

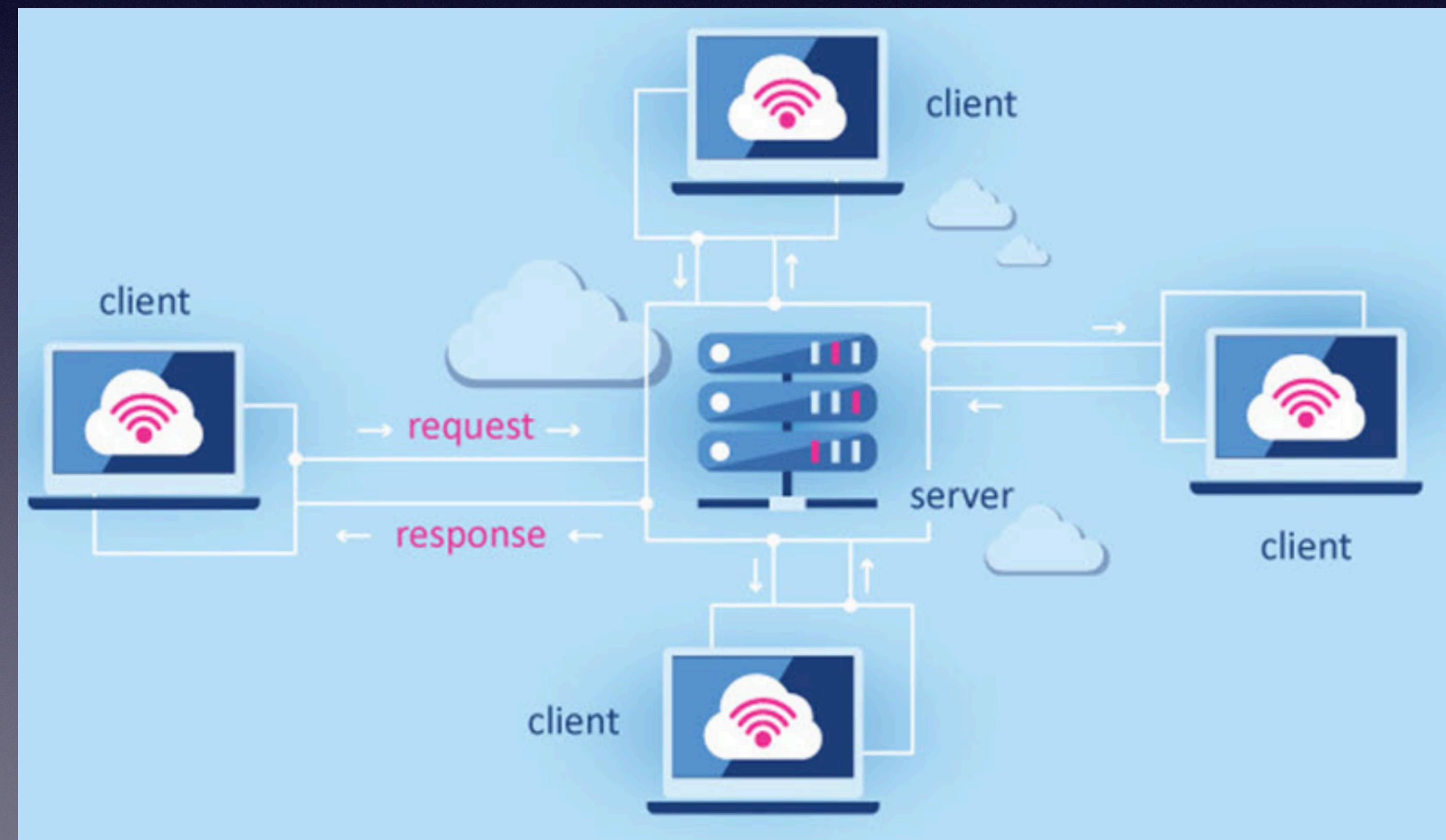
Основи веб-розробки та архітектура клієнт-сервер

Клієнт-серверна архітектура

Клієнт – комп'ютер на стороні користувача, який відправляє запит до сервера для надання інформації або виконання певних дій.

Сервер – більш потужний комп'ютер або обладнання, призначене для вирішення певних завдань з виконання програмних кодів, виконання сервісних функцій за запитом клієнтів, надання користувачам доступу до певних ресурсів, зберігання інформації і баз даних.

