Лекція 8

Інформаційні технології електронного документообігу та електронних архівів

Low-code-технологія

Low-code-технологія – це підхід до створення, налаштування і модифікації систем та застосунків, який практично не вимагає написання програмного коду. На практиці такий підхід реалізується за допомогою low-code-платформ (low-code development platform (LCDP)) , тобто платформ, які забезпечують середовище розроблення, що використовується для створення прикладного програмного забезпечення через графічний користувацький інтерфейс замість традиційного ручного кодування в рамках комп'ютерного програмування. Вибір low-code-платформ для вирішення поставлених завдань насамперед пояснюється їх перевагами, зокрема модульністю, тобто можливістю будувати додатки (бізнес-процеси) з готових модулів; швидким прототипуванням (час від ідеї, що виникла у голові користувача, до реалізації проєкту скорочується до мінімуму); низьким порогом входу (не обов'язково бути кваліфікованим програмістом, щоб автоматизувати бізнес-процеси або створювати додатки); масштабованістю (за зміни бізнес-вимог або потреб компанії можна легко і швидко збільшити функціональні можливості власного рішення, доповнити новими автоматизованими процесами).

На думку експертів "Gartner", лідером у сегменті low-code-платформ є компанія "Microsoft". У 2020 році вона очолила список провідних розробників. Рішення, що пропонуються компанією "Microsoft", мають довгу історію перебування на ринку, тому включають кращі практики, зручність і функціональність яких перевірена часом. До складу low-code-продукту "Microsoft Power Platform" входять такі модулі: PowerApps, Power Automate, Power BI та Power Virtual Agent, використання яких здійснюється через хмарний сервіс "Microsoft365".

Призначення та функціональність модулів платформи Microsoft Power Platform (таблиця 1)

Таблиця 1

№	Назва модуля low-code-платформи Microsoft Power Platform	Призначення та основна функціональність модуля
1	Power Apps	Модуль для швидкого створення мобільних додатків і порталів професійного рівня. Додатки створюється у візуальному редакторі (конструкторі), який містить великий набір готових шаблонів. Готові шаблони можна швидко адаптувати під конкретні задачі або створити свій шаблон.
2	Power Automate	Модуль автоматизації бізнес-процесів і рутинних завдань, що повторюються. Причому автоматизуватися можуть як локальні завдання, так і великомасштабні ланцюжки процесів, інтегровані зі сторонніми системами через сотні готових з'єднувачів.
3	Power business intelligence (Power BI)	Одна з найпопулярніших і масштабних систем бізнес-аналітики. Power BI дає можливість отримувати аналітику в режимі реального часу й оперативно реагувати на зміни. Система дає змогу завантажувати дані з різних джерел, здійснювати їх моделювання та візуалізацію (модуль Power BI Desktop), публікувати візуальні звіти в Інтернеті (модуль Power BI Service) та на мобільних пристроях (модуль Power BI Mobile).
4	Power Virtual Agent	З цим продуктом створення професійних чат-ботів стає простим завданням, оскільки весь процес відбувається в графічному інтерфейсі. Для більш якісної і глибокої роботи з клієнтами в чат-боти вбудована можливість використання штучного інтелекту (ШІ). При цьому ШІ не вимагає програмування, після створення його можна відразу включати в роботу.

