Тема 4. Служба DNS: простір імен, домени. Діагностичні утиліти TCP/IP i DNS. Зони прямого і зворотного перегляду, основні і додаткові зони. Рекурсивний і ітеративний запити на дозвіл імен.

**1.**

**Протокол DHCP та служба DNS.**

**Що таке DNS і навіщо ця служба потрібна?**

**З чого все починалося.**

Що б зрозуміти основне завдання служби DNS ми зробимо короткий екскурс в історію і побачимо як саме життя змусило і підказало створити таку службу, навіть якщо б її і не було б. Коли ще не було мереж природно не було і всяких DNS служб і мережевим адміністраторам не потрібно було собі морочити голову різними протоколами, адресами і мережевими службами.

Але ось настав час, коли саме життя, зажадало обʼєднати між собою ЕОМ (електронні обчислювальні машини). Отже, починають з'являтися перші мережі. І як нам розповідають історики мережевого матеріалізму, вважається, що бабусею всіх мереж була мережа ARPANET. Унаслідок того що ЕОМ в мережі було мало, а користувачі мережі того часу як правило були «все в одному флаконі» і мережеві адміністратори і програмісти, тобто на цих машинах і в даній мережі працювали професіонали, імена ЕОМ, а більше того - їх мережеві адреси знали напам'ять.

Спочатку кожен вузол мережі ARPANET мав унікальне ім'я і унікальний номер. Потім мережа стала рости, і настав такий момент, коли кількість ЕОМ в людській пам'яті стало неможливо утримувати. І тоді мережеві розробники придумали такий текстовий файл, який знаходився на кожній машині в мережі або скажімо ЕОМ, в якому напроти імені вузла писалася його IP-адреса. Приблизно в такому вигляді:

IP адреса         -            ІМʼЯ ВУЗЛА                 ПРИМІТКИ

102.54.94.97                  server1.rk.com          # source server

38.25.63.10                    xyz.rk.com                 # x client host

І отримав назву цей файл **HOSTS**. Ідея полягала в тому, що moycomp – це легко запам'ятовуване імʼя вузла або скажімо вже в термінах DNS **ім'я хоста**, і замість того що б запам'ятовувати різний незв'язаний набір цифр, користувач писав коли хотів звернутися до потрібного хосту його ім'я (в даному прикладі ім'я хоста moycomp). А клієнтський компонент (зараз назваєтся resolver, (така спеціальна програма )) читає цей файл, знаходить ім'я машини, витягує її IP -адресу і виконує пошук за цією адресою. Ось так було на початку. І рудимент даного файлу зберігається досі на всіх версіях операційних систем Windows. Знайти і подивитися на нього ви можете на власні очі на своїй машині за адресою *% SYSTEMROOT % \ system32 \ drivers \ еtс*, відкривши його за допомогою простого блокнота.

Даний файл – HOSTS тоді заповнювався централізовано (вручну) і користувачі скачували його собі на машину. Але мережа росла і даний файл ставав дуже великим, і головне що при ручному введенні записів відповідності IP адрес імені машини, допускалися помилки. Не потрібно забувати також про те, що конфлікти виникали, коли в мережі з'являлися хости з однаковими іменами.

Коли був прийнятий стандарт TCP/IP як протокол мережі ARPANET, ця організація вирішила також удосконалити систему перетворення імен, і зупинилася на DNS, так як DNS є центральною базою даних, що містить вказівники на децентралізовані бази даних, що включають записи для кожного простору імен.

Розробником цієї системи була така організація як DARPA, яка вирішила, використовували доменну систему імен Берклі (**Berkeley Internet Name Domain, BIND**) в якості програмного забезпечення DNS.

Тобто, як бачите, якщо в двох словах, не від хорошого життя, змушені були створити таку службу, яка **автоматично ставить у відповідність «імені вузла» його IP адресу, назвали цю службу DNS**.

Тепер напишемо таку фразу.

**DNS є службою перетворення імені хоста.**

Враховуючи все вище сказане, думаю що ця фраза стає в деякій мірі зрозумілою. Ну давайте ще зупинимося на її основної функції ‒ розпізнавання імен.

Наприклад вам треба зайти на якийсь сайт (який природно знаходиться на якомусь компʼютері, названому сервером, і має свою IP-адресу). Ну самі знаєте що в браузері ви набираєте не IP адресу даного сайту а його ім'я, наприклад ім'я сайту ktpn.lntu.info. Але по суті, в будь-якій мережі компʼютери звертаються один до одного не по імені а по IP адресі. Так от.... набравши в браузері рядок ktpn.lntu.info ви все ж отримуйте зв'язок з даним сайтом, який знаходиться на певному сервері, який має свій унікальний IP адрес, і якого ви природно не знаєте. Так ось (ще раз).... отримавши таке ім'я як ktpn.lntu.info браузер запитує службу DNS типу – «А не підкажеш, який IP адрес в цього імені». DNS шукає в своїй базі даних який IP адрес відповідає цьому імені. Вона, тобто служба DNS займається питанням перетворення даного імені, то треба розуміти, що даному імені ставиться у відповідність IP адреса, якщо таке поєднання як «ім'я вузла» і його «IP-адреса» не існує в базі DNS, то DNS служба займається опитуванням всіх DNS служб типу «підкажіть чи є у вас такий IP відповідний такому імені як ktpn.lntu.info», отримавши відповідь наш компʼютер звертається до іншого компʼютера (сервера) (якого він і шукає по імені), а в нашому випадку до компʼютера з ім'ям ktpn.lntu.info вже не по імені, а як годиться в мережі за його IP адресою. Ось це і є перетворення імені.

**Службою DNS** – називається служба, що виконує перетворення символічних даних доменних імен в IP- адреси, у відповідь на запити клієнтів.

Служба DNS ставить відповідно ось такому символічному (букви це в даному випадку символічні дані) доменному імені, наприклад, як «moycomp.rk.com » відповідний йому IP – адрес, наприклад 10.10.10.10.

**DNS сервером** – називається комп'ютер на якому запушена служба DNS.

**DNS клієнтом** – називається комп'ютер який звертається до DNS сервера з запитом на дозвіл імені. Клієнт DNS функціонує «прозоро» для користувача. Що значить «прозоро»? Те що користувач не бачить, і не чує, і не відчуває як це відбувається.

У відповідь на свій запит по вирішенню імені клієнт DNS повинен отримати або IP-адресу, або повідомлення про неможливість доступу імені, послану сервером DNS. Далі даний клієнт передає отриману IP-адресу додатку якому потрібен даний IP-ардрес.

**Делегування** – операція передачі відповідальності за частину дерева доменних імен іншій особі або організації. За рахунок делегування в DNS забезпечується розподільність, адміністрування та зберігання. Технічно делегування виражається у виділенні цієї частини дерева в окрему зону, і розміщенні цієї зони на DNS-сервері (див. нижче), керованому цією особою чи організацією. При цьому в батьківську зону включаються «склеюючі» ресурсні записи (NS і А), що містять покажчики на DNS-сервера дочірньої зони, а вся інша інформація, що відноситься до дочірньої зоні, зберігається вже на DNS-серверах дочірньої зони.

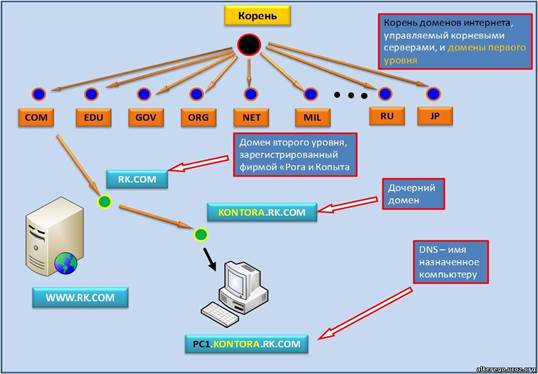
**DNS-клієнт** (від англ. Domain Name System-client − доменних імен система − клієнт) – програма або модуль в програмі, що забезпечує з'єднання із DNS-сервером для визначення IP-адреси по його доменному імені.

**Авторитетність** (англ. authoritative) – ознака розміщення зони на DNS-сервері. Відповіді DNS-сервери можуть бути двох типів: **авторитетні** (коли сервер заявляє, що сам відповідає за зону) і **неавторитетні** (англ. Non-authoritative), коли сервер обробляє запит, і повертає відповідь інших серверів. У деяких випадках замість передачі запиту далі DNS-сервер може повернути вже відоме йому (за запитами раніше) значення (режим кешування).

**Що таке** **простір імен DNS.**

Основними компонентами простору імен DNS є домени. Домени треба розуміти як групу мережевих хостів (вузлів) об'єднані за деяким логічним принципом. Домени взаємодіють один з одним за допомогою відносин «батько-нащадок», утворюючи тим самим певну ієрархію. Тобто, коли йдеться про ієрархію доменів треба розуміти ставлення «батько-нащадок». Положення домену в ієрархії визначає рівень домену (домен другого рівня, домен третього рівня і т.д. зважаючи на його положення від батьківського домену ).

В основі ієрархічної побудови простору імен DNS лежить домен, який називається **кореневим доменом** (rootdomain) (не плутайте з кореневим доменом в Active Directory).



*Рис.3. Простір імен DNS*

Кореневий домен дотримуючись традицій літератури є як «весільний генерал». Тобто, кореневий домен формальний або чисто символічний, але тим не менше він є прабатьком всіх інших нині існуючих в інтернеті доменів. Або батьком доменів першого рівня таких як *com, edu, org* … та інших, що належать різним організаціям,  так само домени належать країнам: *ua, ro, jp*... ну і так далі.

Домени, як ми знаємо виступають в ролі контейнерів. І якщо розглядати це як ієрархію, то в якості листків виступають відомості про ресурси цих доменів. Будь-який об'єкт мережі, який виходячи зі стандартів визначає службу DNS називається хост (host) або хостами.

Будь-який об'єкт простору імен DNS будь-то домен або хост має унікальне ім'я в межах батьківського контейнера.

Імена DNS можуть складатися із символів латинської алфавіту, цифр, і знака «–» тире. Деякі версії  DNS в тому числі і DNS Windows Server допускають використовувати в іменах символу підкреслення «\_», а також символів у форматі UTF- 8.

**Точно ідентифікувати об'єкт** можна тільки за допомогою його **«повного доменного імені» FQDN** (Fully Qualified Domain Name).

Повне доменне ім'я хоста утворюється з імені об'єкта і суфікса DNS.

Що таке суфікс DNS? Це всього лише перерахування імен всіх контейнерів (по суті доменів), що знаходяться між безпосередньо ім'ям хоста або об'єкта і коренем простору імен.

Єдиним об'єктом простору імен DNS, що не має ім'я є кореневий домен. Ось так, найголовніший, а без імені. Для посилання на нього використовується крапка «.»

І сьогоднішні, сучасні DNS так само представляють, по суті, базу даних з тією ж основною інформацією, яку містив попередник DNS, файл HOSTS. Але на відміну від історичного файлу HOSTS який тримав всю інформацію про всі хости в єдиному файлі, сучасні DNS представляють **розподілену систему зберігання інформації**, або DNS це – розподілена база даних.

Що таке розподілена база даних взагалі ? Це така база даних, яка зберігає інформацію не всю в одному місці, а розподіляє її по різних місцях залежно від якоїсь логіки, яка була внесена при побудові (створенні, проектуванні ) даної бази даних. Так от, DNS і є розподілена база даних, і інформація зберігається в різних місцях. І ось чому.

Як ви вже знаєте з попередньої теми – основними компонентами простору імен DNS є домени. Домени взаємодіють один з одним за допомогою відносин «предок-нащадок», утворюючи тим самим певну ієрархію (див. рис. 1). Тобто один домен є «батьком» – решта є «дочірніми» доменами стосовно батьківського.

Наприклад, домен COM – це контейнер. У доменній ієрархії COM, домен першого рівня або TLD. Тоді вирішили – нехай даний контейнер, і зберігає інформацію, про всі об'єкти, що відносяться до даного контейнера. Наприклад, інформація про зареєстрований домен RK.COM зберігатиметься в домені (контейнері) COM.

Слово «зареєстрованому» – означає що дійсно в інтернеті такому імені як RK.COM «чиновники від інтернету», офіційно присвоїли в суворій відповідності IP-адресу, що б до нього могли звертатися.

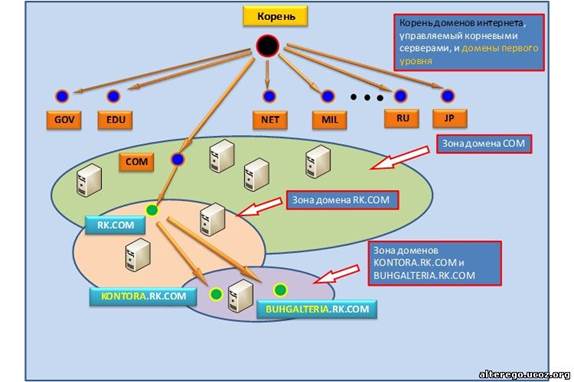
Тобто, COM контейнер або домен першого рівня TLD зберігає інформацію про свої дочірні об'єкти, домени другого рівня такі як, наприклад, у нас домен RK.COM домен другого рівня SLD. Тобто логіка, грубо кажучи така - «зробив – зберігай». А от інформація про піддомени, які породив домен UA першого рівня (TLD) зберігає інформацію про всіх його дочірні об'єкти, домени другого рівня (SDL), наприклад edu.ua.

**Зони DNS**

Ділення доменного простору імен між DNS – здійснюється за допомогою механізму зон (zone).

**Зона** являє собою базу даних, в якій містяться записи про відповідність (деякої безлічі доменних імен IP адресам). Тобто все як звичайно «ім'я – IP адреса». Кожна зона представляє тільки частину доменного простору імен. Можна сказати ще так: «Зоною – називається частина домену DNS». Чому частина? Ось наприклад, домен RK.COM може містити кілька зон, наприклад зону – KONTORA.RK.COM і зону – BUHGALTERIA.RK.COM. Як правило, термін «зона» використовується для позначення частини домену відносно DNS-сервера або нащадка. Термін «піддомен» використовується для позначення частини домену, що міститься в зоні. При наявності в домені тільки однієї зони використовується будь-який з цих термінів.

Зони слід розглядати як основний адміністративний інструмент, на рівні якого відбувається, як управління простору імен в цілому, так і управління процесом перетворення імен.

Межі зони не визначаються межами домену. Одна зона може включати в себе кілька доменів, в той час як об'єкти, що належать одному домену можуть бути розміщені в декількох зонах. Здійснюючи поділ доменного простору імен на зони системний адміністратор повинен виходити в першу чергу з зручності адміністрування і навантаженням на DNS сервери. Див. рис. 4.

*Рис.4. Зони DNS*

Параметри TCP / IP-протоколу

Тут і далі ми будемо розглядати характеристики протоколу IРv4.

**IP-адреса**

Кожен комп’ютер, що працює по протоколу TCP / IP, обов’язково має IP-адреса – 32-бітове число, що використовується для ідентифікації вузла (комп’ютера) в мережі. Адресу прийнято записувати десятковими значеннями кожного октету цього числа з поділом отриманих значень точками. Наприклад: 192.168.101.36.

