# Лекція 3.

**Тема.**
**Мережеве забезпечення глобальних систем обробки та обміну інформацією.**

**Розділ. Ідентифікація вузлів глобальних систем обробки та обміну інформацією на базі системи доменних імен (DNS).**

План лекції:
3.1. Доменні імена. Система доменних імен.
3.2. Види доменних імен.
3.3. Головні характеристики DNS.
3.4. Ключові поняття DNS.
3.5. Рекурсія.
3.6. Зворотній DNS – запит.
3.7. Записи DNS.
3.8. Програмне забезпечення DNS серверів
3.9. Структура вузла з використанням IP адреси та доменного імені. Приклад побудови.

3.1. DNS (англ. Domen Name System – система доменних імен) система для ідентифікації вузлів та мереж у розподіленій комп'ютерній мережі, що об'єднуються у домени. Поняття Домен в системі OSI визначається як зона відповідальності у розподіленій системі DNS.

**Доменне ім'я** - символьне ім'я, що служить для ідентифікації областей ( одиниць адміністративної автономії в мережі Інтернет) у складі вищестоящої по ієрархії області. Кожна з таких областей називається доменом. Загальний простір імен Інтернету функціонує завдяки DNS - системі доменних імен. Доменні імена дають можливість ідентифікації та адресації інтернет-вузлів і розташованих на них мережевих ресурсів (веб-сайтів, серверів електронної пошти, інших служб) в зручній для людини текстовій формі.
Повне доменне ім'я складається з розділених крапкою безпосереднього імені домену та імен всіх доменів, в які він входить. Наприклад, повне ім'я **ns.sethost.net** означає домен третього рівня ns, який входить в домен другого рівня **sethost**, який входить в домен верхнього рівня **net**, який входить в безіменний кореневий домен. У повсякденній промові під доменним ім'ям нерідко розуміють саме повне доменне ім'я.
В затвердженій стандартом системі доменних імен дозволені тільки 26 символів латинського алфавіту, цифри від 0 до 9 і дефіс. В якості базисних символів виступають символи латинського алфавіту від a до z (без

відмінності між великими і малими літерами), цифри від 0 до 9 і дефіс «-»; всього 37 символів.

**Доменна зона** - сукупність доменних імен певного рівня, що входять в конкретний домен. Наприклад, зона **sethost.net** включає всі доменні імена третього рівня в цьому домені, наприклад, **ns.sethost.net**, **ftp.sethost.net**, **www.sethost.net**, **news.sethost.net** і т.д. . Термін «**доменна зона**» застосовується в технічній сфері, при налагодженні DNS-серверів (серверів підтримка зони, делегування зони, трансфер зони).

DNS служить для перетворення доменного імені в IP-адресу і навпаки. Ця система складається з ієрархічної структури DNS-серверів, кожен з яких є утримувачем однієї або декількох доменних зон і відповідає на запити, що стосуються цих зон. Кажуть, що DNS сервер є **відповідальним** за зони, які він утримує. DNS сервер також виконує функцію резолверів (resolver), які відповідають на запити, що стосуються будь-яких зон. Функції утримувача зони і резолвера часто поєднуються в одній програмі. Наприклад, такою є популярний DNS-сервер BIND (Berkeley Internet Name Domain) .

Для забезпечення унікальності і захисту прав власників доменні імена 1-го і 2-го (в окремих випадках і 3-го) рівнів можна використовувати тільки після їх реєстрації, яка проводиться уповноваженими на це реєстраторами. Відомості про власника (адміністратора) того чи іншого зареєстрованого домена загальнодоступні. Їх можна отримати, скориставшись службою WHOIS. Однак, деякі реєстратори надають можливість приховати цю інформацію.

Кожний DNS сервер, відповідальний за ім'я, може **делегувати** відповідальність за якусь частину цього імені іншому серверу. Це дає змогу розділити відповідальність за актуальність інформації, що зберігається на серверах, на кілька організацій (людей), які відповідають тільки за свою частину доменного імені.

Малюнок.