Модель такої системи полягає в тому, що клієнт відправляє запит на сервер, де він обробляється, і готовий результат відправляється клієнтові. Сервер може обслуговувати кілька клієнтів одночасно. Якщо одночасно приходить більше одного запиту, то вони встановлюються в чергу і виконуються сервером послідовно. Іноді запити можуть мати пріоритети. Запити з більш високими пріоритетами повинні виконуватися раніше.



Функції, які реалізуються на сервері:

- зберігання, доступ, захист і резервне копіювання даних;
- обробка клієнтського запиту;
- відправлення результату (відповіді) клієнту.

Функції, які реалізуються на стороні клієнта:

- надання користувальнику інтерфейсу;
- формування запиту до сервера і його відправка;
- отримання результатів запиту і відправка додаткових команд (запитів на додавання, оновлення або видалення даних).

Архітектура клієнт-сервер визначає принципи спілкування між комп'ютерами, а правила і взаємодії визначені в протоколі.

Мережевий протокол – це набір правил, за якими відбувається взаємодія між комп'ютерами в мережі.

Мережеві протоколи:

TCP/IP – набір (стек) протоколів передачі даних. TCP/IP – це позначення всієї мережі, яка працює на основі двох протоколів – TCP і IP.

TCP (Transfer Control Protocol) – протокол, який служить для встановлення надійного з'єднання між двома пристроями, передачі інформації і підтвердження її отримання.

IP (Internet Protocol) – інтернет протокол, який відповідає за правильність доставки повідомлень за певною адресою. При цьому дані розбиваються на пакети, які можуть доставлятися по-різному.

MAC (Media Access Control) – протокол, за допомогою якого відбувається ідентифікація мережевих пристрій. Всі пристрої, підключенні до інтернету, мають свою унікальну MAC адресу.

ICMP (Internet control message protocol) – протокол, який відповідає за обмін інформацією, але не використовується для передачі даних.

UDP (User datagram protocol) – протокол, який керує передачею інформації, але інформація не проходить перевірку при отриманні. Даний протокол працює швидше, ніж TCP.

HTTP (Hyper Text Transfer Protocol) – протокол передачі гіпертексту, на основі якого працюють всі сайти. Він запитує необхідні дані у віддаленій системі (веб-сторінки, файли).

FTP (File Transfer Protocol) – протокол передачі файлів зі спеціального файлового сервера на комп'ютер користувача.

SSH (Secure Shell) – протокол, який служить для забезпечення віддаленого керування системою по захищенному каналу.

POP3 (Post Office Protocol) – стандартний протокол поштового з'єднання, який відповідає за доставку пошти.

SMTP (Simple Mail Transfer Protocol) – протокол, який визначає правила для передачі пошти. Відповідає за повернення або підтвердження про доставку, оповіщення про помилку.

Існують концепції побудови системи клієнт-сервер:

Слабкий клієнт – потужний сервер. У такій моделі вся обробка інформації перенесена на сервер, а у клієнта права доступу сурово обмежені. Сервер відповідає відповідь, яка не вимагає додаткової обробки. Клієнт взаємодіє з користувачем: складає та відправляє запит, приймає результат і виводить інформацію на екран.

Сильний клієнт – концепція, в якій частина обробки інформації надається клієнтові. У такому випадку сервер виступає сховищем даних, а вся робота по обробці та подання інформації переноситься на комп'ютер клієнта.

