
14 IN PTR cs.stu.cn.ua.  
17 IN PTR dragon.stu.cn.ua.

12 IN PTR kid.stu.cn.ua.

Отличие данной зоны от предыдущей опять же только в том, что она является публичной и должна делегироваться в соответствии с правилами выдачи и регистрации IP адресов. Вкратце необходимо отметить следующее. Выдача блоков адресов потребителям производится локальными Интернет регистратурами (LIR), которые, в свою очередь, получают их от региональных регистратур. В Европе это – бесприбыльная организация RIPE (<http://www.ripe.net/>), финансируемая провайдерами. Основные функции региональных регистратур – координация использования IP адресов и, соответственно, маршрутизации в регионе. Для получения своего блока публичных адресов необходимо заполнить соответствующие формы RIPE и направить их вашему LIR.

**Особенности размещения и конфигурирования серверов для корпоративной сети**

Поскольку сервис DNS является критическим для функционирования сети, то в сети должно быть как минимум 2 сервера. Обычно размещают их так, как показано на рисунке 2. Для внутренней сети доступны оба сервера, а для внешней – только внешний сервер. Внутренний сервер является первичным для внутренних доменов и использует внешний в качестве форварда (forwarding server), поскольку напрямую не видит домен “.” . Внешний сервер в общем случае не должен отвечать на рекурсивные запросы извне, поскольку он может быть использован как платформа для DDoS атак на другие сервера.

Об'єкт1

Рисунок 2 –  Расположение серверов имен в сети

**Установка ПО сервера DNS**

Наиболее популярным ПО, реализующим сервис DNS, является сервер bind (Berkeley Internet Name Daеmon), поддерживаемый Internet Software Consortium (http://www.isc.org). Существуют лёгкие реализации, предназначеные для работы в качестве кеширующих серверов, есть реализации, специфичные для конкретних коммерческих ОС, однако на практике лучше использовать bind, поскольку он распространяестя под лицензией GPL, т.е. вместе с исходным кодом, что гарантирует отсутсвие давно не исправляемых уязвимостей и прочих неприязностей закритого кода.

Пакет bind поставляется практически со всеми дистрибутивами Linux и xBSD.

Для работы сервера в ОС Fedora Linux не обходимо установить следующие пакеты: bind-utils – утилиты для работы с DNS и тестирования сервера, bind – собственно сервер, bind-chroot – файлы, необходимые для запуска сервера в индивидульном окружении в режиме chroot. Запуск сервера в этом режиме минимизирует потери при взломе системы через работающий сервис.

Проверить, утсновлены ли пакеты, можно командой

rpm –qa | grep bind

Если пакету не установлены, установите их командой

yum install bind-utils bind bind-chroot

При установке сервера автоматически создается конфигурация для кеширующего сервера зоны «.» и для первичных серверов локальних зон.

Для небольшого сервера на несколько мелких зон конфигурации зон обычно хранятся  в текстових вайлах. В нашем случае – это  /var/named/chroot/var/named/\* .  Для больших зон, содержащих миллионы записей, используются специальные модули хранения, которые могут сохранять данные зон в реляционных БД (PostgreSQL) либо на сервере  LDAP.

**Конфигурирование сервера DNS**

Рассмотрим подробнее пример файла конфигурации для такого случая: наш сервер является основным для внутренней корпоративной сети 192.168.0.0 и внутреннего домена “stu.”, и кэширующим для домена “.”. Все пути, приведенные ниже, в случае работы сервера в окружении chroot, нужно понищать как относительные к /var/named/chroot. Файл конфигурации /etc/named.conf приведен ниже:

options {  
 directory "/etc/namedb";   
 forward first;  
 forwarders {  
  195.69.76.130;  
 };  
};

// caching server for root domain  
zone "." {  
 type hint;  
 file "named.root";  
};

// it's not neccesary but good to resolve localhost   
zone "0.0.127.IN-ADDR.ARPA" {  
 type master;  
 file "localhost.rev";

};

// private zone for our network  
zone "stu" {  
 type master;  
 file "db.stu.private";  
 allow-transfer{  
   195.69.76.130;  
 };  
 allow-query{  
   192.168.0.0/16;  
 };  
};  
// Inverse zone for private zone stu  
zone "0.168.192.IN-ADDR.ARPA" {  
 type master;  
 file "db.192.168.0";  
 allow-transfer{  
  195.69.76.130;  
 };  
 allow-query{  
  192.168.0.0/16;  
 };  
};

В приведенном выше файле конфигурации все интуитивно понятно, однако стоит особо отметить, что сервер в DNS с адресом 195.69.76.130 используется как форвард и как вторичный сервер для внутренних доменов. Предложение allow-transfer используется для ограничения полной перекачки зоны только на вторичный сервер. Предложение allow-query указывает серверу, что отвечать на запросы по данным зонам можно только хостам из приведенного блока адресов.