IP-адреси унікальні. Це означає, що кожен комп’ютер має своє поєднання цифр, і в мережі не може бути двох комп’ютерів з однаковими адресами. IP-адреси розподіляються централізовано. Інтернет-провайдери роблять заявки в національні центри у відповідності зі своїми потребами. Отримані провайдерами діапазони адрес розподіляються далі між клієнтами. Клієнти самі можуть виступати в ролі інтернет-провайдера і розподіляти отримані IP-адреси між субклієнтами і т. д. При такому способі розподілу IP-адрес комп’ютерна система точно знає «розташування» комп’ютера, що має унікальну IP-адресу; їй достатньо переслати дані в мережу «власника». Провайдер у свою чергу проаналізує пункт призначення і, знаючи, кому віддана ця частина адрес, відправить інформацію наступному власникові піддіапазону IP-адрес, поки дані не надійдуть на комп’ютер призначення.

Для побудови локальних мереж організацій виділені спеціальні діапазони адрес. Це адреси 10.х.х.х, 192.168.х.х, 10.х.х.х, з 172.16.х.х по 172.31.х.х, 169.254.х.х (під «х» мається на увазі будь-яке число від 0 до 254). Пакети, що передаються з зазначених адрес, не маршрутизируются (іншими словами, не пересилаються) через Інтернет, тому в різних локальних мережах комп’ютери можуть мати співпадаючі адреси із зазначених діапазонів. Такі адреси часто називають сірими адресами.

**Автоматичне привласнення параметрів IP-протоколу**

Як вже зазначалося, параметри IP-протоколу індивідуальні для кожного комп’ютера. Щоб полегшити користувачам їх призначення, були розроблені спеціальні механізми, що дозволяють автоматизувати даний процес.

Сервери DHCP

У локальних мережах адміністратори встановлюють так званий сервер DHCP.

ПРИМІТКА

**Адресація APIPA**

Для полегшення побудови невеликих мереж передбачена можливість самостійного призначення адрес. Якщо в мережі немає сервера DHCP, а протокол IP встановлений з параметрами автоматичного отримання значень, то Windows присвоїть мережевій платі адреса з діапазону від 169.254.0.1 по 169.254.255.254 (маска під-мережі 255.255.0.0), попередньо перевіривши, чи не використовується вже таку адреса в системі. Даний механізм дозволяє застосовувати IP-протокол в невеликих мережах при мінімальних ручних настройках – комп’ютери побачать один одного в локальній мережі. Природно, що ніяких додаткових параметрів налаштування операційна система в цьому випадку не отримує. Наприклад, вона не буде знати, куди посилати запити, щоб отримати дані з серверів Інтернету.

Використання адреси з зазначеного раніше діапазону (перевіряється командами ipconfig або winipcfg) при наявності в мережі сервера DHCP свідчить або про несправність останнього, або про проблеми кабельного підключення даного комп’ютера.

*Порт*

При передачі даних крім IP-адрес відправника і одержувача пакет інформації містить в собі номери портів. Порт – це якесь число, яке використовується при прийомі і передачі даних для ідентифікації процесу (програми), який повинен обробити дані. Так, якщо пакет відправлений на 80-й порт, то це свідчить, що інформація призначена для сервера HTTP.

Номери портів з 1-го по 1023-й закріплені за конкретними програмами (так звані well-known-порти). Порти з номерами 1024 – 65 535 можуть бути використані в програмах власної розробки. При цьому можливі конфлікти повинні вирішуватися самими програмами шляхом вибору вільного порту. Іншими словами, порти будуть розподілятися динамічно: можливо, що при наступному старті програма вибере інше значення порту.

Знання того, які порти використовують ті чи інші прикладні програми, важливо при налаштуванні брандмауерів. Частина налаштувань в таких програмах для найбільш популярних протоколів зумовлена, і вам достатньо тільки дозволити / заборонити протоколи, керуючись їх назвами. Проте в деяких випадках доведеться звертатися до технічної документації, щоб визначити, які порти необхідно «відкрити», щоб забезпечити проходження пакетів даної програми.

Побачити, які порти реально задіяні на комп’ютері, можна за допомогою команди *netstat*.

**Брандмауер (брандмауер)**

*Міжмережевий екран (МСЕ), або брандмауер (firewall)*, – це комплекс технічних, програмних та організаційних заходів з безпечного підключення однієї мережі до іншої. Залежно від розв’язуваних завдань і конфіденційності інформації, що захищається це може бути просто невелика програма, встановлена ​​на комп’ютері, або ж спеціалізовані апаратні засоби, що реалізують вимоги конкретної організації.

Зазвичай за загальними міркувань безпеки в якості брандмауера рекомендується використовувати автономний пристрій. Іншими словами, у разі програмної реалізації брандмауера на цей комп’ютер не слід покладати рішення ніяких інших завдань.

Варіанти міжмережевих екранів

У розпорядженні адміністраторів є як апаратні міжмережеві екрани, так і програмні рішення. Різниця між цими двома варіантами достатньо умовна. Апаратний модуль – це фактично спеціалізована під певне завдання та чи інша обчислювальна система. Зазвичай для нього створюється операційна система (наприклад, IOS у Cisco) або використовується безкоштовна версія Linux.

Програмні варіанти припускають установку на типові операційні системи, наприклад: на Linux, операційні системи від Microsoft і т. П.

У Мережі існує велика кількість програм, призначених для захисту індивідуальних комп’ютерів. Наприклад, можна відзначити такі персональні міжмережеві екрани, як AtGuard, BlackICE Defender, Jammer, Kerio Personal Firewall, Outpost Firewall, Sygate Personal Firewall, Tiny personal firewall, Zone Alarm та ін. Частина цих продуктів – комерційні програми, частина має версії, доступні для безкоштовного використання.

**DHCP**

*Протокол динамічної конфігурації вузла (Dynamic Host Configuration Protocol)* – це мережевий сервіс, який дозволяє комп'ютерам в мережі автоматично отримувати налаштування мережі із сервера замість того, щоб налаштовувати кожен мережевий хост вручну. Комп'ютери, налаштовані бути клієнтами DHCP, не управляють тим, які налаштування вони отримають від DHCP сервера, і це налаштування зовсім непомітне для користувача комп'ютера. Стандарт протоколу DHCP був прийнятий у жовтні 1993 року

У загальному випадку **налаштування, передані DHCP сервером** DHCP клієнтам включають:

-          *IP адресу і мережеву маску;*

-          *IP адреса шлюза;*

-          *IP адресу DNS серверів;*

Однак DHCP сервер може також надати такі параметри налаштування, як:

*- Ім'я хоста*

*- Ім'я домена*

*- Адреса сервера часу*

*- Адреса сервера друку*

Перевага використання DHCP полягає в нестабільності налаштувань мережі, наприклад, зміна адреси DNS сервера потребує змін тільки на DHCP сервері, а всі мережеві хости будуть переналаштовані в момент наступного запиту їх DHCP клієнта до DHCP сервера. Додаткова перевага полягає в простому підключенні нових комп'ютерів до мережі, оскільки не потрібно перевіряти доступність IP адрес. Конфлікти за виділеними IP адресами також мінімальні.

DHCP сервер може **надавати налаштування**, використовуючи такі методи:

- *Виділення уручну (за MAC адресою)*:

Цей метод передбачає використання DHCP для визначення унікального апаратної адреси кожної мережевої карти, підключеної до мережі, і потім тривалого надання незмінній конфігурації кожен раз, коли DHCP клієнт робить запит на DHCP сервер, використовуючи свій мережевий пристрій. Це гарантує, що певну адресу буде автоматично присвоєно цій мережевій картці на основі її MAC адреси.

-          *Динамічне виділення (пул адрес):*

При цьому методі, DHCP сервер виділятиме IP адреси з пулу адрес (іноді званим діапазоном або областю) на період часу (або в оренду), який налаштовується на сервері, або поки клієнт не проінформує сервер, що більше взагалі не потребує адреси. Таким чином, клієнти отримують свої налаштування динамічно за принципом «перший прийшов – перший обслужений». Коли DHCP клієнт відсутній в мережі певний час, адреса вважається простроченою і повертається в пул адрес для використання іншими DHCP клієнтами. Це означає, що адреса орендується або видається на певний період часу. Після закінчення цього періоду клієнт повинен повторно домовлятися про використання адреси з сервером.

*- Автоматичне виділення*

Використовую цей метод, DHCP автоматично присвоює постійну IP адресу пристрою, вибрану з пулу доступних адрес. Зазвичай DHCP використовується для видачі тимчасового адреси, але DHCP сервер може використовувати нескінченний час оренди.

Два останніх методи можна розглядати, як автоматичні, оскільки DHCP сервер видає адреси без додаткового втручання. Єдина різниця полягає в тому, наскільки часу орендується адреса, іншими словами, коли закінчиться час використання адреси клієнтом.

**2. Служби каталогів – Active Directory, Samba**

Що таке Active Directory? Щоб зрозуміти це «поняття», і чому не лише Microsoft але і інші виробники операційних систем зробили таку ж «фішку» у своїх операційних системах (Макінтош використовує «фішку» Open Directory, Novell службу каталогів NDS), розпочнемо з історії.

Припустимо у нас три компʼютери, три користувачі і один сервер. Поки усе нормально, але ось наша організація розширилася і взяли на роботу ще трьох співробітників, а компʼютерів доки не купили. Разом у нас є мережа з 3 ПК і 6 користувачів, які доки працюють попарно по двоє на кожному ПК. Вийде така картина як на рис. 1.

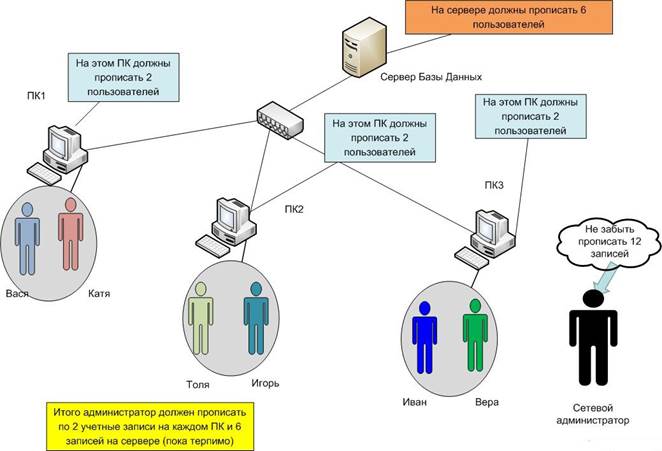


Рис. 1

Тобто, при появі в компанії нових співробітників і з урахуванням того що співробітники строго працюватимуть тільки за тими ПК, які за ними закріплені адміністратор прописує кожного співробітника на відповідному ПК. Виходячи з рис. 1 системний адміністратор прописує на ПК1 облікові записи для користувачів Васі і Каті, на ПК2 ‒ Анатолія і Ігоря, а на ПК3 ‒ Івана і Віри (як він це робить конкретно, ми розглянемо пізніше). І ще йому потрібно прописати цих усіх користувачів у кількості 6 штук на сервері щоб користувачі могли отримати доступ до сервера бази даних. Отже доки терпимо, оскільки користувачів не так вже і багато.

І тільки прописавши користувачів адміністратор, сів попити кави, як нова вказівка від керівництва. Потрібно щоб користувачі не прив'язувались до строго певного комп'ютера, а дозволити їм працювати за будь-яким комп'ютером. І адміністратор опустивши голову, йде і прописує усіх користувачів на усіх комп'ютерах. І якщо раніше, коли за певним комп'ютером працював певний користувач або користувачі, то виходячи з нашої придуманої організації на рис. 1, на кожному комп'ютері було по два облікові записи. Та тепер на кожному комп'ютері мають бути по 6 прописаних користувачів.

Отже, повернемося до нашого адміністратора. Він йде і прописує на кожному ПК усіх співробітників організації. Теж поки що не страшно. А ось вже починаються «сутінки». У нашу організацію узяли на роботу ще співробітників, і купили ще серверів і купу комп'ютерів. Дивіться рис. 2.

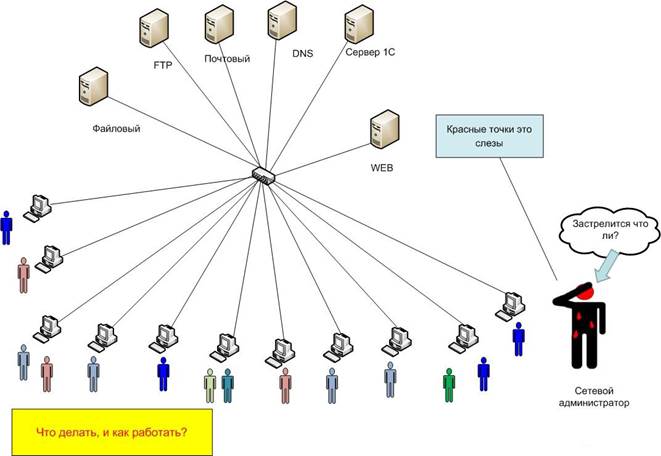


Рис. 2

Тобто саме життя продиктувало необхідність в створення інструменту, який би полегшував роботу користувачів і мережевих адміністраторів в повсякденному житті. Акаунти (облікові записи) стали зберігатися централізовано, тобто в одному місці і відпала необхідність прописувати користувачів на кожному ПК і на кожному сервері в нашій мережі, а користувачам досить один раз зареєструватися в мережі і більше не потрібно це робити на кожному сервері куди він хоче попасти. Краса. Та і взагалі, усе що є у нас в мережі Active Directory зберігає в одному місці, централізовано, а ще дозволяє «пускати» і «не пускати» зухвалих користувачів куди завгодно, і дозволяє так само управляти (адмініструвати) усім, що є у нас в мережі, не витрачаючи додаткові ресурси як у плані узяття на роботу ще адміністраторів, так і в плані купівлі додаткових засобів від сторонніх виробників, для адміністрування мережі і об'єктів що знаходяться в ній.

І що у нас вийшло з впровадженням такого поняття як **Active Directory** або як ще його називають ‒ **Служби каталогів**. Дивимося рис. 3.

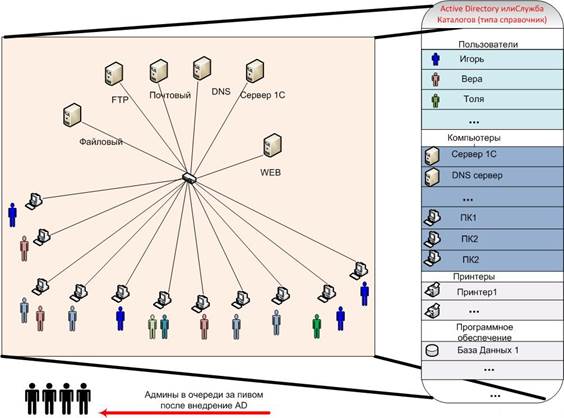


Рис. 3

**Основні терміни і поняття Active Directory.**

**Каталог (directory)** ‒ сукупність інформації про об'єкти, які тим або іншим способом пов'язані один з одним.

 Тобто це певний інформаційний ресурс, використовуваний для зберігання інформації про який-небудь об'єкт. Приміром найпростіший історичний каталог, це такий зошит в який деякий громадянин записує своїх боржників. Чи сучасніший приклад ‒ телефонний довідник що містить інформацію про абонентів телефонної мережі. У файловій системі каталоги (директорії, теки) зберігають інформацію про файли, пов'язані між собою тим або іншим способом, приміром, усі файли з теки такої-то ‒ це деякі оповідання.

У нашій мережі в каталозі може зберігатися інформація про об'єкти нашої мережі. Це можуть бути принтери, комп'ютери, користувачі, програмні продукти і різні бази даних які є у нас. Головна «фішка» або завдання тут в тому, що б надати користувачам можливість виявити (знайти) ці об'єкти і використовувати їх, але під гострозорим оком адміністратора, а не так як хочуть користувачі.

Давайте розберемося із поняттям об'єкт. І розглянемо це поняття саме в контексті мережі.