.

com org net business ua ru

З розвитком інтернету особливу цінність набули "красиві" назви сайтів (доменів). Загальне число зареєстрованих доменів наближається до 200 мільйонів. Підібрати вільне, красиве і коротке доменне ім'я стало дуже важко. Утворився ринок перепродажу доменних імен. Сюди входять компанії, які реєструють домени, купують і продають домени на вторинному ринку. Наприклад, домен business.com був проданий за 360 мільйонів доларів США тому, що тисячі компаній хотіли б мати свій офіційний сайт на домені business.com.

3.2. Види доменних імен

**3.2.1. gTLD** (англ. *generic Top-Level Domain* — Загальний домен верхнього рівня.

**Приклади.**

Не спонсоруються.
.com (commercial) — для комерційних організацій
.net (networks) — для мережевих структур
.org (organizations)— некомерційні організації
.biz (business organizations)— тільки комерційні організації.
.info (information) — домен відкритий для всіх
.name (personal) — для персональних сайтів
.pro (professionals) — для фахівців певних професій

Ті, що спонсоруються.
.int — загальний домен верхнього рівня для міжнародних організацій
.aero — організації та фізичні особи, так чи інакше пов'язані з аероіндустрією
.cat — призначений для представників каталонського лінгвістичного та культурного співтовариства.
.eco — для інтернет-ресурсів, пов'язаних з екологією
.jobs — домен для веб-сайтів з інформацією про затребувані професії і вакансії.

Обмеженого користування
.edu (educational) — для освітніх проектів та вищих навчальних закладів США
.gov (US Government) — зарезервований для уряду США.
.int (international organizations) — для міжнародних організацій
.mil (US Dept of Defense) — для військових організацій і установ США.

Домени для інфраструктури Інтернету
.arpa — для інфраструктури Інтернету
.root — домен прописаний в кореневих серверах DNS, контрольованих компанією VeriSign.

Зарезервовані домени
.example — зарезервовано для прикладів
.invalid — зарезервовано для того щоб уникнути конфліктів з традиційним використанням локального.
.localhost — зарезервовано для того щоб уникнути конфліктів з традиційним використанням localhost.
.test — зарезервовано для використання в тестах.

3.2.2. IDN (англ. *Internationalized Domain Names* — Інтернаціоналізовані доменні імена) — доменні імена, що містять символи національних алфавітів.

Для того, щоб не міняти структуру DNS, було запропоновано перетворювати імена, що містять символи національних алфавітів, в слова, які складаються тільки з допустимих раніше символів ASCII, причому

робити це в клієнтських додатках. Таким чином, для підтримки IDN достатньо, щоб їх розумів браузер користувача. Він повинен уміти переводити їх в символьне кодування **Punycode**, що дозволяє представити будь-які символи Unicode за допомогою дозволеного раніше набору символів ASCII (**http://tools.ietf.org/html/rfc3492**). Щоб у такому вигляді IDN було неможливо сплутати зі звичайними доменними іменами, вони починаються зі спеціального префікса **«xn--»**, наприклад, **«xn--abc.com»** - IDN в Punycode-вигляді, а «abc.com» - звичайне доменне ім'я. Фактично інтернаціоналізованні доменні імена є псевдонімами для імен, що починаються з **«xn--»**. Рядок **«xn--e1afmkfd.xn--80akhbyknj4f»** демонструє Punycode-вигляд для реально існуючого IDN «пример.испытание»: http://пример.испытание/. У базі DNS-сервера зберігається тільки Punycode-вигляд, а в браузері можна вводити як те, так і інше. У браузерах, що не підтримують IDN, вдасться використати тільки Punycode-вигляд IDN.

3.2.3. Національні домени (ccTLD).

Приклади.
.ua
.us
.ru

3.2.4. Зарезервовані доменні імена - RFC 2606 (Reserved Top Level DNS Names — Зарезервовані доменні імена верхнього рівня)

**3.2.5. Найдорожчі доменні імена.**
Insure.com за $16 млн. 2009
Sex.com за $14 млн. 2010
Fund.com за £9.99 млн. 2008
Porn.com за $9.5 млн. 2007
Fb.com за $8.5 млн. 2010

## 3.3. Головні характеристики DNS.

**Розподіленість адміністрування**. Відповідальність за різні частини ієрархічної структури несуть різні люди або організації.