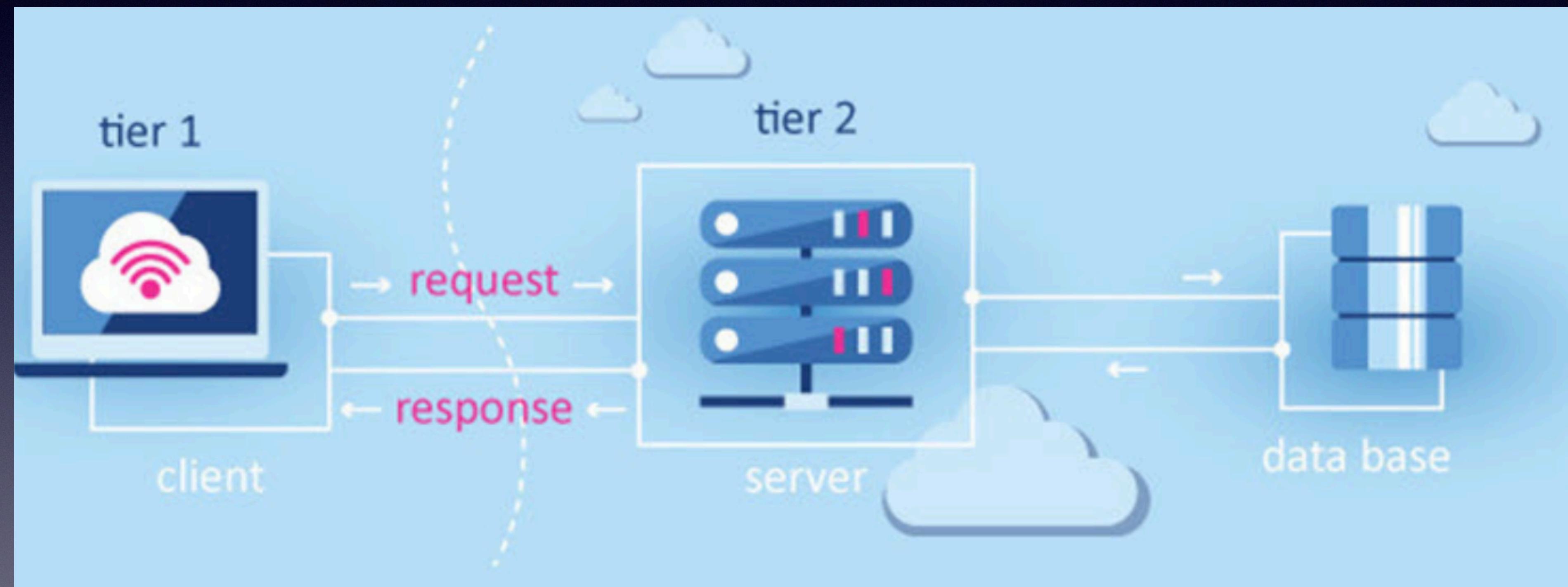
Система (додаток), яка заснована на клієнт-серверній взаємодії, включає три основних компоненти: представлення даних, прикладний компонент, компонент управління ресурсами і їх зберігання. Існують **дворівнева** і **трирівнева** клієнт-серверні архітектури.

Дворівнева архітектура складається з двох вузлів:

сервер, який відповідає за отримання запитів і відправку відповідей клієнту, використовуючи при цьому лише власні ресурси;

клієнт, який представляє користувальський інтерфейс.