**Об'єкт** ‒ самостійна одиниця, яка представляє деякий ресурс, існуючий в мережі (ПК, користувачі, ПО, ...) з усіма його атрибутами. Що таке атрибут? Уявіть, що у нас є автомобіль, так от автомобіль це об'єкт, а його колір, потужність і інше це атрибути або властивості і характеристики. Але ми не будемо використовувати, коли говоритимемо про об'єкти у більшості випадків словосполучення «властивості і характеристики» а говоритимемо ‒ його атрибути. Приміром у такого об'єкту нашої мережі як «користувач» є такі атрибути як ‒ прізвище, ім'я, логін, пароль, адреса електронної пошти і інші. Дивіться рис 4.

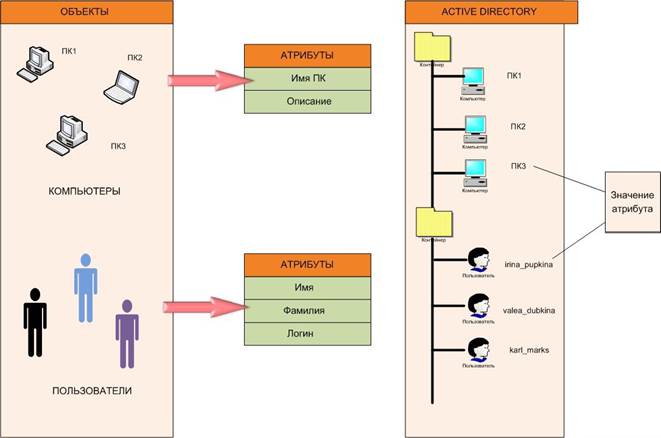


Рис. 4

**Служба каталогів (directory service)**‒ за допомогою допоміжних служб забезпечує зберігання усієї необхідною для застосування об'єктів і управління ними інформації, в одному місці (централізовано), і завдяки цьому процес виявлення і адміністрування ресурсів мережі спрощується. На відміну від каталогу, служба каталогів має одночасно дві ролі:

-          джерела інформації;

-          механізму, за допомогою якого ця інформація готується для доступу з боку користувачів.

**Служба каталогів** ‒ щось на зразок основної панелі керування мережевої операційної системи (центрального пульта). Чи що б було зовсім зрозуміло.

**Служба каталогів** ‒ це таке програмне забезпечення (така програма) у складі мережевої операційної системи, яка **зберігає усю інформацію про об'єкти мережі, і яка дозволяє виявляти ці об'єкти** (різні компʼютери, принтери, користувачів і інше, що є у нас в мережі, а усе, що є у нас в мережі, ми розглядаємо як об'єкти), **надавати їх в розпорядження користувачам, і управляти ними (адмініструвати)**.

Тобто на питання ‒ що таке служба каталогів? Ви жваво відповідайте ‒ це таке ПЗ (програмне забезпечення) яке «зберігає», «виявляє», «робить доступним користувачам» і «управляє ними», тобто об'єктами.

**Контролер домена** ‒ це комп'ютер на якому встановлено таке ПЗ як Active Directory.

Так от, вже слово «фішка», яким ми користувалися на початку нашого важкого оповідання набуло деякого сенсу, і тепер воно з'явилося перед нами як «Служба Каталогів». Що б не плутатися що таке «Служба каталогів» і «Active Directory», то напишемо так.

**Служба каталогів Active Directory ‒**це програмне забезпечення написане (розроблене) Microsoft. Тобто Microsoft назвав «службу каталогів» ім'ям «Active Directory». Макінтош назвав свою службу каталогів ‒ Open Directory, а Novell ‒ NDS.

І виходить двома словами, що **Active Directory містить каталог, в якому зберігається інформація про наші мережеві ресурси і служби що надають доступ до цієї інформації**. Ну щось на кшталт бази даних з інформацією про мережеві ресурси.

Так от «**Служба каталогів»** ‒ це з одного боку інструмент адміністрування, а з іншого засіб взаємодії кінцевого користувача з системою.

Чим більше розростається наша мережа, тим більше в ній об'єктів, якими необхідно управляти, і тут без впровадження Служби Каталогів ‒ Active Directory не обійтися.

**Обліковий запис користувача (акаунт)** ‒ складається з імені користувача (логін) і пароля (наприклад pupkin і пароль «d345rtНfa»). Не маючи цих даних не можна увійти до мережі або працювати на компʼютері. Перший раз при вході в мережу пароль вам повідомить адміністратор і залежно від того як він налаштував роботу з паролем в Active Directory ви можете відразу його змінити на ваш пароль, який окрім вас ніхто не знає. Системний адміністратор дуже МОГУТНЯ ЛЮДИНА і він може змінити будь-які налаштування вашого облікового запису, тому з ним потрібно дружити і шанувати його...

**Контейнер** ‒ контейнер аналогічний об'єкту в тому сенсі, що він також має атрибути і належить простору імен. Проте, на відміну від об'єкту, контейнер не означає нічого конкретного: він може містити групу об'єктів або інші контейнери.

Ще в Active Directory передбачено ряд компонентів, що допомагають збудувати структуру каталогу відповідно до наших потреб і діляться ці компоненти на логічні і фізичні компоненти. Рис. 5.

Логічне ділення робиться на такі компоненти як **домени, підрозділи** (OU, organizational unit), **дерева і ліси**. Фізичне на **контролери домена і сайти**. Логічні і фізичні компоненти розділені.

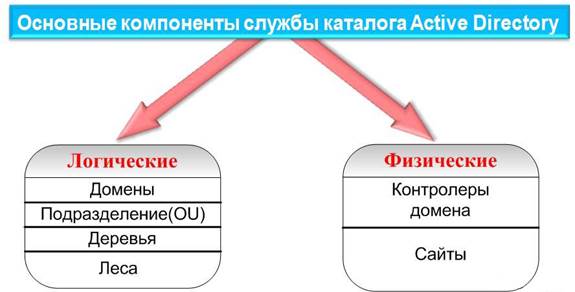


Рис. 5

Підрозділи або організаційні одиниці OU використовуються для групування об'єктів в адміністративних цілях (ну що б нам було зрозуміліше, а головне зручніше працювати). Вони можуть делегувати (передавати) адміністративні права і управляти групою об'єктів як окремим підрозділом.

**Active Directory** ‒ LDAP-сумісна реалізація інтелектуальної служби каталогів корпорації Microsoft для операційних систем родини Windows NT. Active Directory дозволяє адміністраторам використовувати групові політики (GPO) для забезпечення детального налаштування користувацького робочого середовища, розгортати ПЗ на великій кількості комп'ютерів (через групові політики або за допомогою Microsoft Systems Management Server (або System Center Configuration Manager)), встановлювати оновлення ОС, прикладного та серверного ПЗ на всіх комп'ютерах в мережі (із використанням Windows Server Update Services (WSUS); Software Update Services (SUS) раніше). Active Directory зберігає дані і налаштування середовища в централізованій базі данних. Мережі Active Directory можуть бути різного розміру: від кількох сотень до кількох мільйонів об'єктів.

Презентація Active Directory відбулась в 1999 році, продукт був вперше випущений із Windows 2000 Server, а потім був модифікований і покращений при випуску спочатку Windows Server 2003, а потім Windows Server 2003 R2.

На відміну від версій Windows до Windows 2000, котрі використовували в основному протокол NetBIOS для мережевої взаємодії, служба Active Directory інтегрована з DNS та TCP/IP.

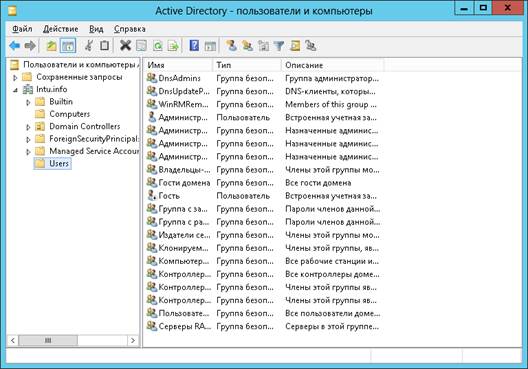


Рис. 16

**Інтеграція з UNIX**

Різні рівні взаємодії з *Active Directory* можуть бути реалізовані у більшості *UNIX* ‒ подібних операційних систем за допомогою відповідних стандарту *LDAP* клієнтів, але такі системи, як правило, не сприймають велику частину атрибутів, що асоціюються з компонентами *Windows*, наприклад групові політики і підтримку односторонніх доручень.

Сторонні постачальники пропонують інтеграцію *Active Directory* на платформах UNIX, включаючи UNIX, Linux, Mac OS X і ряд додатків на базі Java, з пакетом продуктів :

·       *Centrify DirectControl* (Centrify Corporation) *‒ сумісні* з Active Directory централізовану аутентифікацію і контроль доступу.

·       *Centrify Express* (Centrify Corporation) ‒ набір *вільних сумісних* з Active Directory сервісів для централізованої аутентифікації, моніторингу, доступу до файлів, що розділяється, і віддаленого доступу.

·       *UNAB* (Computer Associates).

·       *TrustBroker* (*CyberSafe Limited*) ‒ реалізація Kerberos.

·       *PowerBroker Identity Services*, раніше *Likewise* (BeyondTrust) ‒ дозволяє клієнтові увійти *до домена Active* Directory.

·       *Authentication Services* (Quest Software).

·       *ADmitMac* (*Thursby Software Systems*).

·       *Samba* ‒ може виконувати роль контроллера домена.

Альтернативним варіантом є використання іншої служби каталогів, наприклад *389 Directory Server (раніше Fedora Directory* *Server*, FDS), *eB2Bcom* *ViewDS v7.1XML Enabled* *Directory* або *Sun Java System Directory Server*від Sun Microsystems, що виконує двосторонню синхронізацію з Active *Directory,*реалізовуючи таким чином «відбиту» інтеграцію, коли клієнти UNIX і Linux аутентифікуються FDS, а клієнти Windows аутентифікуються Active Directory. Іншим варіантом є використання OpenLDAP з можливістю напівпрозорого перекриття, що розширює елементи віддаленого сервера LDAP додатковими атрибутами, що зберігаються в локальній базі даних.

*Active Directory* автоматизуються за допомогою Powershell.

**LDAP** (англ. Lightweight Directory Access Protocol - «полегшений протокол доступу до каталогів») - протокол прикладного рівня для доступу до служби каталогів X.500, розроблений IETF як полегшений варіант розробленого ITU-T протоколу DAP. LDAP - відносно простий протокол, який використовує TCP / IP і дозволяє проводити операції аутентифікації (bind), пошуку (search) і порівняння (compare), а також операції додавання, зміни або видалення записів. Зазвичай LDAP-сервер приймає вхідні з'єднання на порт 389 по протоколам TCP або UDP. Для LDAP-сеансів, інкапсульованих в SSL, зазвичай використовується порт 636.

**Samba** ‒ вільна реалізація мережевого протоколу SMB/CIFS. Samba випускається під ліцензією GNU. Назва Samba походить від SMB ‒ назви протоколу, який використовується Microsoft Windows для мережевої файлової системи. Головною перевагою Samba є те, що з її допомогою можливо використовувати у мережі одночасно комп'ютери з операційними системами Windows та Unix, організовувати обмін файлами між ними без окремого Windows-сервера.

Починаючи з третьої версії Samba надає служби файлів і друку для різних клієнтів Microsoft Windows, і може інтегруватися з Windows Server: або як Основний контролер домену (PDC), або як член домену. Вона також може бути частиною домену Active Directory. З версії 3 Samba підтримує файлові сервіси та сервіси для друку.

Samba 4 надає повну реалізацію контролера домену і сервісу Active Directory, сумісного з реалізацією Windows 2000 і здатного обслуговувати усі підтримувані Microsoft версії Windows-клієнтів, у тому числі Windows 8. Samba 4 є багатофункціональним серверним продуктом, що надає також реалізацію файлового сервера, сервісу друку і сервера ідентифікації (winbind).

Виконується на більшості Юнікс-подібних систем, таких як: Linux, Solaris, BSD, Mac OS X Server. Входить до більшості дистрибутивів Лінукс. В OS/2 портований samba-клієнт, плагіном до віртуальної файлової системи NetDrive.

Головною відмінністю від серверних версій Windows є відсутність підтримки для групових політик (непряма підтримка в принципі можлива) і налаштувань профілів користувачів і комп'ютерів.

**3. VPN**(Virtual Private Networks)

Для створення підключення до приватної мережі (private), припустимо мережу нашої фірми з будинку можна скористатися загальнодоступною  мережею  (public),  в нашому випадку це Інтернет.  Віртуальні приватні мережі  VPN (Virtual Private Networks) — технологія створення захищених підключень між комп'ютерами, підключеними до публічних мереж.

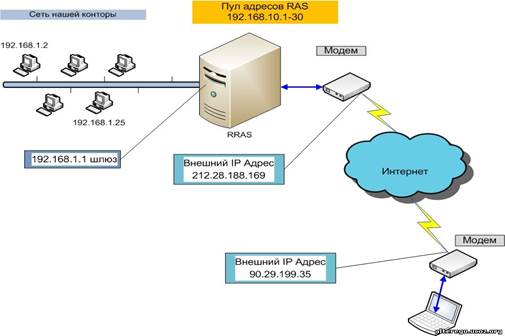
Якщо візьмемо, приміром,  випадок що ми хочемо підключитися з будинку до мережі нашого офісу те, природно ми скористаємося публічною мережею (Інтернет) і через неї підключимося до приватної мережі (private), мережі нашого офісу, або корпоративної мережі. Але виникають два питання:

·                    Як отримати доступ до мережі офісу, якщо адреси корпоративної мережі належать до діапазону внутрішніх адрес, приміром, з діапазону 192.168.0.0/24 (чому так написано читайте інші статті)  і не видні за межами мережі офісу.

·                    Як захистити дані, які пройдуть через інтернет, оскільки інформація через інтернет мережі передається у відкритому виді, і будь-який хакер може її перехопити, що нас не влаштовує з причини, припустимо секретності цієї інформації.

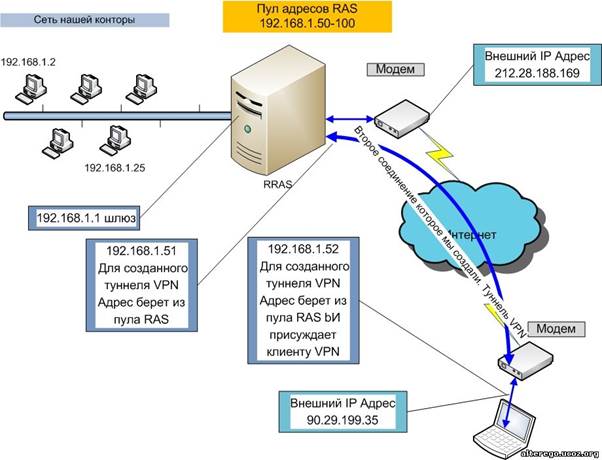
І тут нам на допомогу і приходить технологія VPN. Ця технологія орієнтована на створення так званого «захищеного тунеля» для передачі даних через публічні мережі, тобто інтернет. Усі дані в даному випадку шифруються. І ми в результаті отримаємо двоточкове з'єднання, одна точка наш комп'ютер з будинку, інша точка  комп'ютер, з офісу що відповідає за створення і підтримку такого з'єднання або скажемо точніше - сервером видаленого доступу.