**Розподіленість зберігання інформації**. Кожен вузол мережі в обов'язковому порядку повинен зберігати тільки ті дані, які входять в його зону відповідальності та (можливо) адреси кореневих DNS-серверів.

**Кешування інформації**. Вузол може зберігати деяку кількість даних не своєї зони відповідальності для зменшення навантаження на мережу.

**Ієрархічна структура**, в якій всі вузли об'єднані в дерево і кожен вузол може або самостійно визначати роботу нижчестоящих вузлів, або делегувати їх іншим вузлам.

**Резервування**. За зберігання та обслуговування своїх зон відповідають як правило кілька серверів, розділені як фізично, так і логічно, що забезпечує збереження даних і продовження роботи у разі збою одного з вузлів.

DNS важлива для роботи Інтернету, так як для з'єднання з вузлом необхідна інформація про його IP-адресу, а для людей простіше запам'ятовувати буквені (зазвичай осмислені) назви, ніж послідовність цифр IP-адреси. У деяких випадках це дозволяє використовувати віртуальні сервери, наприклад, HTTP-сервери, розрізняючи їх по імені запиту. DNS була розроблена Полом Мокапетрісом в 1983 році;

### 3.4. Ключові поняття DNS.

**Домен** (англ. область - область) - вузол в дереві імен разом з усіма підпорядкованими йому вузлами, тобто - іменована гілка або піддерево в дереві імен, наприклад **ns.sethost.net**. Структура доменного імені відображає порядок проходження вузлів в ієрархії. Доменне ім'я читається зліва направо від молодших доменів до доменів вищого рівня (у порядку підвищення значущості). Кореневим доменом всієї системи є крапка ('.'), Нижче йдуть домени першого рівня (географічні або тематичні ), потім - домени другого рівня, третього і т. д. На практиці крапку в кінці імені часто опускають, але вона буває важлива у випадках розподілу між відносними доменами і FQDN (англ. Fully Qualifed Name - повністю визначене ім'я домену).

**Піддомен** (англ. субдомен) - підлеглий домен (наприклад, sethost.net - піддомен домена .net, а ns.sethost.net - домену sethost.net). Теоретично такий розподіл може досягати глибини 127 рівнів, а кожна мітка може містити до 63 символів, поки загальна довжина разом з точками не досягне 254 символів. Але на практиці реєстратори доменних імен використовують більш суворі обмеження. Наприклад, якщо у вас є домен виду mydomain.ru, ви можете створити для нього різні піддомени виду mysite1.mydomain.ru, mysite2.mydomain.ru і т. д.

**Ресурсний запис** - одиниця зберігання і передачі інформації в DNS. Кожний ресурсний запис має ім'я (тобто він прив'язаний до певного доменного імені, вузла в дереві імен), тип і поле даних, формат і зміст якого залежить від типу.

**Зона** - частина дерева доменних імен (включаючи ресурсні записи), що розміщується як єдине ціле на деякому сервері доменних імен (DNS-сервері). А частіше - одночасно на декількох серверах. Метою виділення частини дерева в окрему зону є передача відповідальності за відповідний домен іншій особі або організації. Це називається делегуванням. Як зв'язкова частина дерева, зона всередині теж представляє собою дерево.

**Делегування** - операція передачі відповідальності за частину дерева доменних імен іншій особі або організації. За рахунок делегування в DNS забезпечується розподіленість адміністрування та зберігання. Технічно делегування виражається у виділенні частини дерева в окрему зону і

розміщенні цієї зони на DNS-сервері, що керується іншою особою чи організацією. При цьому в батьківську зону включаються «склеюючі» ресурсні записи (NS і A), що містять покажчики на DNS-сервери дочірньої зони, а вся інша інформація, що відноситься до дочірньої зоні, зберігається на DNS-серверах дочірньої зони.

**DNS-сервер** - спеціалізоване ПО для обслуговування DNS, а також комп'ютер, на якому це ПЗ виконується. DNS-сервер може бути відповідальним за деякі зони та / або може перенаправляти запити вищестоящим серверам.

**DNS-клієнт** - спеціалізована бібліотека (або програма) для роботи з DNS. У ряді випадків DNS-сервер виступає і в ролі DNS-клієнта.

