Теоретичні відомості

Процес побудови мережі складається з декількох етапів:

- планування фізичної й логічної структури мережі;
- призначення IP-адрес;
- установка мережевих апаратних засобів;
- настроювання всіх хостів на конфігурування мережевих інтерфейсів під час початкового завантаження;
- настроювання демонів маршрутизації та (або) статичних маршрутів.

Одержання та призначення IP-адрес

Зазвичай кажуть, що IP-адресу призначають конкретній host-машині. Насправді адреси призначають мережевим інтерфейсам, а не машинам. Якщо в машини декілька інтерфейсів, у неї буде кілька адрес, як мінімум по одному в кожній мережі. В кожної адреси буде свій номер мережі, що відбиває той факт, що ці інтерфейси підключені до різних фізичних мереж.

Призначаючи машині IP-адресу, потрібно вказати відповідність між цією адресою й ім'ям машини у файлі /etc/hosts, у доменній системі імен або в мережевій адміністративній базі даних. Ця дозволить звертатися до машин по їх іменам.

Використання файлу /etc/hosts - найпростіший спосіб перетворення імен в IP-адреси. Кожний рядок починається з IP-адреси й містить символьні імена, під якими відома дана адреса.

Конфігурування параметрів TCP/IP

Конфігурування параметрів TCP/IP для локальної мережі можна умовно розділити на 2 частині:

- встановлення мережевих інтерфейсів;
- конфігурування мережевих інтерфейсів.

Для побудови мережі в FreeBSD можуть використовуватися наступні типи інтерфейсів:

- Ethernet/Fast Ethernet
- ATM
- ISDN
- послідовні порти
- паралельні порти

Для правильного конфігурування інтерфейсів Ethernet/Fast Ethernet, необхідно знати ідентифікатор пристрою, використовуваний FreeBSD. Звичайно при першому завантаженні після інсталяції ядро повідомляє про наявність того або іншого Ethernet-інтерфейсу.

Конфігурування мережевих інтерфейсів Ethernet.

Один з головних файлів конфігурації - /etc/rc.conf. У даному файлі вміщується інформація про ім'я комп'ютера, особливості конфігурації, мережевих інтерфейсів, а також які служби запускаються при старті системи.

Початкова ініціалізація файлу починається при установці утилітою /stand/sysinstall. У цьому файлі в секції "Network configuration subsection" перебуває опис мережі.

Спочатку описуються можливі мережеві інтерфейси, а потім команди для настроювання кожного з інтерфейсів.

Наприклад, відомі наступні параметри TCP/IP.

- Адреса мережі 192.168.30.0
- Маска підмережі 255.255.255.0
- Адреса мережевого інтерфейсу для адаптера NE2000 192.168.30.2
- Адреса інтерфейсу ppp для модемного з'єднання 10.18.50.1
- Назва домена company.org
- Ім'я хоста bsdhost.company.org
- Шлюз 192.168.30.1
- Сервер імен DNS 192.168.30.1
- •У файлі конфігурації містяться наступні рядки:

hostname="bsdhost.company.org"

network_interfaces="lo0 ed0 tun0"

ifconfig_lo0="inet lo0 127.0.0.1" # Конфігурується обов'язково

ifconfig_ed0="inet ed0 192.168.30.2 -netmask 255.255.255.0"

ifconfig_tun0="inet tun0 10.18.50.1 -netmask 255.255.255.0"

...defaultrouter="192.168.30.1"

Далі варто визначити перелік серверів імен і доменів. Цей опис можна зробити у файлі /etc/resolv.conf:

search company.org

nameserver 192.168.30.1

Обов'язково звернить увагу на файл /etc/host.conf: рядок bind повинний розташовуватися вище рядка hosts, наприклад так:

% cat /etc/host.conf

 # $Id: ethernet.html,v 1.5 2000/02/24 09:41:11 osa Exp $

 # Default is to use the nameserver first

bind

 # If that doesn't work, then try the /etc/hosts file

hosts

 # If you have YP/NIS configured, uncomment the next line

 # nis

Конфігурування мережевих інтерфейсів утилітою *ifconfig*

Програма ifconfig використовується для включення й вимикання мережевого інтерфейсу, завдання IP-адреси, широкомовної адреси й пов'язаної з ним маски підмережі, а також для установки інших опцій і параметрів. Вона звичайно виконується під час початкового завантаження, але може застосовуватися й для внесення змін на ходу.