У цьому прикладі користувачеві необхідно створити ще одне підключення, окрім вже існуючого підключення до Інтернету, а саме  "Підключення до віртуальної приватної мережі". І що б вказати точно з ким ми хочемо зв'язатися скористаємося  зовнішнім IP -адресом інтерфейсу сервера видаленого доступу (тобто адресою який отриманий від, приміром інтернет провайдера). На малюнку 1 видно що адреса офісу, вірніше зовнішня адреса отримана від інтернет провайдера і присуджений нашому серверу видаленого доступу це 212.28.188.169, зовнішня адреса клієнта це 90.29.199.35. Внутрішні адреси корпоративної мережі 192.168.1.0/24. У кулі для RASS будуть адреси з 192.168.10.1 - по 192.168.10.30, що не викличе проблеми зі збігом адрес.  Маршрутизацією цими підмережами займатиметься сам RASS.



Якщо усі налаштування правильні (з обох боків) те ви підключитеся до мережі вашої контори.

давайте ще раз повернемося до малюнка 1 але додамо ще яке що. Дивіться малюнок 10.



**Малюнок 10.**

Що ж ми маємо. А то що між нашим комп'ютером з будинку і сервером видаленого доступу буде встановлений захищений "віртуальний" канал, який ми і створили вище.  Клієнтський комп'ютер, в даному випадку наш,  отримає IP -адрес з пулу адрес сервера RRAS (таким чином, буде вирішено завдання маршрутизації IP -пакетов між клієнтом і корпоративною мережею), усі пакети, що передаються між клієнтом корпоративною мережею, шифруватимуться.

Аналогічно можна створити захищене віртуальне підключення між двома офісами корпоративної мережі, підключеними до різних Інтернет-провайдерів.

**Технології віртуальних приватних мереж**

Для створення віртуальних приватних мереж в системах сімейства Windows використовуються два різні протоколи — PPTP розробки корпорації Microsoft (Point — to — Point Tunneling Protocol) і L2TP, що об'єднав кращі риси протоколів PPTP і L2F компанії Cisco (Level 2 Tunneling Protocol). Основний принцип роботи обох протоколів полягає в тому, що вони створюють захищений "тунель" між користувачем і корпоративною мережею або між двома підмережами. Тунелювання полягає в тому, що пакети, що передаються в захищеній мережі, забезпечуються спеціальними заголовками (у обох протоколів свої заголовки), вміст даних в цих пакетах шифрується (у PPTP — алгоритмом MPPE компанії Microsoft, в L2TP — технологією IPSec), а потім пакет, призначений для захищеної корпоративної мережі і заголовок, що має, з IP -адресами внутрішньої корпоративної мережі, інкапсулюється в пакет, що передається по мережі Інтернет і відповідний заголовок, що має, і IP — адреси посилача і одержувача.

Відмінності між двома протоколами наступні:

·                    алгоритми шифрування (MPPE для PPTP, IPSec для L2TP);

·                    транспортне середовище (PPTP працює тільки поверх протоколу TCP/IP, L2TP може працювати також поверх протоколів X.25, Frame Relay, ATM, хоча реалізація L2TP в системі Windows працює тільки поверх TCP/IP);

·                    L2TP здійснює взаємну аутентифікацію обох сторін, що беруть участь в створенні захищеної мережі, для це використовуються сертифікати X.509 або загальний секрет (preshared key). Загальний секрет (попередній ключ) реалізований починаючи з версії Windows 2003, встановлюється у Властивостях служби RRAS на закладці "Безпека"

**VPN сервер**

Сервер видаленого доступу може обслуговуючий VPN -підключення, виконує роль VPN - сервера. Хочу звернути увагу  що при VPN -і, йде мова про той же видалений доступ до ресурсів мережі офісу або краще скажемо корпоративній мережі. Але на відміну від звичайного видаленого доступу взаємодія між клієнтом і корпоративною мережею здійснюється по захищеному каналу, який створюється за рахунок застосування спеціальних протоколів тунелювання.

Якщо мережевий адміністратор вирішив використати сервер RAS для обслуговування VPN підключень, він перед розгортанням цього сервера повинен вирішити який протокол туннелирования використовуватиме, для створення захищеного каналу зв'язку. Правда вибір тут не великий і проте він повинен буде вибирати між протоколами PPTP і L2TP.

Протокол PPTP підтримується усіма клієнтами Microsoft. Недолік цього протоколу у відсутності механізмів передаваних даних, що гарантують цілісність, і достовірність учасників з'єднання.

**Примітка:**  Під словом "цілісність" слід розуміти не лише то що що те не розбилося або не прийшло в заданій кількості, а також то що то що передавалося, не було змінене (додано або видалено з готового пакету, якщо дивитися в нашому контексті).

Протокол L2PT позбавлений цих недоліків. В цілому це більше просунутий протокол чим PPTP. Цей протокол базується на протоколі IPSec який не підтримується старішими клієнтами. Для того що б старі версії Windows - платформ працювали з цим протоколом необхідно встановити на них спеціальний клієнт - Microsoft L2TP/IPSec Client, який вільно можна викачати з сайту Microsoft.

Якщо адміністратор вибрав протокол L2TP те він ще повинен визначиться, в тому - як саме буде здійснюється взаємна аутентифікація учасників VPN з'єднання.  справа в тому що протокол IPSec на якому базується L2PT, підтримує два способи аутентифікації учасників з'єднання.  А саме:

·                    цифрові сертифікати (мова про цифрові сертифікати, що призначаються комп'ютерам);

·                    попередній  ключ (pre - shared - key)

З точки зору безпеки прийнятніше використання сертифікатів.

**Примітка:** Для використанні цифрових сертифікатів в корпоративній мережі, має бути реалізована служба сертифікації (PIK).

Розгортання VPN.

Перш ніж виконати конфігурацію сервера видаленого доступу, необхідно чітко визначиться з наступних питань:

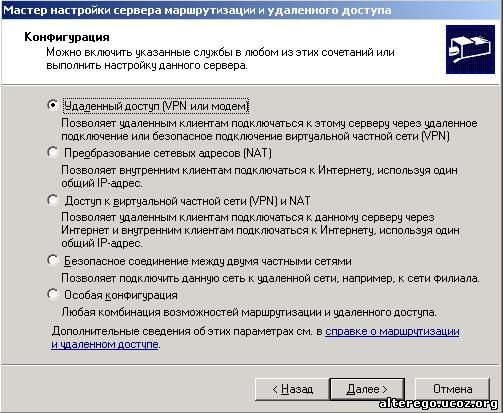
**Як здійснюватиметься розподіл IP -адрес, між клієнтами видаленого доступу?** Існують два варіанти вирішення цього питання. Перший це використання DHCP сервера нашої мережі. Другий - це визначення самим адміністратором деякого пулу статистичних адрес. Як це робиться дивитеся тут на малюнку 14.

**Яка максимальна кількість підключень, що входять, використовуватиметься?** Від відповіді на це питання залежить кількість модемів орієнтованих на підключення видалених користувачів, потрібно буде підключити до сервера видаленого доступу.

**Яка модель конфігурації параметрів видаленого підключення використовуватиметься?** Параметри видаленого підключення можуть задаватися на рівні облікових записів окремих користувачів або визначаться політикою видаленого доступу. На цьому етапі мають бути зроблені усі необхідні налаштування для облікових записів користувачів, або визначені параметри політик видаленого доступу. Як це робиться дивитеся тут малюнки 17 і 18.

**Яка схема аутентифікації використовуватиметься?** Є два варіанти - виконувати безпосередньо сервером видаленого доступу або використати засоби протоколу RADIUS і служби аутентифікації IAS. Останній варіант слід використати тоді коли мережа має декілька серверів видаленого доступу. В цьому випадку адміністратор  веде централізоване управління процесом аутентифікації користувачів.

Запустимо майстер конфігурації сервера. Як це робиться описується тут. Але тільки на відміну від конфігурації яке описане там ми виберемо конфігурацію для обслуговування VPN з'єднань. У що відкрилося вікні малюнок 12 виберіть опцію "Видалений доступ (VPN або модем)".



**Лекція 3. Основи функціонування протоколу TCP/IP (IP-адреса, маска підмережі, основний шлюз; ділення на підмережі за допомогою маски підмережі; введення в IP-маршрутизацію).**

**Адресація вузлів в IP-мережах**

У мережах TCP/IP прийнято розрізняти адреси мережевих вузлів трьох рівнів

* фізична (або локальна) адреса вузла (МАС-адрес мережевого адаптера або порту маршрутизатора); ці адреси призначаються виробниками мережевого устаткування;
* IP-адреса вузла (наприклад, 192.168.0.1), дані адреси призначаються мережевими адміністраторами або Інтернет-провайдерами;
* символьне ім'я (наприклад, [www.microsoft.com](http://www.microsoft.com/)); ці імена також призначаються мережевими адміністраторами компаній або Інтернет-провайдерами.

Розглянемо докладніше IP-адресацію.

Комп'ютери або інші складні мережеві пристрої, приєднані до декількох фізичних мереж, мають декілька IP-адрес — по одній на кожен мережевий інтерфейс. Схема адресації дозволяє проводити одиничну, широкомовну і групову адресацію. Таким чином, виділяють 3 типи IP-адрес.

1. Unicast-адреса (одинична адресація конкретному вузлу) — використовується в комунікаціях "один-до-одного".
2. Broadcast-адреса (широкомовна адреса, що відноситься до всіх адрес підмережі) — використовується в комунікаціях "один-до-всіх". У цих адресах поле ідентифікатора пристрою заповнене одиницями. IP-адресація допускає широкомовну передачу, але не гарантує її — ця можливість залежить від конкретної фізичної мережі. Наприклад, в мережах Ethernet широкомовна передача виконується з тією ж ефективністю, що і звичайна передача даних, але є мережі, які взагалі не підтримують такий тип передачі або підтримують вельми обмежено.
3. Multicast-адреса (групова адреса для багатоадресної відправки пакетів) — використовується в комунікаціях "один-до-багатьох". Підтримка групової адресації використовується в багатьох випадках, наприклад, додатках інтерактивних конференцій. Для групової передачі робочі станції і маршрутизатори використовують протокол IGMP, який надає інформацію про належність пристроїв певним групам.

**Unicast-адреса.**

Кожен мережевий інтерфейс на кожному вузлі мережі повинен мати унікальну unicast-адресу. IP-адреса має довжину 4 байти (або 32 бита). Для зручності читання адреси 32-бітні числа розбивають на октети по 8 бітів, кожен октет переводять в десяткову систему числення і при записі розділяють крапками. Наприклад, IP-адреса 11000000101010000000000000000001 записується як 192.168.0.1.

IP-адреса складається з двох частин — ідентифікатор мережі (префікс мережі, Network ID) і ідентифікатор вузла (номер пристрою, Host ID). Така схема приводить до дворівневої адресної ієрархії. Структура IP-адреси зображена на рис. 1.

https://www.e-helper.com.ua/sites/default/files/styles/images/028/clip_image001.gif

**Рисунок - 1.**

Ідентифікатор мережі ідентифікує всі вузли, розташовані на одному фізичному або логічному сегменті мережі, обмеженому IP-маршрутизаторами. Всі вузли, що знаходяться в одному сегменті повинні мати однаковий ідентифікатор мережі.

Ідентифікатор вузла ідентифікує конкретний мережевий вузол (мережевий адаптер робочої станції або сервера, порт маршрутизатора). Ідентифікатор вузла повинен бути унікальний для кожного вузла усередині IP-мережі, що має один ідентифікатор мережі.

Таким чином, в цілому IP-адреса буде унікальна для кожного мережевого інтерфейсу всієї мережі TCP/IP.

Співвідношення між ідентифікатором мережі і ідентифікатором вузла в IP-адресі визначається за допомогою маски підмережі (Network mask), яка має довжину також 4 байти і також записується в десятковій формі по 4 октети, розділених крапками. Старші біти маски підмережі, що складаються з 1, визначають, які розряди IP-адреси відносяться до ідентифікатора мережі. Молодші біти маски, що складаються з 0, визначають, які розряди IP-адреси відносяться до ідентифікатора вузла.

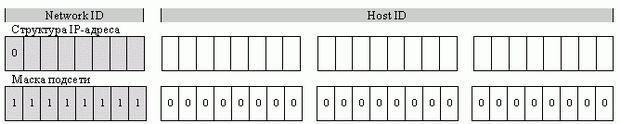
IP-адреса і маска підмережі — мінімальний набір параметрів для конфігурації протоколу TCP/IP на мережевому вузлі.

Для забезпечення гнучкості в присвоєнні адрес комп'ютерним мережам розробники протоколу визначили, що адресний простір IP повинен бути розділений на три різні класи — А, В і С.

На додаток цим трьом класам виділяють ще два класи. D — цей клас використовується для групової передачі даних. Е — клас, зарезервований для проведення експериментів.

IP-адреси класу А.

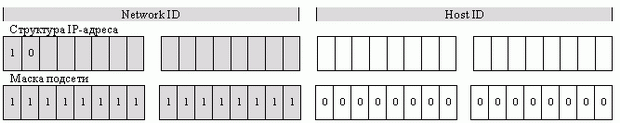
Старший біт будь-якої IP-адреси в мережі класу А завжди рівний 0. Ідентифікатор мережі складається з 8 бітів, ідентифікатор вузла — 24 біта. Маска підмережі для вузлів мереж класу A — 255.0.0.0. Структура IP-адрес класу А приведена на рис. 2.



**Рисунок - 2.**

IP-адреси класу B.

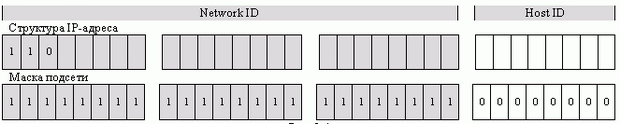
Два старших біта будь-якої IP-адресив мережі класу B завжди рівні 10. Ідентифікатор мережі складається з 16 бітів, ідентифікатор вузла — 16 бітів. Маска підмережі для вузлів мереж класу B — 255.255.0.0. Структура IP-адрес класу B приведена на рис. 3.



**Рисунок - 3.**

IP-адреси класу C.

Три старші розряди будь-якої IP-адреси в мережі класу C завжди рівні 110. Ідентифікатор мережі складається з 24 розрядів, ідентифікатор вузла — з 8 розрядів. Маска підмережі для вузлів мереж класу C — 255.255.255.0. Структура IP-адрес класу C приведена на рис. 4.



**Рисунок - 4.**

Клас D

IP-адреси класу D використовуються для групових адрес (multicast-адреса). Чотири старші розряди будь-якої IP-адресив мережі класу D завжди рівні 1110. 28 біт, що залишилися використовуються для призначення групової адреси.

Клас E

П'ять старших розрядів будь-якої IP-адреси в мережі класу E рівні 11110. Адреси даного класу зарезервовані для майбутнього використання (і не підтримуються системою Windows Server).

Правила призначення ідентифікаторів мережі (Network ID)

* перший октет ідентифікатора мережі не може бути рівний 127 (адреси вигляду 127.x.y.z призначені для відправки вузлом пакетів самому собі і використовуються як правило для відладки мережевих застосувань, такі адреси називаються loopback-адресами, або адресами зворотного зв'язку);
* всі розряди ідентифікатора мережі не можуть складатися з одних 1 (IP-адреси, всі біти яких встановлені в 1, використовуються при широкомовній передачі інформації);
* всі розряди ідентифікатора мережі не можуть складатися з одних 0 (у IP-адресі всі біти, встановлені в нуль, відповідають або даному пристрою, або даній мережі);
* ідентифікатор кожної конкретної мережі повинен бути унікальним серед підмереж, об'єднаних в одну мережу за допомогою маршрутизаторів.

