

Перечень заданий лабораторных работ Курс: «Технологии Интернет»

1. DHCP сервер.

Сконфигурировать DHCP-серверы на Аресах 1 и 2, DHCP-клиенты на Аресах 3,4,5 в соответствии с нижеследующим.

Арес-1 и Арес-2 являются DHCP-серверами, выдающими адреса в сети 212.16.196.0/24 (default router - 212.16.196.1). Первый выдает адреса в диапазоне 10-20, второй 40-50. Арес-3 получает адрес от Ареса-1, Арес-4 получает адрес от Ареса-2. Арес-5 получает адрес от DHCP-сервера ВГУЭС.

Обеспечить прием предложений только от "своего" сервера. Обеспечить разумную конфигурацию всех необходимых параметров стека TCP/IP даже в том случае, когда сервер не предоставил необходимых данных.

Для Ареса-1 и Ареса-2 выдавать сначала динамические адреса, потом фиксированные (по Ethernet-адресу).

Добиться полной коннективности. Каждый студент должен участвовать как в конфигурировании сервера, так и клиента.

2. DNS сервер часть 1.

Задание 1. Объяснить и устранить ошибку в конфигурации клиента DNS на своей рабочей станции.

Задание 2. Выполнить прямое преобразование для указанного преподавателем доменного имени. Обратить внимание на наличие канонического доменного имени и псевдонима (alias). Выполнить обратное преобразование для полученного IP-адреса. Преобразования выполнять путем посылки *итеративных* запросов, не забудьте указывать абсолютные доменные имена. Выполнить прямое и обратное преобразования для cache.roscredit.ru, объяснить асимметричность.

Задание 3. Получить список хостов указанной преподавателем организации.

Задание 4. Переконфигурировать хост так, чтобы он обращался к другому DNS-серверу своей зоны. Список серверов своей зоны найти.

Задание 5. Если при вызове соединения указывается только имя хоста без имени домена, то к нему автоматически присоединяется имя своего домена. Сделать так, чтобы при указании имени хоста "www" (например, "ping www") производилось соединение с хостом "www.ru", а не "www.vvsu.ru"), а к любым другим именам по-прежнему добавлялось имя своего домена.

3. DNS сервер часть 2.

Задание 6. Обновить файл инициализации кэша. Примечание: данные в файле [root.cache](#), помещенном здесь в качестве примера, существенно устарели.

Задание 7. Для указанного преподавателем хоста хоста найти зону DNS, к которой он принадлежит; серверы DNS, которые ее обслуживают; временные характеристики взаимодействия первичного и вторичного серверов для этой зоны; хосты, получающие почту для этой зоны и для данного хоста; возможные псевдонимы данного хоста. Получать только авторитативные ответы.

Задание 8. Организовать доменную систему, изображенную на рис. 2.6.