**Авторитетність** (англ. *authoritative*) - ознака розміщення зони на DNS-сервері. Відповіді DNS-сервера 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 попередніми запитами) значення (режим кешування).

**DNS-запит** (англ. *DNS query*) - запит від клієнта сервера. Запит може бути рекурсивним або нерекурсивним.

Система DNS містить ієрархію DNS-серверів що відповідає ієрархії зон. Кожна зона підтримується як мінімум одним авторитетним сервером DNS, на якому розташована інформація про домен.

Ім'я та IP-адреса не тотожні - одна IP-адреса може мати безліч імен, що дозволяє підтримувати на одному комп'ютері безліч веб-сайтів (це називається віртуальний хостинг). Зворотнє теж справедливо - одному імені може бути зіставлено безліч IP-адрес. Це дозволяє створювати балансування навантаження.

Для підвищення стійкості системи використовується велика кількість серверів, що містять ідентичну інформацію, а в протоколі є засоби, що дозволяють підтримувати синхронність інформації, розташованої на різних серверах. Існує 13 кореневих серверів (**http://www.root-servers.org**), їх адреси практично не змінюються. Протокол DNS використовує для роботи TCP-або UDP-порт 53 для відповідей на запити. Традиційно запити та відповіді відправляються у вигляді однієї UDP датаграми. TCP використовується для AXFR-запитів.

### 3.5. Рекурсія.

Терміном Рекурсія в DNS позначають алгоритм поведінки DNS-сервера, при якому сервер виконує від імені клієнта повний пошук потрібної інформації у всій системі DNS, при необхідності звертаючись до інших DNS-серверів.

DNS-запит може бути рекурсивним - вимагає повного пошуку, - і

нерекурсівним (або ітеративним) - не вимагає повного пошуку.

Аналогічно, DNS-сервер може бути рекурсивним (який вміє виконувати повний пошук) і нерекурсивним (які не вміють виконувати повний пошук). Деякі програми DNS-серверів, наприклад, BIND, можна конфігурувати так, щоб запити одних клієнтів виконувалися рекурсивно, а запити інших - нерекурсивно.

При відповіді на нерекурсивний запит, а також - при невмінні або заборону виконувати рекурсивні запити, - DNS-сервер або повертає дані про зону, за яку він відповідальний, або повертає адреси серверів, які володіють більшим обсягом інформації про запитану зону, ніж відповідаючий сервер, найчастіше - адреси кореневих серверів.

У разі рекурсивного запиту DNS-сервер опитує сервери (в порядку убування рівня зон в імені), поки не знайде відповідь або не виявить, що домен не існує. Розглянемо на прикладі роботу всієї системи.

Припустимо, ми набрали в браузері адресу ns.sethost.net. Браузер запитує у сервера DNS: «яка IP-адреса у ns.sethost.net»? Однак, сервер DNS може нічого не знати не тільки про запрошене ім’я, але навіть про весь домен sethost.net. У цьому випадку сервер звертається до кореневого сервера - наприклад, 198.41.0.4. Цей сервер повідомляє - «У мене немає інформації про дане ім’я, але я знаю, що 204.74.112.1 є відповідальним за зону .net» Тоді сервер DNS направляє свій запит до 204.74.112.1, але той відповідає «У мене немає інформації про дане ім’я , але я знаю, що 207.142.131.234 є відповідальним за зону sethost.net » Нарешті, той самий запит відправляється до 207.142.131.234 і отримує відповідь - IP-адреса, яка і передається клієнтові - браузеру.

В даному випадку при розділенні імені, тобто в процесі пошуку IP по імені:

1. Браузер відправив відомому йому DNS-серверу рекурсивний запит - у відповідь на такий тип запиту сервер зобов'язаний повернути «готовий результат», тобто IP-адресу, або порожню відповідь і код помилки NXDOMAIN.

2. DNS-сервер, що отримав запит від браузера, послідовно відправляв нерекурсивні запити, на які отримував від інших DNS-серверів відповіді, поки не отримав відповідь від сервера, відповідального за запитану зону.

3. Інші згадувані DNS-сервери обробляли запити нерекурсивно (і, швидше за все, не стали б обробляти запити рекурсивно, навіть якщо б така вимога стояла в запиті).