Ниже приведена конфигурация второго сервера. Кроме того, он является первичным для публичных зон stu.cn.ua и 76.69.195.IN-ADDR.ARPA.

options {  
 directory "/etc/namedb";   
 forward first;  
 forwarders {  
   212.86.96.10;  
 };  
};

// caching server for root domain  
zone "." {  
 type hint;  
 file "named.root";  
};

// it's not neccesary but good to resolve localhost   
zone "0.0.127.IN-ADDR.ARPA" {  
 type master;  
 file "localhost.rev";

};

// private zone for our network  
zone "stu" {  
 type slave;  
 file "db.stu.private";  
 master {  
   192.168.0.10;  
 };  
 allow-query{  
   192.168.0.0;  
 };  
};  
// Inverse zone for private zone stu  
zone "0.168.192.IN-ADDR.ARPA" {  
 type slave;  
 file "db.192.168.0";  
 masters {  
  192.168.0.10;  
 };  
 allow-query{  
   192.168.0.0;  
 };  
};

// public zone for our network  
zone "stu.cn.ua." {  
 type master;  
 file "db.stu.cn.ua";  
 allow-transfer{  
   212.86.96.10;  
 };  
};

// Inverse zone for public zone stu.cn.ua.  
zone "76.69.195.IN-ADDR.ARPA" {  
 type master;  
 file "db.195.69.76";  
 allow-transfer{  
   212.86.96.10.  
 };  
};

Заметим, что здесь появилось предложение allow-query, необходимое для того, чтобы данный сервер мог отвечать на запросы о внутренних зонах только во внутреннюю сеть. Предложение allow-transfer используется так же, как и в предыдущем случае и ограничивает полную перекачку всей зоны только серверу с адресом 212.86.96.10, который является вторичным для данной зоны. Этот же сервер используется и как форвард. Строго говоря, такой необходимости нет, но в ситуации, когда почему-то пол-Интернета не видно, а этот форвард находится в пределах досягаемости и видит весь Интернет, использование форварда оказывается оправданным.

Естественно, многие вопросы остались за пределами этого короткого обзора. Например, конфигурирование мощной системы логов сервера bind, специальные ресурсные записи и т.д.

**Тестирование работы сервера имён**

Тестирование работы сервера производится командой dig, которая позволяет производить произвольные запросы к указанному в командной строке серверу. Ниже приведен пример запроса к DNS серверу, расположенному на локальном компьютере на предмет записи SOA для зоны stu.

al@stalker$>dig @127.0.0.1 SOA stu

; <<>> DiG 8.3 <<>> @127.0.0.1 SOA stu  
; (1 server found)  
;; res options: init recurs defnam dnsrch  
;; got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 12648  
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2,   
;; ADDITIONAL: 2  
;; QUERY SECTION:  
;; stu, type = SOA, class = IN

;; ANSWER SECTION:  
stu. 8H IN SOA ns.stu. dnsmaster.stu. (  
2005033100 ; serial  
8H ; refresh  
2H ; retry  
1W ; expiry  
1D ) ; minimum

;; AUTHORITY SECTION:  
stu. 8H IN NS ns.stu.  
stu. 8H IN NS ns1.stu.

;; ADDITIONAL SECTION:  
ns.stu. 8H IN A 192.168.0.10  
ns1.stu. 8H IN A 192.168.0.14

;; Total query time: 19 msec  
;; FROM: stalker.stu.cn.ua to SERVER: 127.0.0.1  
;; WHEN: Wed Sep 14 14:37:22 2005  
;; MSG SIZE sent: 21 rcvd: 134

Разберём вывод команды. Следует отметить, что вывод содержит собственно ответ на заданный вопрос и дополнительные сведения, отмеченные как комментарии знаками “;;” в начале строки. Поскольку мы запрашивали запись SOA, то она и показана в секции ответов. Далее следует секция авторитетности, в которой указываются авторитетные сервера для данной зоны, и, наконец, дополнительная секция, где обычно указываются IP адреса (записи IN A) для авторитетных серверов данной зоны. Ниже приведен еще один пример запроса типа A.

al@stalker$dig @127.0.0.1 A stalker.stu

; <<>> DiG 8.3 <<>> @127.0.0.1 A stalker.stu  
; (1 server found)  
;; res options: init recurs defnam dnsrch  
;; got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 14417  
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2,   
;; ADDITIONAL: 2

;; QUERY SECTION:  
;; stalker.stu, type = A, class = IN

;; ANSWER SECTION:  
stalker.stu. 8H IN A 192.168.0.10

;; AUTHORITY SECTION:  
stu. 8H IN NS ns.stu.  
stu. 8H IN NS ns1.stu.