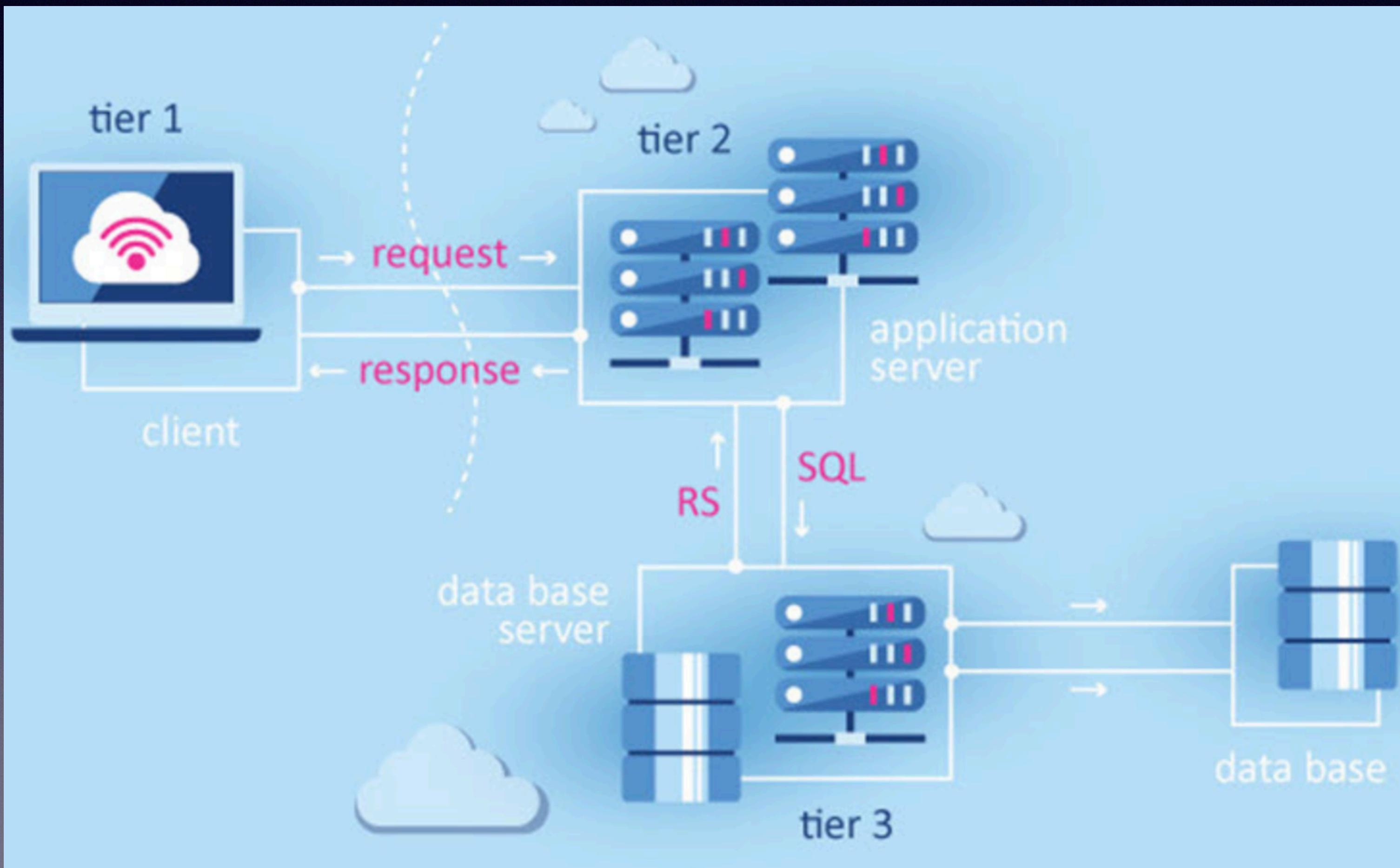
Принцип роботи полягає в тому, що сервер отримує запит, обробляє його і відповідає безпосередньо, без використання сторонніх ресурсів.