Діапазони можливих ідентифікаторів мережі приведені в табл. 1.

|  |  |  |  |
| --- | --- | --- | --- |
| **Таблиця 1.** | | | |
| **Клас мережі** | **Найменший ідентифікатор мережі** | **Найбільший ідентифікатор мережі** | **Кількість мереж** |
| Клас A | 1.0.0.0 | 126.0.0.0 | 126 |
| Клас B | 128.0.0.0 | 191.255.0.0 | 16384 |
| Клас C | 192.0.0.0 | 223.255.255.0 | 2097152 |

Правила призначення ідентифікаторів вузла (Host ID)

* всі розряди ідентифікатора вузла не можуть складатися з одних 1 (ідентифікатор вузла, що складається з одних 1, використовується для широкомовних адрес, або broadcast-адрес);
* всі розряди ідентифікатора мережі не можуть складатися з одних 0 (якщо розряди ідентифікатора вузла рівні 0, то така адреса означає всю підмережу, наприклад, адреса 192.168.1.0 з маскою підмережі 255.255.255.0 позначає всю підмережу з ідентифікатором мережі 192.168.1;
* ідентифікатор вузла повинен бути унікальним серед вузлів однієї підмережі.

Діапазони можливих ідентифікаторів вузла приведені в табл. 2.

|  |  |  |  |
| --- | --- | --- | --- |
| **Таблиця 2.** | | | |
| **Клас мережі** | **Найменший ідентифікатор вузла** | **Найбільший ідентифікатор вузла** | **Кількість вузлів** |
| Клас A | w.0.0.1 | w.255.255.254 | 16777214 |
| Клас B | w.x.0.1 | w.x.255.254 | 65534 |
| Клас C | w.x.y.1 | w.x.y.254 | 254 |

Іншим способом позначення мережі, зручнішим і коротшим, є позначення мережі з мережевим префіксом. Таке позначення має вид "мережа/число біт маски підмережі". Наприклад, підмережа 192.168.1.0 з маскою підмережі 255.255.255.0 можна більш стисло записати у вигляді 192.168.1.0/24, де число 24 довжина маски підмережі в бітах.

**Публічні і приватні (приватні) IP-адреси**

Весь простір IP-адрес розділений на 2 частини: публічні адреси, які розподіляються між Інтернет-провайдерами і компаніями міжнародних організацій Internet Assigned Numbers Authority (скорочено IANA), і приватні адреси, які не контролюються IANA і можуть призначатися внутрішньо-корпоративним вузлам по розсуду мережевих адміністраторів. Якщо яка-небудь компанія придбала IP-адреси в публічній мережі, то її мережеві вузли можуть безпосередньо передавати мережевий трафік в мережу Інтернет і можуть бути прозоро доступні зі всього Інтернету. Якщо внутрішньо-корпоративні вузли мають адреси з приватної мережі, то вони можуть діставати доступ в Інтернет за допомогою протоколу трансляції мережевих адрес (NAT, Network Address Translation) або за допомогою проксі-сервера. У простому випадку за допомогою NAT можливо організувати роботу всієї компанії з використанням єдиної зареєстрованої IP-адреси.

Механізм трансляції адрес NAT перетворить IP-адреси з приватного адресного простору IP (ці адреси ще називають "внутрішні", або "сірі IP") в зареєстрований відкритий адресний простір IP. Зазвичай ці функції (NAT) виконує або маршрутизатор, або міжмережевий екран (firewall) — ці пристрої підміняють адреси в заголовках IP-пакетів, що проходять через них.

На практиці зазвичай компанії отримують через Інтернет-провайдерів невеликі мережі в просторі публічних адрес для розміщення своїх зовнішніх ресурсів — web-сайтів або поштових серверів. А для внутрішньо-корпоративних вузлів використовують приватні IP-мережі.

Простір приватних IP-адрес складається з трьох блоків:

* 10.0.0.0/8 (одна мережа класу A);
* 172.16.0.0/12 (діапазон адрес, що складається з 16 мереж класу B, — від 172.16.0.0/16 до 172.31.0.0/16);
* 192.168.0.0/16(діапазон адрес, що складається з 256 мереж класу C, — від 192.168.0.0/24 до 192.168.255.0/16).

Окрім даних трьох блоків є ще блок адрес, використовуваних для автоматичної IP-адресації (APIPA, Automatic Private IP Addressing). Автоматична IP-адресація застосовується у тому випадку, коли мережевий інтерфейс настроюється для автоматичної настройки IP-конфигурації, але при цьому в мережі відсутній сервер DHCP. Діапазон адрес для APIPA — мережа класу B 169.254.0.0/16.

**Відображення IP-адрес на фізичні адреси**

Кожен мережевий адаптер має свю унікальну фізичну адресу (або MAC-адрес). За розіменування IP-адрес адаптерів в їх фізичні адреси відповідає протокол ARP (Address Resolution Protocol). Необхідність протоколу ARP продиктована тією обставиною, що IP-адреси пристроїв в мережі призначаються незалежно від їх фізичних адрес. Тому для доставки повідомлень по мережі необхідно визначити відповідність між фізичною адресою пристрою і його IP-адресом — це називається розіменовкою адрес. В більшості випадків прикладні програми використовують саме IP-адреси. А оскільки схеми фізичної адресації пристроїв вельми різноманітні, то необхідний спеціальний, універсальний протокол. Протокол визначення адрес ARP був розроблений так, щоб його можна було використовувати для визначення адрес в різних мережах. Фактично ARP можна використовувати з довільними фізичними адресами і мережевими протоколами. Протокол ARP припускає, що кожен пристрій знає як свою IP-адресу, так і свою фізичну адресу. ARP динамічно зв'язує їх і заносить в спеціальну таблицю, де зберігаються пари "IP-адреса — фізична адреса" (зазвичай кожен запис в ARP-таблиці має час життя 10 хв.). Ця таблиця зберігається в пам'яті комп'ютера і називається кешом протоколу ARP (ARP-cache).

Робота протоколу ARP полягає у відправці повідомлень між мережевими вузлами:

* ARP Request (запит ARP) — широкомовний запит, що відправляється на фізичному рівні моделі TCP/IP, для визначення MAC-адреси вузла, що має конкретну IP-адресу;
* ARP Reply (відповідь ARP) — вузол, IP-адреса якого міститься в ARP-запмті, відправляє вузлу, що послав ARP-запит, інформацію про свою MAC-адресу;
* RARP Request, або Reverse ARP Request (зворотний ARP-запит) — запит на визначення IP-адреси по відомій MAC-адресі;
* RARP Reply, або Reverse ARP Reply (зворотна ARP-відповідь) — відповідь вузла на зворотний ARP-запрос.

**Розбиття мереж на підмережі за допомогою маски підмережі**

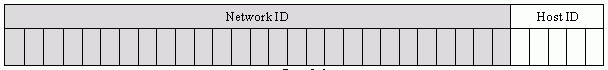
Для ефективнішого використання простору адрес IP-мережі за допомогою маски підмережі можуть бути розбиті на дрібніші підмережі (subnetting) або об'єднані в крупніші мережі (supernetting).

Розглянемо на прикладі розбиття мережі 192.168.1.0/24 (мережа класу C) на дрібніші підмережі. У початковій мережі в IP-адресі 24 біта відносяться до ідентифікатора мережі і 8 бітів — до ідентифікатора вузла. Використовуємо маску підмережі з 27 бітів, або, в десятковому позначенні, — 255.255.255.224, в двійковому позначенні — 11111111 11111111 11111111 11100000. Отримаємо наступне розбиття на підмережі:

|  |  |  |
| --- | --- | --- |
| **Таблиця 3.** | | |
| **Підмережа** | **Діапазон IP-адрес** | **Широкомовна адреса в підмережі** |
| 192.168.1.0/27 | 192.168.1.1–192.168.1.30 | 192.168.1.31 |
| 192.168.1.32/27 | 192.168.1.33–192.168.1.62 | 192.168.1.63 |
| 192.168.1.64/27 | 192.168.1.65–192.168.1.94 | 192.168.1.95 |
| 192.168.1.96/27 | 192.168.1.97–192.168.1.126 | 192.168.1.127 |
| 192.168.1.128/27 | 192.168.1.129–192.168.1.158 | 192.168.1.159 |
| 192.168.1.160/27 | 192.168.1.161–192.168.1.190 | 192.168.1.191 |
| 192.168.1.192/27 | 192.168.1.193–192.168.1.222 | 192.168.1.223 |
| 192.168.1.224/27 | 192.168.1.225–192.168.1.254 | 192.168.1.255 |

Таким чином, ми отримали 8 підмереж, в кожній з яких може бути до 30 вузлів. Нагадаємо, що ідентифікатор вузла, що складається з нулів, позначає всю підмережу, а ідентифікатор вузла, що складається з одних одиниць, позначає широкомовну адресу (пакет, відправлений на таку адресу, буде доставлений всім вузлам підмережі).

IP-адресив даних підмережах матимуть структуру:

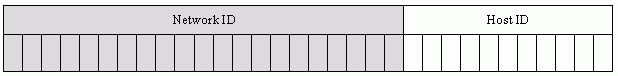


Відзначимо дуже важливий момент. З використанням такої маски вузли з такими, наприклад, IP-адресаами, як 192.168.1.48 і 192.168.1.72, знаходяться в різних підмережах, і для взаємодії даних вузлів необхідні маршрутизатори, що пересилають пакети між підмережами 192.168.1.32/27 і 192.168.1.64/27.

Примітка. Згідно стандартам протоколу TCP/IP для даного прикладу не повинно існувати підмереж 192.168.1.0/27 і 192.168.1.224/27 (тобто перша і остання підмережі). На практиці більшість операційних систем (в т.ч. системи сімейства Microsoft Windows) і маршрутизаторів підтримують роботу з такими мережами.

Аналогічно, можна за допомогою маски підмережі об'єднати дрібні мережі більші.

Наприклад, IP-адреси мережі 192.168.0.0/21 матимуть наступну структуру:



Діапазон IP-адрес даної мережі: 192.168.0.1–192.168.7.254 (всього — 2046 вузлів), широкомовна адреса підмережі — 192.168.7.255.

Переваги підмереж усередині приватної мережі:

* розбиття великих IP-мереж на підмережі (subnetting) дозволяє понизити об'єм широкомовного трафіку (маршрутизатори не пропускають широкомовні пакети);
* об'єднання невеликих мереж в крупніші мережі (supernetting) дозволяє збільшити адресний простір за допомогою мереж нижчого класу;
* зміна топології приватної мережі не впливає на таблиці маршрутизації в мережі Інтернет (зберігають тільки маршрут із загальним номером мережі);
* розмір глобальних таблиць маршрутизації в мережі Інтернет не росте;
* адміністратор може створювати нові підмережі без необхідності отримання нових номерів мереж.

Старші біти IP-адреси використовуються робочими станціями і маршрутизаторами для визначення класу адреси. Після того, як клас визначений, пристрій може однозначно обчислити межу між бітами, що використовуються для ідентифікації номера мережі, і бітами номера пристрою в цій мережі. Проте при розбитті мереж на підмережі або при об'єднанні мереж для визначення меж бітів, що ідентифікують номер підмережі, така схема не підходить. Для цього якраз і використовується 32-бітова маска підмережі, яка допомагає однозначно визначити необхідну межу. Нагадаємо, що для стандартних класів мереж маски мають наступні значення:

* 255.0.0.0 – маска для мережі класу А;
* 255.255.0.0 - маска для мережі класу В;
* 255.255.255.0 - маска для мережі класу С.

Для адміністратора мережі надзвичайно важливо знати чіткі відповіді на наступні питання:

* Скільки підмереж потрібно організації сьогодні?
* Скільки підмереж може потрібно організації в майбутньому?
* Скільки пристроїв в найбільшій підмережі організації сьогодні?
* Скільки пристроїв буде в найбільшій підмережі організації в майбутньому?

Відмова від використання тільки стандартних класів IP-мереж (A, B, і C) називається безкласовою міждоменною маршрутизацією (Classless Inter-Domain Routing, CIDR).

**Введення в IP-маршрутизацію**

Спершу уточнимо деякі поняття:

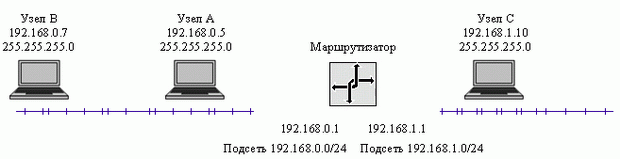
* мережевий вузол (node) — будь-який мережевий пристрій з протоколом TCP/IP;
* хост (host) — мережевий вузол, що не володіє можливостями маршрутизації пакетів;
* маршрутизатор (router) — мережевий вузол, що володіє можливостями маршрутизації пакетів

IP-маршрутизація — це процес пересилки unicast-трафіку від вузла-відправника до вузла – отримувача в IP-мережі з довільною топологією.

Коли один вузол IP-мережі відправляє пакет іншому вузлу, в заголовку IP-пакету вказуються IP-адреса вузла відправника і IP-адреса вузла-одержувача. Відправка пакету відбувається таким чином:

1. Вузол-відправник визначає, чи знаходиться вузол-одержувач в тій же самій IP-мережі, що і відправник (у локальній мережі), або в іншій IP-мережі (у віддаленій мережі). Для цього вузол-відправник проводить порозрядне логічне множення своєї IP-адресин а маску підмережі, потім порозрядне логічне множення IP-адреси вузла одержувача також на свою маску підмережі. Якщо результати співпадають, значить, обидва вузли знаходяться в одній підмережі. Якщо результати різні, то вузли знаходяться в різних підмережах.
2. Якщо обидва мережеві вузли розташовано в одній IP-мережі, то вузол-відправник спочатку перевіряє ARP-кэш на наявність в ARP-таблиці MAC-адреси вузла-одержувача. Якщо потрібний запис в таблиці є, то далі відправка пакетів проводиться безпосередньо вузлу-одержувачеві на канальному рівні. Якщо ж в ARP-таблиці потрібного запису немає, то вузол-відправник посилає ARP-запит для IP-адреси вузла-одержувача, відповідь поміщає в ARP-таблицю і після цього передача пакету також проводиться на канальному рівні (між мережевими адаптерами комп'ютерів).
3. Якщо вузол-відправник і вузол-одержувач розташовані в різних IP-мережах, то вузол-відправник посилає даний пакет мережевому вузлу, який в конфігурації відправника вказаний як "Основний шлюз" (default gateway). Основний шлюз завжди знаходиться в тій же IP-мережі, що і вузол-відправник, тому взаємодія відбувається на канальному рівні (після виконання ARP-запита). Основний шлюз — це маршрутизатор, який відповідає за відправку пакетів в інші підмережі (або безпосередньо, або через інші маршрутизатори).

Розглянемо приклад, зображений на рис. 5.



**Рисунок - 5.**

У даному прикладі 2 підмережі: 192.168.0.0/24 і 192.168.1.0/24. Підмережі об'єднані в одну мережу маршрутизатором. Інтерфейс маршрутизатора в першій підмережі має IP-адреси 192.168.0.1, в другій підмережі - 192.168.1.1. У першій підмережі є 2 вузли: вузол A (192.168.0.5) і вузол B (192.168.0.7). У другій підмережі є вузол C з IP-адресою 192.168.1.10.

Якщо вузол A відправлятиме пакет вузлу B, то спочатку він обчислить, що вузол B знаходиться в тій же підмережі, що і вузол A (тобто в локальній підмережі), потім вузол A виконає ARP-запит для IP-адреси 192.168.0.7. Після цього вміст IP-пакету буде переданий на канальний рівень, і інформація буде передана мережевим адаптером вузла A мережевому адаптеру вузла B. Це приклад прямої доставки даних (або прямої маршрутизації, direct delivery).