;; ADDITIONAL SECTION:  
ns.stu. 8H IN A 192.168.0.10  
ns1.stu. 8H IN A 192.168.0.14

;; Total query time: 3 msec  
;; FROM: stalker.stu.cn.ua to SERVER: 127.0.0.1  
;; WHEN: Wed Sep 14 14:47:33 2005  
;; MSG SIZE sent: 29 rcvd: 112

Как видно из примера, кроме затребованной записи А, выданы опять же две секции об авторитетных серверах, т.е. тех серверах, где информация о данной зоне наиболее достоверна.

Если запрос попадает к кэширующему серверу, то информация о зоне в нём может быть устаревшей. Запросы для проверки обратной зоны нужно давать в форме полного имени записи PTR.

**Ход работы**

Выполнение данной лабораторной работы  состоит из следующих шагов:

1. Создайте файлы зон для прямой и обратной приватной зоны. Возьмите блок адресов 172.16.X.0, где Х – номер машины в классе. Сконфигурируйте первичный сервер DNS для этих зон на локальном компьютере и запустите его. Не забудьте, что все сообщения выводятся не на консоль, а в системный журнал. Сообщения удобнее всего просматривать в отдельном терминале командой:

tail -f /var/log/messages

2. Проведите тестирование созданных прямой и обратной зон при помощи команды dig.

3. Проведите тестирование разрешения внешних имён вашим сервером. Запросите, например информацию о зоне slashdot.org.

4. Сконфигурируйте ваш сервер как вторичный для зон, которые создал ваш сосед по лаборатории. Произведите корректировку и проверку записей зон на предмет записей типа NS. Произведите проверку командой dig, обращая особое внимание на секцию “AUTHORITY SECTION”.

**Отчет**

Отчет по проделанной работе представляется преподавателю в стандартной форме: на листах формата А4, с титульным листом (включающим тему, фио, номер зачетки и пр.), целью, ходом работы и выводами по выполненной работе. Каждое задание должно быть отражено в отчете следующим образом: 1) что надо было сделать, 2) как это сделали, 3) что получилось.

**Контрольные вопросы**

1. Почему для корпоративных зон удобнее использовать 3-х буквенные имена?
2. В каком случае сервер имён считается авторитетным?
3. Какие параметры отвечают за время обновления зоны вторичным сервером?
4. Как обеспечить защиту зоны от скачивания и от просмотра?
5. Какая запись RR применяется при создании виртуальных серверов?
6. Какая запись RR задаёт почтовый обменник для всей зоны?
7. Кокой файл содержит адреса корневых серверов имен, необходимых для инициализации кеша и рекурсивних запросив?
8. Что такое рекурсивный запрос?
9. Как производится установка ведомого сервера для конкретной зоны

**Литература**

1. Кристиан К. Введение в операционную систему UNIX. - М.: Финансы и статистика, 1985. -318 с.
2. Готье Р. Руководство по операционной системе UNIX. -М.: Финансы и статистика, 1985. -232 с.
3. Браун П. Введение в операционную систему UNIX. -М.: Мир, 1987. -287 с.
4. Томас Р., Йейтс Дж. Операционная система UNIX. Руководство для пользователей. -М.: Радио и связь, 1986. -352 с.
5. Банахан М., Раттер Э. Введение в операционную систему UNIX. -М.: Радио и связь, 1986. -341 с.
6. Тихомиров В.П., Давидов М.И. Операционная система UNIX: Инструментальные средства программирования. -М.: Финансы и статистика, 1988. -206 с.
7. Баурн С. Операционная система UNIX. -М.: Мир, 1986. -462 с.
8. Беляков М.И. и др. Инструментальная мобильная операционная система ИНМОС. -М.: Финансы и статистика, 1985 -231 с.
9. Топхем Д., Чьюнг Х.В. Юникс и Ксеникс. -М.: Мир, 1988. -392 с.
10. Беляков М.И., Рабовер Ю.И., Фридман А.Л. Мобильная операционная система. -М.: Радио и связь, 1991 -208 с.
11. Керниган Б.В., Пайк Р. UNIX - Универсальная среда программирования. -М.: Финансы и статистика, 1992 -304 с.