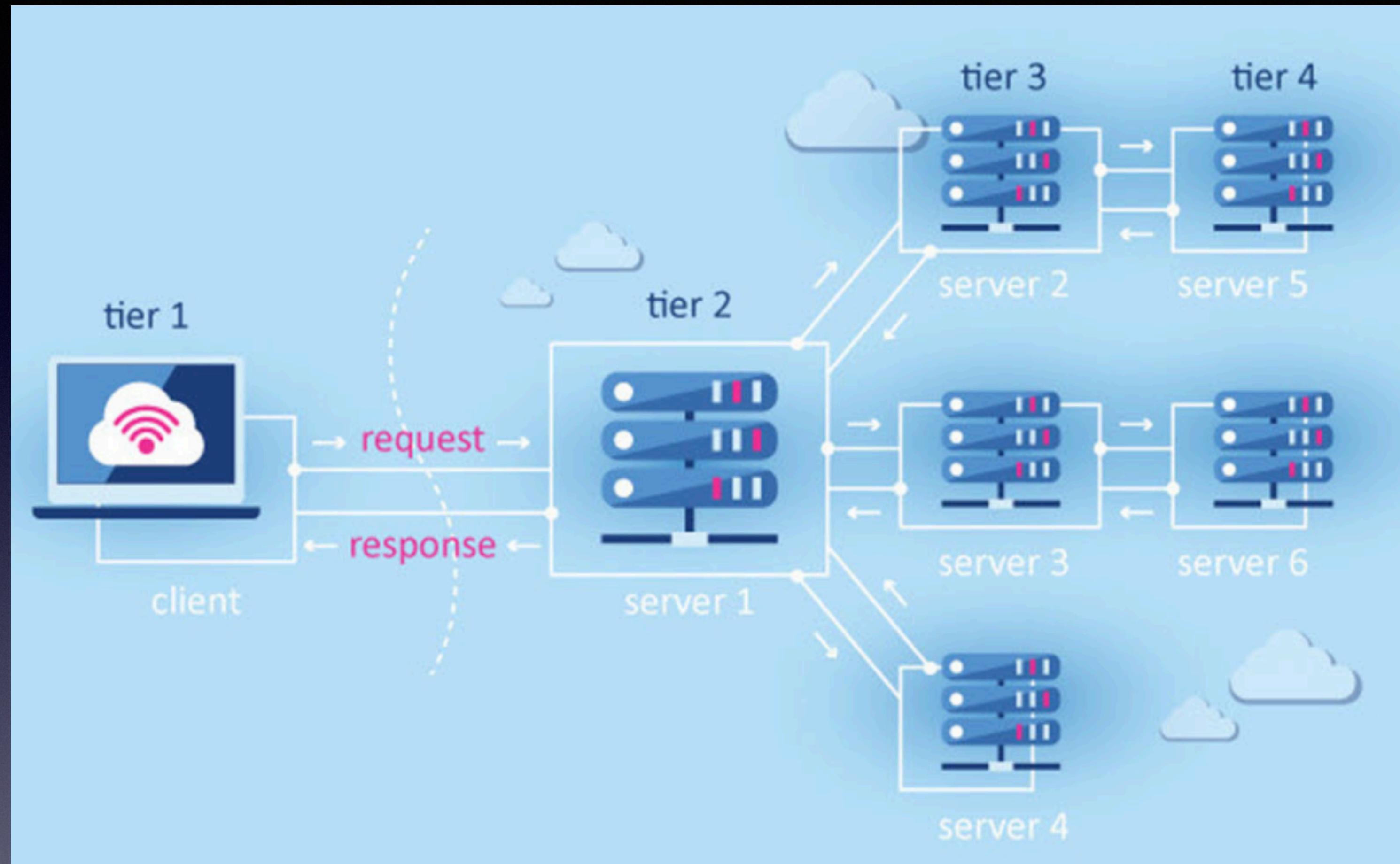
Трирівнева архітектура складається з наступних компонентів:

- представлення даних – призначений для користувача інтерфейс;
- прикладний компонент – сервер додатків;
- керування ресурсами – сервер бази даних, який надає інформацію.

Принцип роботи полягає в тому, що декілька серверів обробляють запит клієнта. Розподіл операцій знижує навантаження на сервер.



Трирівневу архітектуру можна розширити до **багаторівневої (N-tier, Multi-tier)** способом встановлення додаткових серверів. Багаторівнева архітектура дозволяє підвищити ефективність роботи інформаційної системи, а також оптимізувати розподіл її програмно-апаратних ресурсів.



Взаємодія клієнт-сервер дозволяє розділяти функціонал і обчислювальне навантаження між клієнтськими додатками (замовниками послуг) і серверними додатками (постачальниками послуг). Знання архітектури додатка дозволяє тестувальнику більш якісно провести функціональне, крос-браузерне тестування, тестування юзабіліті і швидкодії.

HTTP та HTTPS

HTTP — це протокол, в якому описано правила передачі даних в інтернеті. Він допомагає браузеру завантажувати вебсторінки, а серверу отримати інформацію, яку користувач ввів на сайті. **HTTPS** — це той самий протокол, але з надбудовою безпеки.

У чому різниця між HTTP та HTTPS

За HTTP інформація передається у звичайному вигляді, а за HTTPS — у зашифрованому. Шифрувати дані потрібно, щоб хакери не змогли нічого прочитати, якщо їх перехоплять.