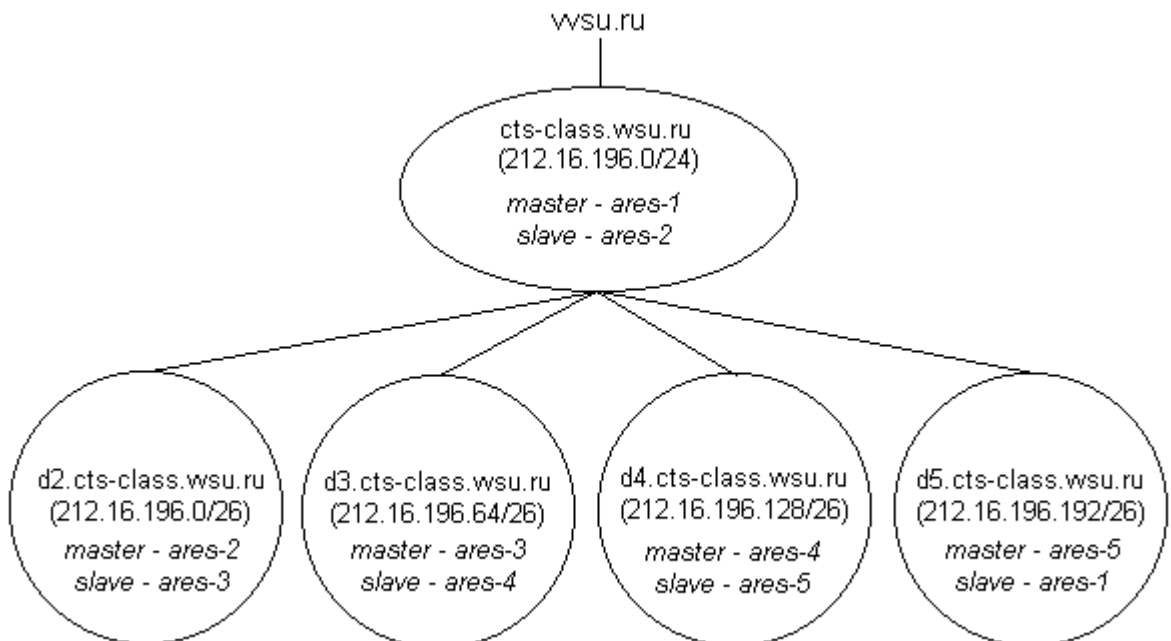


Рис. 2.6. Доменная система для лабораторной работы

Для каждого домена указаны его имя, область занимаемых IP-адресов, первичный (*master*) и вторичный (*slave*) серверы. За конфигурацию домена отвечает студент, работающий на первичном сервере этого домена. В каждом домене (кроме `cts-class.vvsu.ru`) зарегистрировать хосты А и В с IP-адресами в пространстве домена, хост А назначить обработчиком почты, приходящей на имя домена, а хост В - `www`-сервером домена. Кроме этого в каждом домене (включая `cts-class.vvsu.ru`) определить имя `ns.имя_домена`, указывающее не первичный сервер домена. Ограничения на обслуживание запросов и передачу зон не ставить.

С помощью программы `nslookup` убедиться в корректном функционировании своего домена (опрашивать 3 сервера: свой, вторичный и `maria.vvsu.ru`). Убедиться, что первичный сервер своего домена обслуживает также запросы о внешних доменных именах (например, `www.ibm.com` и т.п.).

Задание 9. Используется доменная система, построенная в задании 8. Разрешить динамическое редактирование своей зоны для первичного и вторичного серверов и `uran.vvsu.ru`. С помощью программы `nsupdate` добавить хост С с IP-адресами в пространстве домена (предварительно проверив его несуществование). С помощью программы `nslookup` убедиться, что запись появилась в базе данных (проверить прямое и обратное преобразования). Удалить хост С (предварительно проверив его существование). С помощью программы `nslookup` убедиться, что запись удалена из базы данных (проверить прямое и обратное преобразования). В домен `cts-class.vvsu.ru` вставлять хост С с IP-адресом из пространства домена `d2`.

Задание 10 (доп). Предложить способ делегирования обратной подзоны, в случае когда исходный домен располагается в сети класса А или В 15.1.0.0, а для нового поддомена выделяется область IP-адресов 15.1.200.0 netmask 255.255.248.0.

4. Организация службы электронной почты в Интернет.

Задание 1. Отправить почтовое сообщение по адресу `firekeeper@iae.nsk.su` через непосредственный диалог с SMTP-сервером, предварительно установив адрес почтового сервера, который принимает почту для этого адресата.

Задание 2. Создать на своем компьютере два специальных почтовых адреса: один - список рассылки сообщений студентам группы; другой - автомат, отвечающий на каждое письмо новогодним поздравлением.

Задание 3. Обеспечить доступ к почте для пользователей своего компьютера по протоколу POP-3. Создать POP-пользователя (пользователя, который может только получать/отправлять почту и менять пароль).

Задание 4. Получить почту по протоколу POP-3 вручную.

Задание 5. На своем компьютере проанализировать политику приема сообщений для отправки через SMTP. Проверить комбинации следующих параметров: MTA-клиент находится в своем домене или в другом; в MAIL FROM указывается адрес локального пользователя, или адрес, принадлежащий нашему домену, или адрес из другого домена; получатель (RCPT TO) - локальный пользователь, или находится в нашем домене, или находится в другом домене. Найти соответствующие правила в конфигурационном файле.