Якщо вузол A відправлятиме пакет вузлу C, то спочатку він обчислить, що вузол C знаходиться в іншій підмережі (тобто у віддаленій підмережі). Після цього вузол A відправить пакет вузлу, який в його конфігурації вказаний як основний шлюз (в даному випадку це інтерфейс маршрутизатора з IP-адресом 192.168.0.1). Потім маршрутизатор з інтерфейсу 192.168.1.1 виконає пряму доставку вузлу C. Це приклад непрямої доставки (або непрямої маршрутизації, indirect delivery) пакету від вузла A вузлу C. В даному випадку процес непрямої маршрутизації складається з двох операцій прямої маршрутизації.

В цілому процес IP-маршрутизації є серіями окремих операцій прямої або непрямої маршрутизації пакетів.

Кожен мережевий вузол приймає рішення о маршрутизації пакету на основі таблиці маршрутизації, яка зберігається в оперативній пам'яті даного вузла. Таблиці маршрутизації існують не тільки у маршрутизаторів з декількома інтерфейсами, але і у робочих станцій, що підключаються до мережі через мережевий адаптер. Таблицю маршрутизації в системі Windows можна подивитися по команді route print. Кожна таблиця маршрутизації містить набір записів. Записи можуть формуватися різними способами:

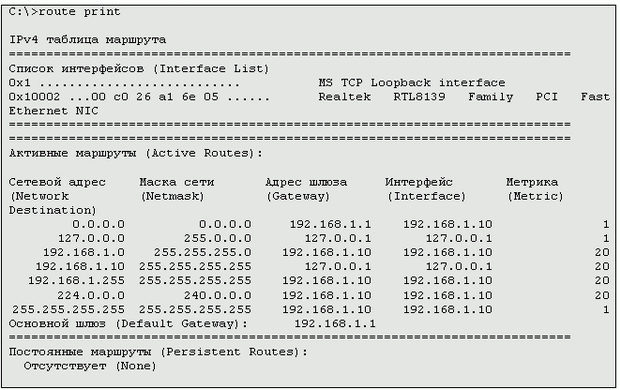
* записи, створені автоматично системою на основі конфігурації протоколу TCP/IP на кожному з мережевих адаптерів;
* статичні записи, створені командою route add або в консолі служби Routing and Remote Access Service;
* динамічні записи, створені різними протоколами маршрутизації (RIP або OSPF).

Розглянемо два приклади: таблицю маршрутизації типової робочої станції, розташованої в локальній мережі компанії, і таблицю маршрутизації сервера, що має декілька мережевих інтерфейсів.

Робоча станція.

У даному прикладі є робоча станція з системою Windows XP, з одним мережевим адаптером і такими настройками протоколу TCP/IP: IP-адреса — 192.168.1.10, маска підмережі — 255.255.255.0, основний шлюз — 192.168.1.1.

Введемо в командному рядку системи Windows команду route print, результатом роботи команди буде наступний екран ([рис. 6](http://www.intuit.ru/department/os/sysadmswin/class/free/4/3.html#image.4.6); у дужках приведений текст для англійської версії системи):



**Рисунок - 6.**

Список інтерфейсів — список мережевих адаптерів, встановлених в комп'ютері. Інтерфейс MS TCP Loopback interface присутній завжди і призначений для звернення вузла до самого себе. Інтерфейс Realtek RTL8139 Family PCI Fast Ethernet NIC — мережева карта.

Далі йде сама таблиця маршрутів. Кожен рядок таблиці — це маршрут для якої-небудь IP-мережі. Її стовпці:

Мережева адреса — діапазон IP-адрес, які досяжні за допомогою даного маршруту.

Маска мережі — маска підмережі, в яку відправляється пакет за допомогою даного маршруту.

Адреса шлюзу — IP-адреса вузла, на який пересилаються пакети, відповідні даному маршруту.

Інтерфейс — позначення мережевого інтерфейсу даного комп'ютера, на який пересилаються пакети, відповідні маршруту.

Метрика — умовна вартість маршруту. Якщо для однієї і тієї ж мережі є декілька маршрутів, то вибирається маршрут з мінімальною вартістю. Як правило, метрика — це кількість маршрутизаторів, які повинен пройти пакет, щоб потрапити в потрібну мережу.

Проаналізуємо деякі рядки таблиці.

Перший рядок таблиці відповідає значенню основного шлюзу в конфігурації TCP/IP даної станції. Мережа з адресою "0.0.0.0" позначає решту "всіх мереж, не відповідних іншим рядкам даної таблиці маршрутизації".

Другий рядок — маршрут для відправки пакетів від вузла самому собі.

Третій рядок (мережа 192.168.1.0 з маскою 255.255.255.0) — маршрут для відправки пакетів в локальній IP-мережі (тобто тій мережі, в якій розташована дана робоча станція).

Останній рядок — широкомовна адреса для всіх вузлів локальної IP-мережі.

Останній рядок на рис. 6 — список постійних маршрутів робочої станції. Це статичні маршрути, які створені командою route add. У даному прикладі немає жодного такого статичного маршруту.

Сервер.

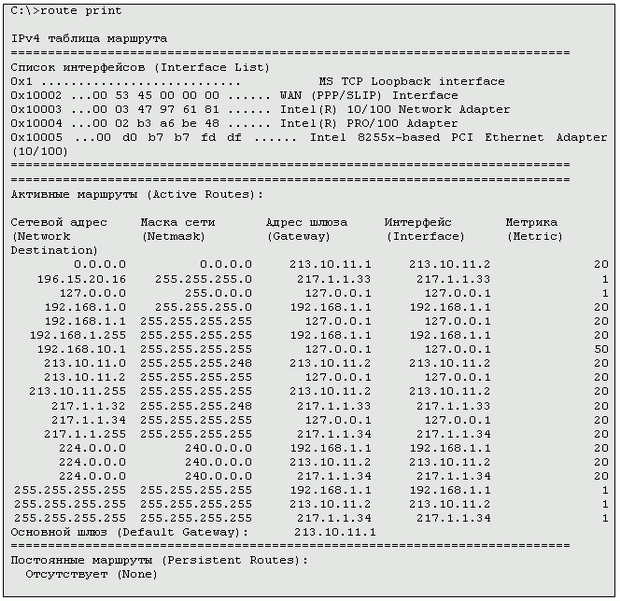
Тепер розглянемо сервер з системою Windows 2003 Server, з трьома мережевими адаптерами:

* Адаптер 1 — розташований у внутрішній мережі компанії (IP-адреса — 192.168.1.10, маска підмережі — 255.255.255.0);
* Адаптер 2 — розташований в зовнішній мережі Інтернет-провайдера ISP-1 (IP-адреса — 213.10.11.2, маска підмережі — 255.255.255.248, найближчий інтерфейс в мережі провайдера — 213.10.11.1);
* Адаптер 3 — розташований в зовнішній мережі Інтернет-провайдера ISP-2 (IP-адреса — 217.1.1.34, маска підмережі — 255.255.255.248, найближчий інтерфейс в мережі провайдера — 217.1.1.33).

IP-мережі провайдерів — умовні, IP-адреси вибрані лише для ілюстрації (хоча цілком можливо випадковий збіг з якою-небудь існуючою мережею).

Крім того, на сервері встановлена Служба маршрутизації і віддаленого доступу для управління маршрутизацією пакетів між IP-мережами і доступу в мережу компанії через модемний пул.

В даному випадку команда route print видасть таблицю маршрутизації, зображену на рис. 7.



**Рисунок - 7.**

У таблиці в списку інтерфейсів відображено три мережеві адаптери різних моделей, адаптер зворотного зв'язку (MS TCP Loopback interface) і WAN (PPP/SLIP) Interface — інтерфейс для доступу в мережу через модемний пул.

Відзначимо особливості таблиці маршрутів сервера з декількома мережевими інтерфейсами.

Перший рядок схожий на перший рядок в таблиці робочої станції. Він також відповідає значенню основного шлюзу в конфігурації TCP/IP даної станції. Відмітимо, що тільки на одному інтерфейсі можна задавати параметр "Основний шлюз". В даному випадку цей параметр був заданий на одному із зовнішніх інтерфейсів (це ж значення виведене і в кінці таблиці в рядку "Основний шлюз").

Як і в робочій станції, для кожного інтерфейсу є маршрути як для unicast-пакетів, так і для широкомовних (broadcast) для кожної підмережі.

У другому рядку міститься статичний маршрут, сконфігурований в консолі Служби маршрутизації і видаленого доступу, для пересилки пакетів в мережу 196.15.20.16/24.

Підтримка таблиць маршрутизації.

Є два способи підтримки актуального стану таблиць маршрутизації: ручний і автоматичний.

Ручний спосіб підходить для невеликих мереж. В цьому випадку в таблиці маршрутизації вручну заносяться статичні записи для маршрутів. Записи створюються або командою route add, або в консолі Служби маршрутизації і віддаленого доступу.

У великих мережах ручний спосіб стає дуже трудомістким і чреватий помилками. Автоматична побудова і модифікація таблиць маршрутизації проводиться так званими "динамічними маршрутизаторами". Динамічні маршрутизатори відстежують зміни в топології мережі, вносять необхідні зміни до таблиць маршрутів і обмінюються даною інформацією з іншими маршрутизаторами, що працюють по тих же протоколах маршрутизації. У Windows Server реалізована динамічна маршрутизація в Службі маршрутизації і віддаленого доступу. У даній службі реалізовані найбільш поширені протоколи маршрутизації — протокол RIP версій 1 і 2 і протокол OSPF.

**2 Служба DNS (домени, зони; зони прямого і зворотного перегляду; основні і додаткові зони; рекурсивний і ітеративний запити на з’ясування імен).**

**Історична довідка:** Систему доменних імен розробив в 1983 році Пол Мокапетріс. Тоді ж було проведено перше успішне тестування DNS, що стало пізніше одним з базових компонентів мережі Internet. За допомогою DNS стало можливим реалізувати масштабований розподілений механізм, що встановлює відповідність між ієрархічними іменами сайтів і числовими IP-адресами.

У 1983 році Пол Мокапетріс працював науковим співробітником інституту інформатики (Information Sciences Institute, ISI), входившого до складу інженерної школи університету Південної Каліфорнії (USC). Його керівник, Джон Постел, запропонував Полу придумати новий механізм, що встановлює зв'язки між іменами комп'ютерів і адресами Internet, - на заміну використовуваному тоді централізованому каталогу імен і адрес хостів, який підтримувала каліфорнійська компанія SRI International.

"Всі розуміли, що стара схема не зможе працювати вічно, - згадує Мокапетріс. - Зростання Internet ставало лавиноподібним. До мережі, що виникла на основі проекту ARPANET, ініційованого Пентагоном, приєднувалися все нові і нові компанії і дослідницькі інститути".

Запропоноване рішення Мокапетрісом - DNS - було розподіленою базою даних, яка дозволяла організаціям, що приєдналися до Internet, отримати свій домен.

"Як тільки організація підключалася до мережі, вона могла використовувати скільки завгодно багато комп'ютерів і сама призначати їм імена", - підкреслив Мокапетріс. Назви доменів компаній отримали суфікс .com, університетів - .edu і так далі.

Спочатку DNS була розрахована на підтримку 50 млн. записів і допускала безпечне розширення до декількох сотень мільйонів записів. По оцінках Мокапетріса, зараз налічується близько 1 млрд. імен DNS, зокрема майже 20 млн. загальнодоступних імен. Останні належать системам, розташованим за міжмережевими екранами. Їх імена невідомі звичайним Internet-користувачам.

Нова система впроваджувалася поступово, протягом декількох років. В цей час ряд дослідників експериментували з її можливостями, а Мокапетріс займався в ISI обслуговуванням і підтримкою стабільної роботи "головного сервера", побудованого на мейнфреймах компанії Digital Equipment. Копії таблиць хостів зберігалися на кожному комп'ютері, підключеному до Internet, ще приблизно до 1986 року. Потім почався масовий перехід на використання DNS.

**Необхідність відображення імен мережевих вузлів в IP-адресах**

Комп'ютери і інші мережеві пристрої, відправляючи один одному пакети по мережі, використовують IP-адреси. Проте користувачеві (людині) набагато простіше і зручніше запам'ятати деякі символічні імена мережевих вузлів, чим чотири беззмістовних для нього числа. Проте, якщо люди в своїх операціях з мережевими ресурсами використовуватимуть імена вузлів, а не IP-адреси, тоді повинен існувати механізм, що зіставляє імена вузлів і їх IP-адреси.

Є два таких механізму - локальний для кожного комп'ютера файл hosts і централізована ієрархічна служба імен DNS.

**Використання локального файлу hosts і системи доменних імен DNS для з’ясування імен мережевих вузлів**

На початковому етапі розвитку мереж, коли кількість вузлів в кожній мережі була невелика, достатньо було на кожному комп'ютері зберігати і підтримувати актуальний стан простого текстового файлу, в якому містився список мережевих вузлів даної мережі. Список влаштований дуже просто - в кожному рядку текстового файлу міститься пара "IP-адреса - ім'я мережевого вузла". У системах сімейства Windows даний файл розташований в директорії %system root%\system32\drivers\etc (де %system root% визначає директорію, в якій встановлена операційна система). Відразу після встановлення системи Windows створюється файл hosts з одним записом 127.0.0.1 localhost.

Із зростанням мереж підтримувати актуальність і точність інформації у файлі hosts стає все важчим. Для цього треба постійно оновлювати вміст цього файлу на всіх вузлах мережі. Крім того, така проста технологія не дозволяє організувати простір імен в яку-небудь структуру. Тому з'явилася необхідність в централізованій базі даних імен, що дозволяє проводити перетворення імен в IP-адреси без зберігання списку відповідності на кожному комп'ютері. Такою базою стала DNS (Domain Name System) - система іменування доменів, яка почала масову роботу в 1987 році.

Відмітимо, що з появою служби DNS актуальність використання файлу host зовсім не зникла, у ряді випадків використання цього файлу виявляється дуже ефективним.

**Служба DNS: простір імен, домени**

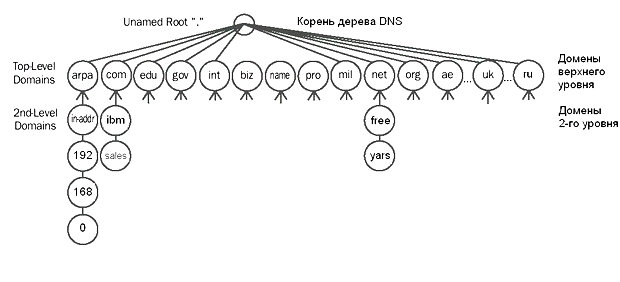
DNS - це ієрархічна база даних, що зіставляє імена мережевих вузлів і їх мережевих служб IP-адресам вузлів. Вміст цієї бази, з одного боку, розподілений по великій кількості серверів служби DNS, а з іншого боку, є централізовано керованим. У основі ієрархічної структури бази даних DNS лежить доменний простір імен (domain namespace), основною структурною одиницею якого є домен, об'єднуючий мережеві вузли (хости), а також піддомени. Процес пошуку в БД служби DNS імені якогось мережевого вузла і зіставлення цьому імені IP-адреси називається "З’ясуванням імені вузла в просторі імен DNS".

Служба DNS складається з трьох основних компонент:

* **Простір імен DNS і відповідні ресурсні записи (RR, resource** **record)** - це сама розподілена база даних DNS;
* **Сервери імен DNS** - комп'ютери, що зберігають базу даних DNS і що відповідають на запити DNS-клиєнтів;
* **DNS-клієнти (DNS-clients, DNS-resolvers)** –комп’ютери, що посилають запити серверам DNS для отримання ресурсних записів.