Команда ifconfig зазвичай має наступний формат:

ifconfig *інтерфейс [сімейство] адреса* up *опція ...*

Наприклад:

ifconfig ed0 128.138.240.1 up netmask 255.255.255.0 broadcast

128.138.240.255

Тут *інтерфейс* означає апаратний інтерфейс, до якого застосовується команда. Як правило, це двох-трьох символьне ім'я пристрою, за яким ставиться число. Приклади розповсюджених імен: ed, de, ie, le, ln, en, we, qe, lan. Ім'я інтерфейсу утворюється з ім'я драйвера пристрою, використовуваного для керування їм; звичайно воно відповідає комплекту мікросхем, що використовується в інтерфейсі. Для того, щоб з'ясувати, які інтерфейси є в системі, можна скористатися командою netstat -i.

Завдяки багаторівневій архітектурі мережевого програмного забезпечення з кожним інтерфейсом можна зв'язувати не один, а кілька протоколів. Аргумент *сімейство* показує, протоколи якого рівня потрібно конфігурувати наступними аргументами.

Для випадку використання протоколу IP аргумент *сімейство* повинен мати значення inet. Деякі версії команди ifconfig припускають значення inet, якщо аргумент *сімейство* відсутній; у системі FreeBSD потрібно вказувати його явним чином.

Параметр *адреса* задає IP-адресу інтерфейсу. Як правило, вона дається в традиційному записі із точками, але в більшості систем її можна вказувати як ім'я машини.

Інтерфейс, що закільцьовує звичайно називається lo0. Це - фіктивний елемент апаратури, через який можна маршрутизувати пакети, призначені для самої локальної машини, що дозволяє мережевим протоколам і сервісним програмам функціонувати нормально навіть на автономній машині. Інтерфейс, що закільцьовує потрібно конфігурувати як будь-який інший мережевий інтерфейс; йому варто привласнити IP-адресу 127.0.0.1 (він також відомий під ім'ям localhost).

Ключове слово up включає інтерфейс, а ключове слово down виключає його. Потім ідуть інші опції (їх може бути кілька; значення опцій задаються символічними іменами). Найбільше часто використовувані опції:

netmask задає маску піжмережі для даного інтерфейсу.

broadcast задає широкомовну IP-адресу інтерфейсу в шістнадцятковому записі або записі із точками. Правильна широкомовна адреса - та, у якій всі біти номера машини встановлені в 1.

metric стосується маршрутизації. Звичайно вартість передачі пакета з однієї мережі в іншу становить один "перехід" (якщо мережі з'єднані безпосередньо, переходів немає). Аргумент опції metric - лічильник (число) переходів, що зв'язується з даним інтерфейсом.

Команда ifconfig *інтерфейс* друкує поточні установки для зазначеного інтерфейсу. У багатьох системах -a розуміються як "всі інтерфейси".

Деякі приклади використання даної утиліти.

 # ifconfig ed0 inet 192.168.30.1 netmask 255.255.255.0

У загальному випадку це повинне працювати, але не завжди. Краще вказати необхідну кількість параметрів. Наприклад:

 # ifconfig ed0 inet 192.168.30.1 netmask 255.255.255.0 media

10base/UTP

Або

 # ifconfig ed0 inet 192.168.30.1 netmask 255.255.255.0 media

10base/UTP broadcast 192.168.30.255

У загальному випадку драйвер вибирає підходящі параметри з'єднання.

Перевірка працездатності мережевих інтерфейсів командою *ping*

Команда ping служить для примусового виклику відповіді конкретної машини. Для цього використовується дейтаграмма ECHO_REQUEST протоколу ICMP. Це протокол мережевого рівня, що не вимагає наявності серверних процесів на фондованій машині. Більшість версій команди ping

працюють у нескінченному циклі, якщо не заданий аргумент "число пакетів". Припинити нескінченне тестування можна за допомогою комбінації клавіш [Ctrl-C]. Деякі приклади використання команди ping:

% ping tigger

PING tigger.Colorado.EDU (128.138.240.26): 56 data bytes

64 bytes from 128.138.240.26: icmp_seq=0 time=12 ms

64 bytes from 128.138.240.26: icmp_seq=l time=ll ms

64 bytes from 128.138.240.26: icmp_seq=2 time=ll ms

64 bytes from 128.138.240.26: icmp_seq=3 time=ll ms

64 bytes from 128.138.240.26: icmp_seq=4 time=10 ms

^C

------itigger.Colorado.EDU PING Statistics ------

6 packets transmitted, б packets received, 0 % packet loss

round-trip (ms) min/avg/max = 10/11/12

% ping ginkgo

PING ginkgo.Colorado.EDU (128.138.241.3): 56 data bytes

^C

------iginkgo.Colorado.EDU PING Statistics ------

7 packets transmitted, 0 packets received, 100% loss

Інформація про машину tigger містить її IP-адресу, порядковий номер пакета по протоколу ICMP і час повного обходу. Машина ginkgo у другому прикладі швидше за все відключена.