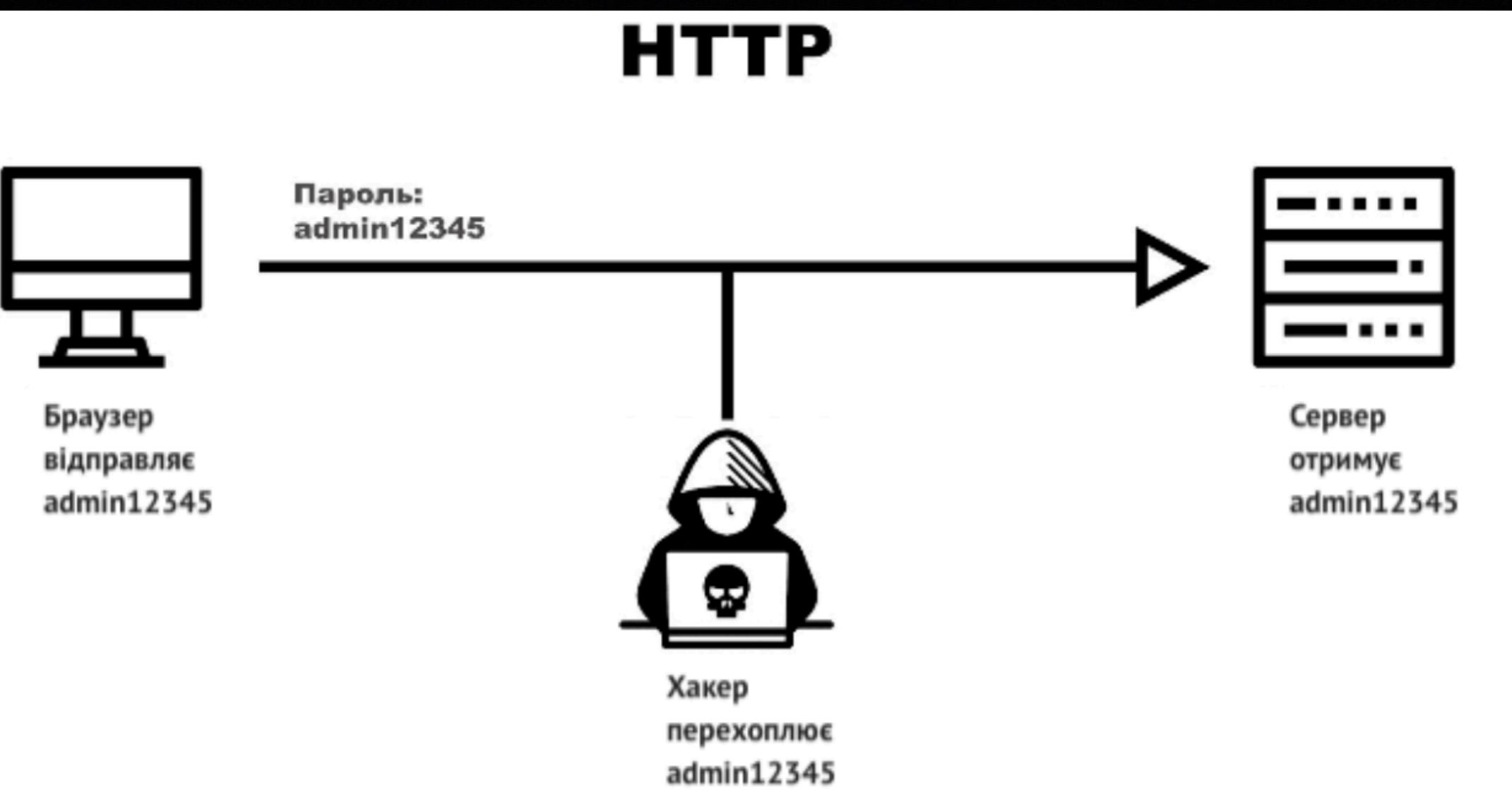
Допустимо, ви проходите опитування на сайті, який працює за HTTP-протоколом. Ось ви заповнили порожні поля та натиснули кнопку «Надіслати». Браузер надсилає ваші відповіді серверу. У цей момент хакер може перехопити інформацію та прочитати, що ви там відповідали. Ви навіть цього не помітите.

Імовірно, хакерів не цікавлять ваші відповіді на опитування. Але перехопити можна будь-яку інформацію. Наприклад, ваші паролі або номер банківської картки.

Щоб цього не сталося, HTTP-протокол вирішили вдосконалити. До існуючої технології додали шифрування і вийшов HTTPS — безпечний протокол передачі даних.

Коли ви вводите щось на сайті, що працює за HTTPS, перед відправкою даних на сервер браузер зашифрує інформацію. Щоб розшифрувати та прочитати її, потрібен спеціальний ключ, який зберігається лише на сервері. Таке шифрування називається криптографічним. Якщо навіть шахрай перехопить інформацію, він не зможе її прочитати. На те, щоб підібрати ключ до шифру, підуть роки безперервного перебору.

HTTP



HTTPS