Простір імен.

Простір імен DNS - ієрархічна деревовидна структура, що починається з кореня, що не має імені і що позначається точкою ".". Схему побудови простору імен DNS краще всього проілюструвати на прикладі мережі Інтернет (рис. 8).



**Рисунок - 8.**

Для доменів 1-го рівня розрізняють 3 категорії імен:

* **ARPA** - спеціальне ім'я, використовуване для зворотного з’ясування DNS (з IP-адреси в повне ім'я вузла);
* **Загальні (generic) імена 1-го рівня**- 16 (на даний момент) імен, призначення яких приведене в табл. 4;
* **Двохбуквені імена для країн**- імена для доменів, зареєстрованих у відповідних країнах (наприклад, ru - для Росії, ua - для України, uk - для Великобританії і так далі).

|  |  |
| --- | --- |
| **Таблиця 4.** | |
| **Ім'я домена** | **Призначення** |
| aero | Співтовариства авіаторів |
| biz | Компанії (без прив'язки до країни) |
| com | Комерційні організації, переважно в США (наприклад, домен microsoft.com для корпорації Microsoft) |
| coop | Кооперативи |
| edu | Освітні установи в США |
| gov | Урядові установи США |
| info | Домен для організацій, що надають будь-яку інформацію для споживачів |
| int | міжнародні організації (наприклад, домен nato.int для НАТО) |
| mil | Військові відомства США |
| museum | Музеї |
| name | Глобальний домен для приватних осіб |
| net | Домен для Інтернет-провайдерів і інших організацій, керівників структурою мережі Інтернет |
| org | Некомерційні і неурядові організації, переважно в США |
| pro | Домен для професійних об'єднань (лікарів, юристів, бухгалтерів і ін.) |
| job | Кадрові агентства |
| travel | Туроператори |

Для безпосереднього відображення простору імен в простір IP-адрес служать т.з. ресурсні записи (RR, resource record). Кожен сервер DNS містить ресурсні записи для тієї частини простору імен, за яку він несе відповідальність (authoritative). табл. 5 містить опис найбільш з часто використовуваних типів ресурсних записів.

|  |  |  |
| --- | --- | --- |
| **Таблиця 5.** | | |
| **Тип ресурсного запису** | **Функція запису** | **Опис використання** |
| A | Host Address Адреса хоста, або вузла | Відображає ім'я вузла IP-адреси (наприклад, для домена microsoft.com вузлу з ім'ям [http://www.microsoft.com](http://www.microsoft.com/) зіставляється IP-адреса за допомогою такого запису: www A 207.46.199.60) |
| CNAME | Canonical Name (alias) Канонічне ім'я (псевдонім) | Зпівставляє одне ім'я з іншим |
| MX | Mail Exchanger Обмін поштою | Управляє маршрутизацією поштових повідомлень для протоколу SMTP |
| NS | Name Server Сервер імен | Указує на сервери DNS, відповідальні за конкретний домен і його піддомени |
| PTR | Pointer Покажчик | Використовується для зворотного з’ясування IP-адрес. |
| SOA | Start of Authority Початковий запис зони | Використовується для вказівки основного сервера для даної зони і опису властивостей зони |
| SRV | Service Locator Покажчик на службу | Використовується для пошуку серверів, на яких функціонують певні служби (наприклад, контроллери доменів Active Directory або сервери глобального каталога) |

Повне ім'я вузла (FQDN, fully qualified domain name) складається з декількох імен, званих мітками (label) і розділених крапкою. Найлівіша мітка відноситься безпосередньо до вузла, решта міток - список доменів від домена першого рівня до того домена, в якому знаходиться вузол (даний список розглядається справа наліво).

Сервери імен DNS.

Сервери імен DNS (або DNS-сервери) - це комп'ютери, на яких зберігаються ті частини БД простору імен DNS, за які дані сервери відповідають, і функціонує програмне забезпечення, яке обробляє запити DNS-клієнтів на розіменовку імен і видає відповіді на отримані запити.

DNS-клієнти.

DNS-клієнт - це будь-який мережевий вузол, який звернувся до DNS-сервера для з’ясування імені вузла в IP-адресу або, навпаки, IP-адреси в ім'я вузла.

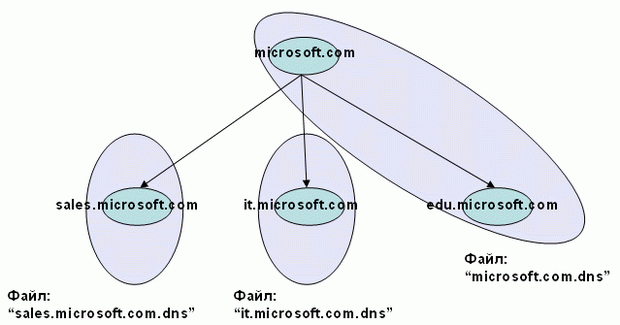
**Служба DNS: домени і зони**

Як вже розподалося вище, кожен DNS-сервер відповідає за обслуговування певної частини простору імен DNS. Інформація про домени, що зберігається в БД сервера DNS, організовується в особливі одиниці, звані зонами (zones). Зона - основна одиниця реплікації даних між серверами DNS. Кожна зона містить певну кількість ресурсних записів для відповідного домена і, можливо, його піддоменів.

Системи сімейства Windows Server підтримують наступні типи зон:

* **Стандартна основна (standard** **primary)** - головна копія стандартної зони; тільки у даному екземплярі зони допускається проводити які-небудь зміни, які потім відобразяться на серверах, що зберігають додаткові зони;
* **Стандартна додаткова (standard** **secondary)** - копія основної зони, доступна в режимі "тільки-читаня", призначена для підвищення відмовостійкості і розподілу навантаження між серверами, що відповідають за певну зону; процес реплікації змін в записах зон називається "Передачею зони" (zone transfer) (інформація про стандартні зони зберігається в текстових файлах, файли створюються в каталозі "%system root%\system32\dns", ім'я файлу, як правило, створюється з імені зони з додаванням розширення до файлу ".dns"; термін "стандартна" використовується тільки в системах сімейства Windows);
* **Інтегрована в Active** **Directory (Active** **Directory–integrated)** - вся інформація про зону зберігається у вигляді одного запису в базі даних Active Directory (такі типи зон можуть існувати тільки на серверах Windows, доменів Active Directory, що є контроллерами; у інтегрованих зонах можна жорсткіше управляти правами доступу до записів зони; зміни записів зони між різними екземплярами інтегрованої зони проводяться не за технологією передачі зони службою DNS, а механізмами реплікації служби Active Directory);
* **Зона-заглушка** (**stub**; тільки у Windows 2003) - особливий тип зони, яка для даної частини простору імен DNS містить наймінімальніший набір ресурсних записів (початковий запис зони SOA, список серверів імен, що відповідають за дану зону, і декілька записів типу A для посилання на сервери імен для даної зони).

Розглянемо на прикладі співвідношення між поняттями домена і зони. Проаналізуємо інформацію, представлену на рис. 9.



**Рисунок - 9.**

У даному прикладі простір імен DNS починається з домена [microsoft.com](http://www.intuit.ru/department/os/sysadmswin/class/free/4/microsoft.com) який містить 3 піддомени: [sales.microsoft.com](http://www.intuit.ru/department/os/sysadmswin/class/free/4/sales.microsoft.com) [it.microsoft.com](http://www.intuit.ru/department/os/sysadmswin/class/free/4/it.microsoft.com) і [edu.microsoft.com](http://www.intuit.ru/department/os/sysadmswin/class/free/4/edu.microsoft.com) (домени на малюнку позначені маленькими горизонтальними овалами). Домен - поняття чисто логічне. До розподілу імен відноситься - деревовидна структура доменів і піддоменів і для кожного домена свій список вузлів. Поняття домена ніяк не пов'язане з технологією зберігання інформації про домен. Зона - це спосіб представлення інформації про домен і його піддомени в сховищі тих серверів DNS, які відповідають за даний домен і піддомени. У даній ситуації, якщо для зберігання вибрана технологія стандартних зон, то розміщення інформації про домени може бути реалізоване таким чином:

* записи, що відносяться до доменів [microsoft.com](http://www.intuit.ru/department/os/sysadmswin/class/free/4/microsoft.com) і [edu.microsoft.com](http://www.intuit.ru/department/os/sysadmswin/class/free/4/edu.microsoft.com) зберігаються в одній зоні у файлі "microsoft.com.dns" (на малюнку зона позначена великим похилим овалом);
* управління доменами [sales.microsoft.com](http://www.intuit.ru/department/os/sysadmswin/class/free/4/sales.microsoft.com) і [it.microsoft.com](http://www.intuit.ru/department/os/sysadmswin/class/free/4/it.microsoft.com) делеговано іншим серверам DNS, для цих доменів на інших серверах створені відповідні файли "sales.microsoft.com.dns" і "it.microsoft.com.dns" (дані зони позначені великими вертикальними овалами).

Делегування управління - передача відповідальності за частину простору імен іншим серверам DNS.

**Зони прямого і зворотного перегляду**

Зони, розглянуті в попередньому прикладі, є зонами прямого перегляду (forward lookup zones). Дані зони служать для з’ясування імен вузлів в IP-адреси. Найбільш часто використовувані для цього типи записів: A, CNAME, SRV.

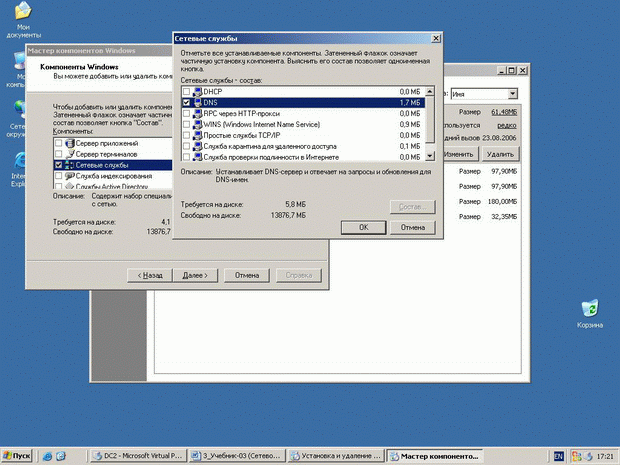
Для визначення імені вузла по його IP-адресу слугують зони зворотного перегляду (reverse lookup zones), основний тип запису в "зворотних" зонах - PTR. Для вирішення даного завдання створений спеціальний домен з ім'ям in-addr.arpa. Для кожної IP-мережі в такому домені створюються відповідні піддомени, створені з ідентифікатора мережі, записаного в зворотному порядку. Записи в такій зоні зіставлятимуть ідентифікатору вузла повне FQDN-ім’я даного вузла. Наприклад, для IP-мережі 192.168.0.0/24 необхідно створити зону з ім'ям "0.168.192.in-addr.arpa". Для вузла з IP-адресом 192.168.0.10 і ім'ям host.company.ua у даній зоні повинен бути створена запис "10. 0. 168. 192 PTR host.company.ua".

**Алгоритми роботи ітеративних і рекурсивних запитів DNS**

**Встановлення служби DNS**

Встановлення служби DNS (як і інших компонент системи) проводиться досить просто за допомогою майстра установки компонент Windows:

1. Відкрийте Панель управління.
2. Виберіть пункт "Установка і видалення програм".
3. Натисніть кнопку "Установка компонентів Windows".
4. Виберіть "Мережеві служби" - кнопка "Додаткове" (у жодному випадку не знімайте галочку у назви "Мережеві служби").
5. Відзначте службу DNS.



**Рисунок - 10.**

1. Кнопка "ОК", кнопка "Далі", кнопка "Готово".

Якщо система попросить вказати шлях до дистрибутива системи, введіть шлях до каталогу з дистрибутивом.

Виконаємо дану дію на обох серверах.

**Створення основної зони прямого перегляду.**

На сервері DC1 створимо стандартну основну зону з ім'ям world.ua.

1. Відкриємо консоль DNS.
2. Виберемо розділ "Зони прямого перегляду".
3. Запустимо майстер створення зони (тип зони - "Основна", динамічні оновлення - встановити, решта параметрів - за умовчанням).
4. Введемо ім'я зони - world.ua.
5. Встановимо передачу даної зони на будь-який сервер DNS (Консоль DNS - зона world.ua - Властивості - Закладка "Передачі зон" - Відзначте "Вирішити передачі" і "На будь-який сервер").

**Створення додаткової зони прямого перегляду.**

На сервері DC2 створимо стандартну додаткову зону з ім'ям world.ua.

1. Відкриємо консоль DNS.
2. Виберемо розділ "Зони прямого перегляду"
3. Запустимо майстер створення зони (вибрати: тип зони - "Додаткова", IP-адреса master-серверу (з якого копіюватиметься зона) - адреса сервера DC1, решта параметрів - за замовчанням)
4. Введемо ім'я зони - world.ua.
5. Перевіримо в консолі DNS появу зони.

**Настройка вузлів для виконання динамічної реєстрації на сервері DNS.**

Для виконання даного завдання потрібно виконати ряд дій як на сервері DNS, так і в налашуваннях клієнта DNS.

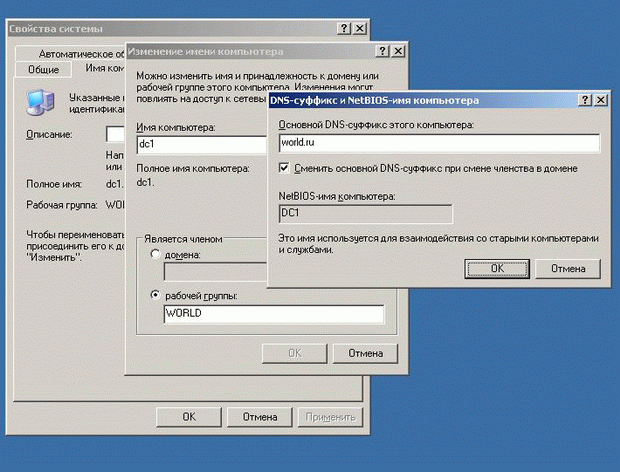
Сервер DNS.

1. Створити відповідну зону.
2. Встановити динамічні оновлення.

Це нами вже виконано.

Клієнт DNS.

1. Вказати в налаштуваннях протоколу TCP/IP адресу DNS-сервера, якому віддається перевага, - той сервер, на якому дозволені динамічні оновлення (у нашому прикладі - сервер DC1).
2. У повному імені комп'ютера вказати відповідний DNS-суфікс (у нашому прикладі - world.ua). Для цього - "Мій комп'ютер" - "Властивості" - Закладка "Ім'я комп'ютера" - Кнопка "Змінити" - Кнопка "Додатково" - в порожньому текстовому полі впишемо назву домена world.ua - кнопка "ОК" (3 рази)).



**Рисунок - 11.**

Після цього система запропонує перезавантажити комп'ютер. Після виконання перезавантаження на сервері DNS в зоні world.ua автоматично створяться записи типу A для наших серверів (рис. 12).