Изменить политику так, чтобы через SMTP принимались только сообщения, адрес отправителя или получателя которых - локальный для вашего компьютера; при этом адрес MTA-клиента не имеет значения.

Задание 6.

Написать агента доставки почтовых сообщений, который работает следующим образом:

Имеется рабочий каталог, в котором содержится файл `userlist` и файлы по именам пользователей. В файле `userlist` перечислены имена пользователей по одному на строку. После запуска агента имя пользователя, которому направлено сообщение, берется из командной строки и проверяется на наличие в файле `userlist`. Если проверка прошла успешно, текст сообщения считывается из стандартного ввода и помещается в конец файла, имя которого совпадает с именем пользователя (если необходимо, файл создается). Текст сообщения в файле предваряется пустой строкой, а завершается комбинацией `"\n\n"`.

Агент завершает работу со следующими состояниями выхода (exit status):

- `EX_OK = 0` - сообщение доставлено,
- `EX_NOUSER = 67` - нет такого пользователя (пользователь не найден в `userlist` или файл `userlist` отсутствует),
- `EX_USAGE = 64` - неверный вызов агента (не указано имя пользователя в командной строке),
- `EX_SOFTWARE = 70` - ошибка в программе агента (например, ошибка при открытии файла).

(Если состояние выхода агента не равно нулю, `sendmail`, запустивший этого агента, вернет отправителю соответствующее сообщение об ошибке.)

Далее при выполнении задания используется доменная система, созданная при выполнении [задания 8 темы 2](#). Сконфигурировать `sendmail` на `ares-N` так, чтобы он получал почту для адресов `user@ваш_домен.cts-class.vvsu.ru`, где `ваш_домен` - один из доменов `""`, `"d2"`, `"d3"`, `"d5"`, `"d5"`, за который ваш компьютер первично отвечает. Вся почта, направленная на такие адреса, должна быть передана написанному вами агенту доставки. При этом, разумеется, адреса `user@ares-N.vvsu.ru` должны работать как обычно.

Возможные проблемы:

- Вероятно, вы решите, что ваш агент должен запускаться sendmail'ом от имени пользователя nobody. В таком случае не забудьте установить права 777 на рабочий каталог агента и, как минимум, 644 на файл userlist.
- Не забудьте в базе данных DNS внести MX-запись для своего домена, указывающую на ваш компьютер; а при конфигурировании sendmail внести имя этого домена в класс C, чтобы sendmail опознал его как локальное.

Задание 7. В файле находится список пользователей. Сконфигурировать sendmail так, чтобы в обратных адресах писем, отправленных этими пользователями, имя вашего компьютера заменялось на athena.

5. Информационные сервисы.

Написать при помощи языков WEB-программирования скрипт для игры в виселицу (угадывание слова по буквам).

Правила игры

Сервер загадывает слово из словаря и показывает его пользователю в замаскированном виде (буквы заменены звездочками). Пользователь имеет некоторое число попыток; во время каждой попытки он может угадать одну букву. Если пользователь правильно угадывает букву или называет букву, которую он уже использовал, попытка не засчитывается. Иначе число попыток уменьшается на единицу.

Если пользователь правильно угадывает букву, сервер демаскирует в отображении слова все вхождения угаданной буквы. В любом случае сервер добавляет предложенную пользователем букву в список использованных букв, который демонстрируется во время каждой попытки для удобства пользователя. Также демонстрируется число оставшихся попыток.

Игра прекращается, если число попыток стало равным нулю (пользователь проиграл) или если угаданы все буквы в слове (пользователь выиграл).

Если пользователь на какой-либо попытке предлагает более одной буквы, считается, что пользователь пытается угадать слово целиком. При верной догадке пользователь выигрывает, иначе проигрывает независимо от числа оставшихся попыток.

Реализация

Слова выбираются случайным образом из заданного текстового файла.

Соотношение числа попыток и длины слова разумно определяется программистом. Например, число попыток есть заданная функция от длины слова; число попыток жестко привязано к каждому слову в словаре; длина слов в словаре и число попыток являются константами; при определении числа попыток используется заявленный пользователем уровень сложности.

[Подробная информация по выполнению лабораторных работ выложена на сайте athena.vvsu.ru](http://athena.vvsu.ru)