Іноді допускається, щоб запитаний сервер передавав рекурсивний запит «вищестоячому» DNS-серверу і чекав готової відповіді.

При рекурсивній обробці запитів всі відповіді проходять через DNS-сервер, і він отримує можливість кешувати їх. Повторний запит на ті ж імена зазвичай не йде далі кеша сервера, звернення до інших серверів не відбувається взагалі. Допустимий час зберігання відповідей в кеші приходить разом з відповідями (поле TTL ресурсного запису).

Рекурсивні запити вимагають більше ресурсів від сервера (і створюють більше трафіку), так що зазвичай приймаються від «відомих» власнику сервера вузлів (наприклад, провайдер надає можливість робити рекурсивні запити тільки своїм клієнтам, в корпоративній мережі рекурсивні запити приймаються тільки з локального сегмента). Нерекурсивні запити зазвичай приймаються від усіх вузлів мережі (і змістовна відповідь дається тільки на запити про зону, яка розміщена на вузлі, в DNS-запиті про інші зони зазвичай повертаються адреси інших серверів).

**3.6. Зворотній DNS-запит.** DNS використовується в першу чергу для перетворення символьних імен в IP-адреси, але він також може виконувати зворотній процес. Для цього використовуються вже наявні ресурси DNS. Справа в тому, що із записом DNS можуть бути зіставлені різні дані, в тому числі і будь-яке символьне ім'я. Існує спеціальний домен **in-addr.arpa**, записи в якому використовуються для перетворення IP-адрес в символьні імена. Наприклад, для отримання DNS-імені для адреси 77.47.136.36 можна запросити в DNS-сервері запис 36.136.47.77.in-addr.arpa, і той поверне відповідне символьне ім'я. Зворотний порядок запису частин IP-адреси пояснюється тим, що в IP-адресах старші біти розташовані на початку, а в символьних DNS-іменах старші (що знаходяться ближче до кореня) частини розташовані в кінці.

**3.7. Записи DNS.** Записи DNS, або Ресурсні записи (англ. Resource Records, RR) - одиниці зберігання і передачі інформації в DNS. Кожний ресурсний запис складається з наступних полів:

ім'я (**NAME**) - доменне ім'я, до якого прив'язана або якому «належить» даний ресурсний запис,

**TTL** (Time To Live) - допустимий час зберігання даного ресурсного запису в кеші невідповідального DNS-сервера,

тип (**TYPE**) ресурсного запису - визначає формат і призначення даного ресурсного запису,

клас (**CLASS**) ресурсного запису, теоретично вважається, що DNS може використовуватися не тільки з TCP / IP, але і з іншими типами мереж, код в поле клас визначає тип мережі,

довжина поля даних (**RDLEN**),

поле даних (**RDATA**), формат та зміст якого залежить від типу запису.

Найбільш важливі типи DNS-записів:

Запис **A** (*address record*) зв'язує ім'я хоста з адресою IP. Наприклад, запит A-запису на ім'я ns.sethost.net поверне його IP-адресу - 188.40.85.133

Запис **AAAA** (IPv6-адресу запису) зв'язує ім'я хоста з адресою протоколу IPv6. Наприклад, запит AAAA-запису на ім'я K.ROOT-SERVERS.NET поверне його IPv6 адресу - 2001:7 FD :: 1

Запис **CNAME** (канонічне ім'я запису) або канонічний запис імені (псевдонім) використовується для перенаправлення на інше ім'я

Запис **MX** (Mail Exchange) вказує на сервер (и) обміну поштою для

даного домену.

Запис **NS** (сервер імен) вказує на DNS-сервер для даного домену.

Запис **PTR** (покажчик) зв'язує IP хоста з його канонічним ім'ям. Запит в домені in-addr.arpa з IP хоста в зворотній формі поверне ім'я (FQDN) даного хоста. Наприклад, для IP адреси 192.0.34.164: запит запису PTR 164.34.0.192.in-addr.arpa поверне його канонічне ім'я referrals.icann.org. З метою зменшення обсягу спаму багато поштових серверів перевіряють наявність PTR запису для хоста, з якого відбувається відправлення. У цьому випадку PTR запис для IP адреси повинен відповідати імені поштового сервера, яким він представляється в процесі SMTP-сесії.