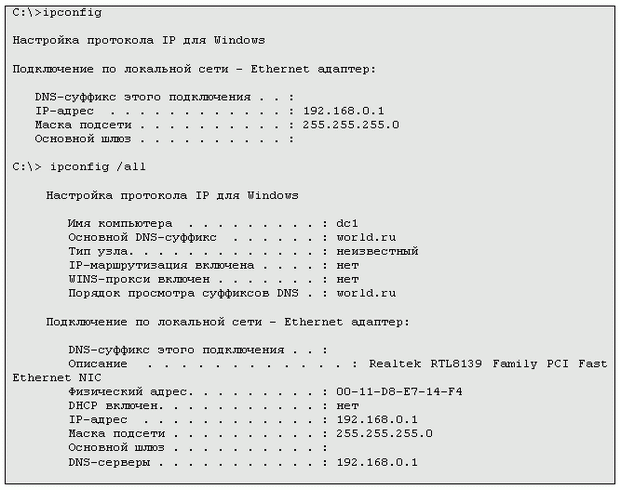


**Рисунок - 12.**

**Створення зони зворотного перегляду.**

1. Відкриємо консоль DNS.
2. Виберемо розділ "Зони зворотного перегляду".
3. Запустимо майстер створення зони (вибрати: тип зони - "Основна", динамічні оновлення - встановити, решта параметрів - за замовчанням)
4. У полі "Код мережі (ID)" введемо параметри ідентифікатора мережі - 192.168.0.
5. Виконаємо команду примусової реєстрації клієнта на сервері DNS - ipconfig /registerdns.

Наші сервери реєструватимуться в зворотній зоні DNS (рис. 13):



**Рисунок - 13.**

**3 Діагностичні утиліти TCP/IP і DNS.**

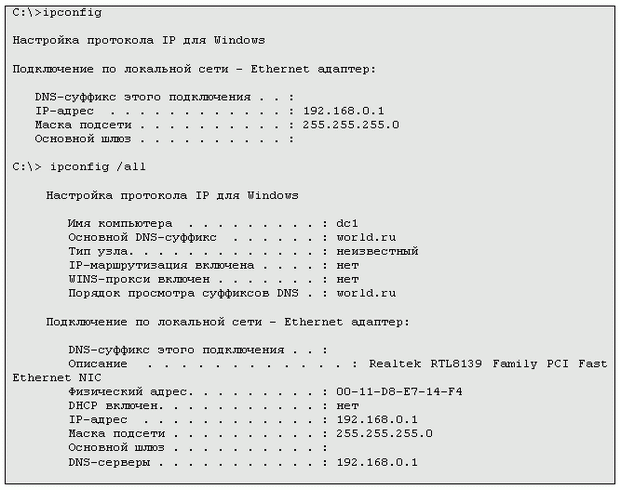
Будь-яка операційна система має набір діагностичних утиліт для тестування мережевих настройок і функціонування комунікацій. Великий набір діагностичних засобів є і в системах сімейства Windows (як графічних, так і режимі командного рядка).

Перерахуємо утиліти командного рядка, що є інструментами першої необхідності для перевірки настройок протоколу TCP/IP і роботи мереж і комунікацій. Докладний опис даних утиліт міститься в системі інтерактивної допомоги Windows. У Таблиці 6 вкажемо основні і найбільш часто використовувані параметри цих команд і дамо їх короткий опис.

|  |  |  |
| --- | --- | --- |
| **Таблиця 6.** | | |
| **Назва утиліти** | **Параметри** | **Коментарі** |
| ipconfig | /? - Відобразити довідку по команді  /all - Відобразити повну інформацію про настройку параметрів всіх адаптерів  /release - Звільнити динамічну IP-конфигурацію  /renew - Відновити динамічну IP-конфигурацію з DHCP-сервера  /flushdns - Очистити кеш імен DNS  /registerdns - Відновити реєстрацію на DNS-сервері  /displaydns - Відобразити вміст кеша імен DNS | Служить для відображення всіх поточних параметрів мережі TCP/IP і оновлення параметрів DHCP і DNS. При виклику команди ipconfig без параметрів виводяться IP-адреса, маска підмережі і основний шлюз для кожного мережевого адаптера. |
| arp | -а - Відображає поточні ARP-записи | Відображення і зміна ARP-таблиць. |
| ping | Формат команди:  "ping <мережевий вузол> параметри"  Параметри:  -t - Нескінченна (до натиснення клавіш <Ctrl>+<Break>) відправка пакетів на вказаний вузол  -а - Визначення імені вузла по IP-адресу  -n <число> - Число запитів, що відправляються  -l <розмір> - Розмір ПАКЕТУ відправки  -w <таймаут> - Таймаут очікування кожної відповіді в мілісекундах | Могутній інструмент діагностики (за допомогою протоколу ICMP).  Команда ping дозволяє перевірити:   * працездатність IP-з’єднання; * правильність налаштування протоколу TCP/IP на вузлі; * працездатність маршрутизаторів; * працездатність системи з’ясування імен FQDN або NETBIOS; * доступність і працездатність якого-небудь мережевого ресурсу. |
| tracert | -d - Без визначення IP-адреси в ім’я вузла  -h <макс.Число> - Максимальне число стрибків при пошуку вузла  -w <таймаут> - Таймаут кожної відповіді в мілісекундах | Службова програма для трасування маршрутів, використовується для визначення шляху, по якому IP-дейтаграмма доставляється за місцем призначення. |
| pathping | -n - Без визначення IP-адреси в ім’я вузла  -h макс.Число - Максимальне число стрибків при пошуку вузла  -q <число\_запитів> - Число запитів при кожному стрибку  -w <таймаут> - Таймаут кожної відповіді в мілісекундах | Засіб трасування маршруту, що поєднує функції програм ping і tracert і що володіє додатковими можливостями.  Ця команда показує ступінь втрати пакетів на будь-якому маршрутизаторі або каналі, з її допомогою легко визначити, які маршрутизатори або канали викликають несправності в роботі мережі. |
| netstat | -а - Відображення всіх підключень і підключень що слухають порти  -n - Відображення адрес і номерів портів в числовому форматі  - nро - Відображення кодів (ID) процесу кожного підключення  -r - Відображення вмісту локальної таблиці маршрутів | Використовується для відображення статистики протоколу і поточних TCP/IP-з’єднань. |
| nbtstat | –n - Виводить імена і зони імен NETBIOS, зареєстровані локальними процесами  –c - Відображає кеш імен NETBIOS (імена NetBIOS в IP-адреси)  –R - Очищає кеш імен і перезавантажує його з файлу Lmhosts  –RR - Звільняє імена NETBIOS, зареєстровані на WINS-сервері, а потім оновлює їх реєстрацію | Засіб діагностики розіменовки імен NETBIOS |

Приклади використання утиліт командного рядка.

**Приклад 1**. Команда ipconfig (без параметрів і з параметром /all).



**Рисунок - 14.**

**Приклад 2**. Команда arp.

Оскільки в наший мережі тільки два вузли, то в кеші сервера DC1 буде тільки один запис - відображення IP-адреси сервера DC2 на MAC-адресу мережевого адаптера.

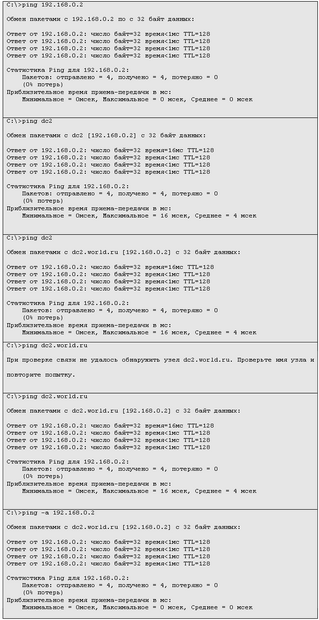


**Рисунок - 15.**

**Приклад 3**. Команда ping.

Варіанти використання:

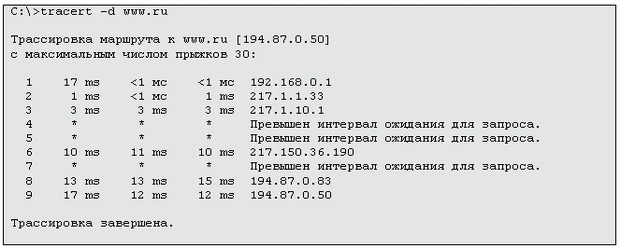
* ping <IP-адреса>;
* ping <NetBIOS-ім’я вузла>, коли в зоні сервера DNS немає запису для сервера DC2 (пошук IP-адреси проводиться широкомовним запитом );
* ping <NetBIOS-ім’я вузла>, коли в зоні сервера DNS є запис для сервера DC2 (треба звернути увагу на підстановку клієнтом DNS суфікса домена в запиті на розіменовку вузола, т.е в команді використовується коротке NetBIOS-ім’я сервера, а в статистиці команди виводиться повне ім'я);
* ping <FQDN-имя вузла>, коли в зоні сервера DNS немає запису для сервера DC2 (вузол DC2 не буде знайденро в мережі);
* ping <FQDN-имя вузла>, коли в зоні сервера DNS є запис для сервера DC2 (вузол успішно знайдений);
* ping – a <IP-адреса> (зворотня розіменовка IP-адреси в ім'я вузла)



**Рисунок - 16.**

**Приклад 4**. Команда tracert.

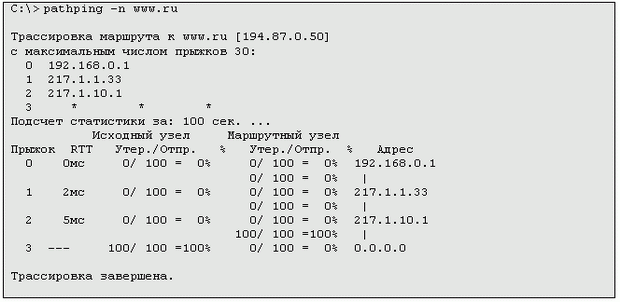
Трасування маршруту до вузла www.ua (якщо у вашому розпорядженні тільки одна IP-мережа, то вивчити роботу даної команди буде неможливо).



**Рисунок - 17.**

**Приклад 5**. Команда pathping.

Аналогічне завдання (трасування маршруту до вузла www.ua), виконане командою pathping.



**Рисунок - 18.**

**Приклад 6**. Команда netstat (з параметрами – an - відображення в числовій формі списку активних підключень і слухаючих портів).

# Алгоритми роботи ітеративних і рекурсивних запитів DNS

Усі запити, Що відправляються DNS-клієнтом DNS-сервера для розпізнавання імен, діляться на два типи:

* ітеративні запити (клієнт посилає серверу DNS запит, В якому вимагає дати найкращу відповідь без звернень до інших DNS-серверів);
* рекурсивні запити (клієнт посилає серверу DNS запит, в якому вимагає дати остаточну відповідь навіть якщо DNS-сервера доведеться відправити запити інших DNS-серверів; посилаються в цьому випадку іншим DNS-серверів запити будуть ітеративними).

Звичайні DNS-клієнти (наприклад, робочі станції користувачів), як правило, посилають рекурсивні запити.

Розглянемо на прикладах, як відбувається взаємодія DNS-клієнта і DNS-сервера при обробці ітеративних і рекурсивних запитів.

Припустимо, що користувач запустив програму Оглядач Інтернету і ввів в адресному рядку адресу http://www.microsoft.com. Перш ніж Оглядач встановить сеанс зв'язку з веб-сайтом по протоколу HTTP, клієнтський комп'ютер повинен визначити IP-адресу веб-сервера. Для цього клієнтська частина протоколу TCP / IP робочої станції користувача (так званий resolver) Спочатку переглядає свій локальний кеш дозволених раніше імен в спробі знайти там ім'яwww.microsoft.com. Якщо ім'я не знайдене, то клієнт посилає запит DNS-сервера, вказаною в конфігурації TCP / IP на вашому комп'ютері (назвемо цей DNS-сервер "Локальним DNS-сервером"), На дозвіл імені www.microsoft.com в IP-адреса цього сайту. Далі DNS-сервер обробляє запит в залежності від типу запиту.

Варіант 1 (ітеративний запит).

Якщо клієнт відправив сервера ітеративний запит (нагадаємо, що зазвичай клієнти посилають рекурсивні запити), то обробка запиту відбувається за такою схемою:

* спочатку локальний DNS-сервер шукає серед зон, за які він відповідає, зону microsoft.com;

якщо така зона знайдена, то в ній шукається запис для вузла www; якщо запис знайдений, то результат пошуку відразу ж повертається клієнтові;

в іншому випадку локальний DNS-сервер шукає запитане ім'я www.microsoft.com в своєму кеші дозволених раніше DNS-запитів;

якщо шукане ім'я є в кеші, то результат пошуку повертається клієнтові; якщо локальний DNS-сервер не знайшов у своїй базі даних шукану запис, то клієнту надсилається IP-адреса одного з кореневих серверів DNS;

* клієнт отримує IP-адреса кореневого сервера і повторює йому запит на дозвіл імені www.microsoft.com;

кореневий сервер не містить у своїй БД зони "microsoft.com", але йому відомі DNS-сервери, що відповідають за зону "com", і кореневий сервер посилає клієнтові IP-адреса одного з серверів, які відповідають за цю зону;

* клієнт отримує IP-адресу сервера, що відповідає за зону "com", і посилає йому запит на дозвіл імені www.microsoft.com;

сервер, який відповідає за зону com, не містить у своїй БД зони microsoft.com, Але йому відомі DNS-сервери, що відповідають за зону microsoft.com, І даний DNS-сервер посилає клієнтові IP-адреса одного з серверів, що відповідають вже за зону microsoft.com;

* клієнт отримує IP-адресу сервера, що відповідає за зону microsoft.com, І посилає йому запит на дозвіл імені www.microsoft.com;

сервер, який відповідає за зону microsoft.com, Отримує даний запит, знаходить в своїй базі даних IP-адреса вузла www, розташованого в зоні microsoft.com, І посилає результат клієнту;

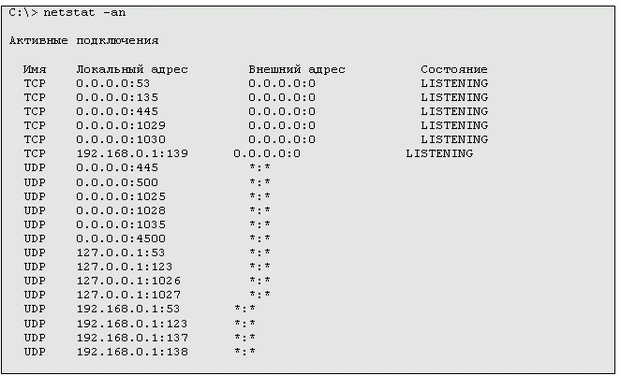
клієнт отримує шуканий IP-адресу, зберігає дозволений запит в своєму локальному кеші і передає IP-адреса веб-сайту програми Оглядач Інтернету (після чого Оглядач встановлює зв'язок з веб-сайтом по протоколу HTTP).

Варіант 2 (рекурсивний запит).

Якщо клієнт відправив сервера рекурсивний запит, То обробка запиту відбувається за такою схемою:

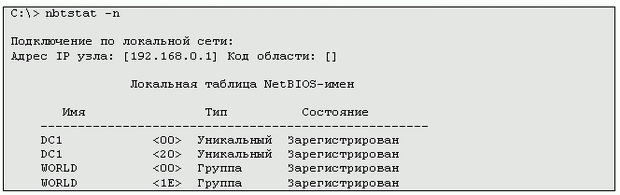
* спочатку локальний DNS-сервер шукає серед зон, за які він відповідає, зону microsoft.com; якщо така зона знайдена, то в ній шукається запис для вузла www; якщо запис знайдений, то результат пошуку відразу ж повертається клієнтові;

в іншому випадку локальний DNS-сервер шукає запитане ім'я www.microsoft.com в своєму кеші дозволених раніше DNS-запитів; якщо шукане ім'я є в кеші, то результат пошуку повертається клієнтові;



**Рисунок - 19.**

**Приклад 7**. Команда nbtstat (з параметром – n - відображення локальних імен NETBIOS).



**Рисунок - 20.**