Інформація про стан мережі: команда *netstat*

Найпоширеніші варіанти використання команди netstat:

• перевірка стану мережевих з'єднань;

• аналіз інформації про конфігурацію інтерфейсів;

• вивчення таблиці маршрутизації;

• одержання статистичних даних про різні мережеві протоколи.

Команда netstat -і показує стан мережевих інтерфейсів. От, приміром, вихідна інформація команди netstat -і, видана на машині host:

netstat -i

Name Mtu Net/Dest Adress Ipkts Ierrs Opkts Oerrs Coll

le0 1500 cu-capp host 51307 452 40114 311 253

iel 1500 cu-cr host 74196 902 79038 103 2271

lo0 1536 127.0.0.0 localhost 1079 0 1079 0 0

Мережі й адреси шлюзів показані - за замовчуванням - у символічній формі: їх числові еквіваленти можна одержати за допомогою опції -n. Два інтерфейси з однією і тією ж адресою host мають різні IP-адреси, які й показує команда netstat -n.

Команда netstat -r видає таблицю маршрутизації ядра.

Пункти призначення й шлюзи можуть показуватися під іменами машин або під IP-адресами. Прапори дають оцінку маршруту: й означає up (активний), g -gateway (шлюз), H - host (машинний). Прапор d (не показаний) позначає маршрут, отриманий у результаті переадресації пакетом ICMP. Прапори G і H разом означають маршрут до машини, що проходить через проміжний шлюз. Інші поля містять статистичні дані про маршрут: поточна кількість TCP-з'єднань із використанням цього маршруту, кількість посланих пакетів і ім'я використаного інтерфейсу.

Відстеження маршрутів проходження IP-пакетів утилітою *traceroute*

Програма traceroute дозволяє виявляти послідовність шлюзів, через які проходить IP-пакет на шляху до пункту свого призначення. У багатьох системах traceroute відсутній.

Синтаксис команди: traceroute *ім'я машини*

У цієї команди є дуже багато опцій, більшість із яких у повсякденній роботі не застосовуються. Як правило, *ім'я машини* може бути задане в символічній або числовій формі. Вихідна інформація - простий список машин, починаючи з першого шлюзу й закінчуючи пунктом призначення. Наприклад, на деякій машині src команда traceroute dst може видати такий результат:

traceroute to dst (128.138.202.80), 40 byte packets

1 gw1 (128.138.243.120) 3ms 2 ms 2 ms

2 gw2 (128.138.243.41) 3 ms 3 ms 3 ms

3 dst (128.138.202.80) 4 ms 4 ms 4 ms

Ця інформація говорить про те, що для того, щоб потрапити з машини src на машину dst, пакети повинні пройти два наших внутрішніх шлюзи (gw1 і gw2). Крім того, показаний час повного обходу для кожного шлюзу.

Поточний контроль трафіку утилітою *tcpdump*

Дані програми відносяться до класу інструментів перехоплення пакетів. Вони стежать за трафіком у мережі й реєструють або виводять на екран пакети, які задовольняють певним критеріям, заданим користувачем. Наприклад, можна аналізувати всі пакети, що посилаються на якусь машину або з неї, або TCP-пакети, що відносяться до конкретного мережевого з'єднання.

Запуск програми виконується з командного рядка системи FreeBSD з використанням наступних ключів:

tcpdump [-deflnNOpqSTvx] [-c count] [-F file] [-i interface] [-r file] [-s snaplen] [-T type] [-w file] [expression].

Найбільш важливими опціями є наступні:

c - вихід після перехоплення count пакетів;

d - печатка типу пакета в зручно читаємій формі й вихід;

F - файл file використовується в якості вхідного для вираження фільтрації пакетів;

i - переглядати інтерфейс interface (якщо не вказується, то проглядається

інтерфейс із найменшим номером у системному списку, наприклад, ed0);

n - не перетворювати адреси хостів в імена DNS;

N - не виводити повне DNS-ім'я ;

O - не запускати оптимізатор коду пакетів;

r - читати пакети з файлу file, створеного з використанням опції -w;

w - записувати інформацію у файл file;

Вираз expression дозволяє вказати, які пакети обробляти, а які пропускати.

Найбільш зручним для наступного аналізу є запуск програми у вигляді:

tcpdump -c CNT -i IF