Запис **SOA** (Start Of Authority) або початковий запис зони вказує, на якому сервері зберігається еталонна інформація про даний домен, містить контактну інформацію особи, що відповідає за дану зону, таймінги кешування зонної інформації та взаємодії DNS-серверів.

**SRV**-запис (Вибір сервера) вказує на сервери для сервісів. Використовується, зокрема, для Jabber і Active Directory.

### 3.8. Програмне забезпечення DNS серверів.

BIND (Berkeley Internet Name Domain)
djbdns (Daniel J. Bernstein's DNS)
MaraDNS
NSD (Name Server Daemon)
PowerDNS
OpenDNS
Microsoft DNS Server
MyDNS

### 3.10. Структура вузла з використанням IP адреси та доменного імені. Приклад побудови.

**ns сервер ns.sethost.net, ns.itci.net**
**IP 77.47.136.37**
**Мережа 77.47.136.32/29**

Пряма зона:

```
; BIND version named 4.9.4-P1 Thu Apr 24 21:17:33 EEST 1997
; BIND version root@itc.ipri.kiev.ua:/usr/obj/usr/src/libexec/named-xfer
; zone 'itci.net'   last serial 20000122202
; from 62.244.54.211   at Fri Jul 17 11:30:34 1998
$TTL    86400   ;1 day
@            IN      SOA     ns.itci.kiev.ua. samj.itci.net. (
                2012012601 28800 3600 720000 86400 )
```

```
                IN      NS      ns.itci.net.
                IN      NS      ns.sethost.net.
                IN      MX      20 ns.sethost.net.
                IN      A       77.47.136.37
ns              IN      A       77.47.136.37
router          IN      A       77.47.136.33
buragas         IN      A       77.47.136.34
ip-phone        IN      A       77.47.136.35
bookkeep        IN      A       77.47.136.36
stas            IN      A       77.47.136.38
test            IN      A       77.47.136.39
ftp             IN      CNAME   ns.itci.net.
www             IN      CNAME   ns.itci.net.
smtp            IN      CNAME   ns.itci.net.
hosting         IN      A       188.40.85.133
design          IN      A       188.40.85.133
```

Зворотня зона. Приклад безкласового делегування:

Зі сторони ns сервера провайдера:

```
$ORIGIN 136.47.77.in-addr.arpa.
 @      IN      SOA     my-ns.my.domain. hostmaster.my.domain. (...)
 ;...
 ;  32-39 /29
 32-39          NS      ns.sethost.net.
 32-39          NS      ns.itci.net.
 ;
 33             CNAME   33.32-39.136.47.77.in-addr.arpa.
 34             CNAME   34.0/25.2.0.192.in-addr.arpa.
 35             CNAME   35.0/25.2.0.192.in-addr.arpa.
 36             CNAME   36.0/25.2.0.192.in-addr.arpa.
 37             CNAME   37.0/25.2.0.192.in-addr.arpa.
 38             CNAME   38.0/25.2.0.192.in-addr.arpa.
 39             CNAME   39.0/25.2.0.192.in-addr.arpa.
```

Зі сторони ns.sethost.net сервера:

```
; BIND version named 4.9.4-P1 Thu Apr 24 21:17:33 EEST 1997
; BIND version root@itc.net:/usr/obj/usr/src/libexec/named-xfer
; zone '32-39.136.47.77.IN-ADDR.ARPA'   last serial 2000112901
; from 194.44.146.211   at Mon August  30 12:10:52 2012
$TTL    86400   ;1 day
$ORIGIN 32-39.136.47.77.IN-ADDR.ARPA.
```

```
@       IN      SOA     itci.net. samj.itci.net. (
                        2012030906 28800 1800 720000 86400 )
                NS      ns.itci.net.
                NS      ns.sethost.net.
33      IN      PTR     router.itci.net.
34      IN      PTR     buragas.itci.net.
35      IN      PTR     ip-phone.itci.net.
36      IN      PTR     bookkeep.itci.net.
37      IN      PTR     ns.itci.net.
38      IN      PTR     stas.itci.net.
39      IN      PTR     test.itci.net.
```