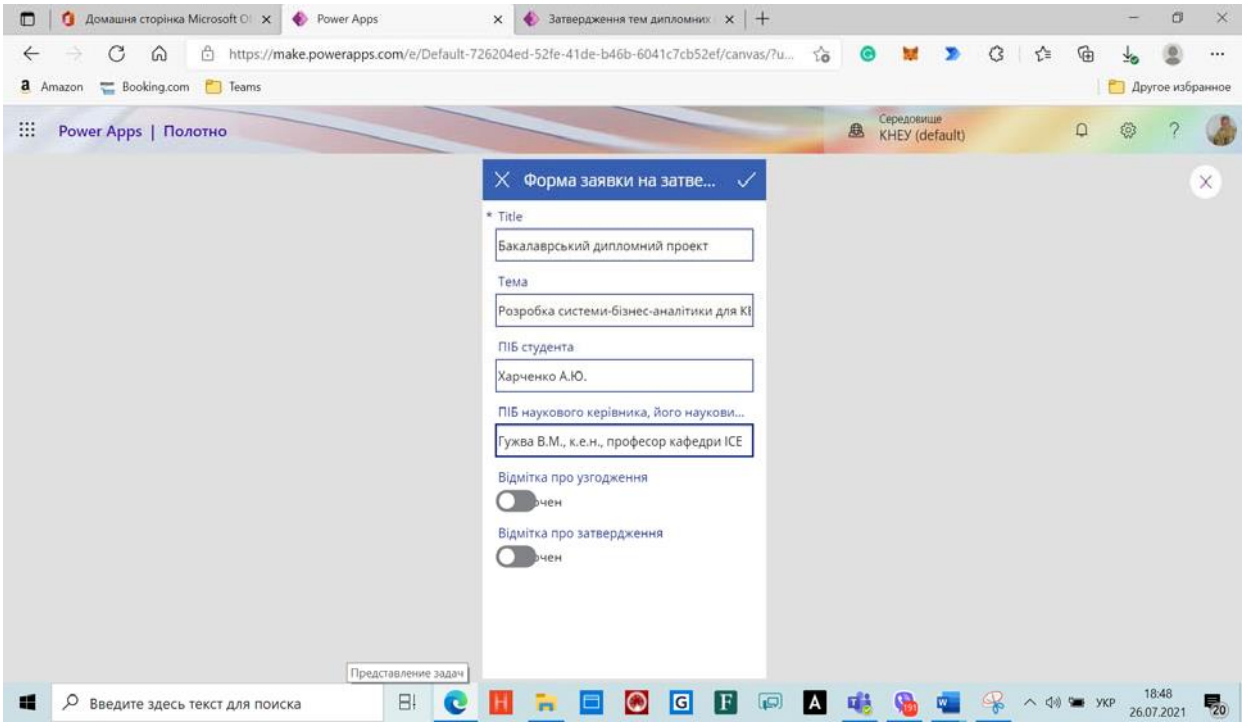


Рисунок 1 – Розроблена за допомогою Power Apps форма для вибору теми і наукового керівника

Хмарні технології у діловодстві

Сучасні хмарні сервіси можуть функціонувати у вигляді чотирьох моделей розгортання (deployment models):

приватна хмара (private cloud),

хмара співтовариства або загальна хмара (community cloud),

публічна хмара (public cloud) і

гібридна хмара (hybrid cloud).

Приватна хмара — це інфраструктура, яка забезпечує обслуговування лише однієї організації. Вона може управлятися самою організацією або іншою стороною й існувати як на стороні споживача (on premise), так і в зовнішнього провайдера (off premise).

Хмара співтовариства використовується спільно кількома організаціями із спорідненими обчислювальними (інформаційними) ресурсами і завданнями. При цьому, завдяки взаємній довірі, забезпечується вищий рівень конфіденційності і захисту інформації, ніж в публічній хмарі. Така хмарна інфраструктура може управлятися самими організаціями або третьою стороною й існувати як на стороні споживача, так і в зовнішнього провайдера.

Публічна хмара відноситься до моделі хмарних технологій, в якій провайдер надає відповідні ІКТ-ресурси для широкої аудиторії Інтернету. Сервіси публічної хмари, як правило, пропонуються на уже використовуваній моделі.

Гібридна хмара є композицією (поєднанням) двох і більше хмар попередніх типів (приватних, співтовариства або публічних). При цьому хмари, що входять до її складу, залишаються унікальними сутностями й об’єднуються відповідними технологіями для забезпечення належного рівня обміну даних між ними. Гібридні моделі хмар дозволять організаціям зберігати конфіденційність своєї інформації в межах локальних центрів обробки даних, передаючи менш конфіденційні дані в хмару для економії витрат і ширшого доступу.

Технологічний рівень сучасних апаратних і програмних комплексів повністю дозволяє перенести СЕД у хмару. При цьому під перенесенням документообігу в хмару розуміється виконання частини (або всіх) його функцій системою сторонньої компанії, яка взаємодіє з ІТ-інфраструктурою замовника через Інтернет.

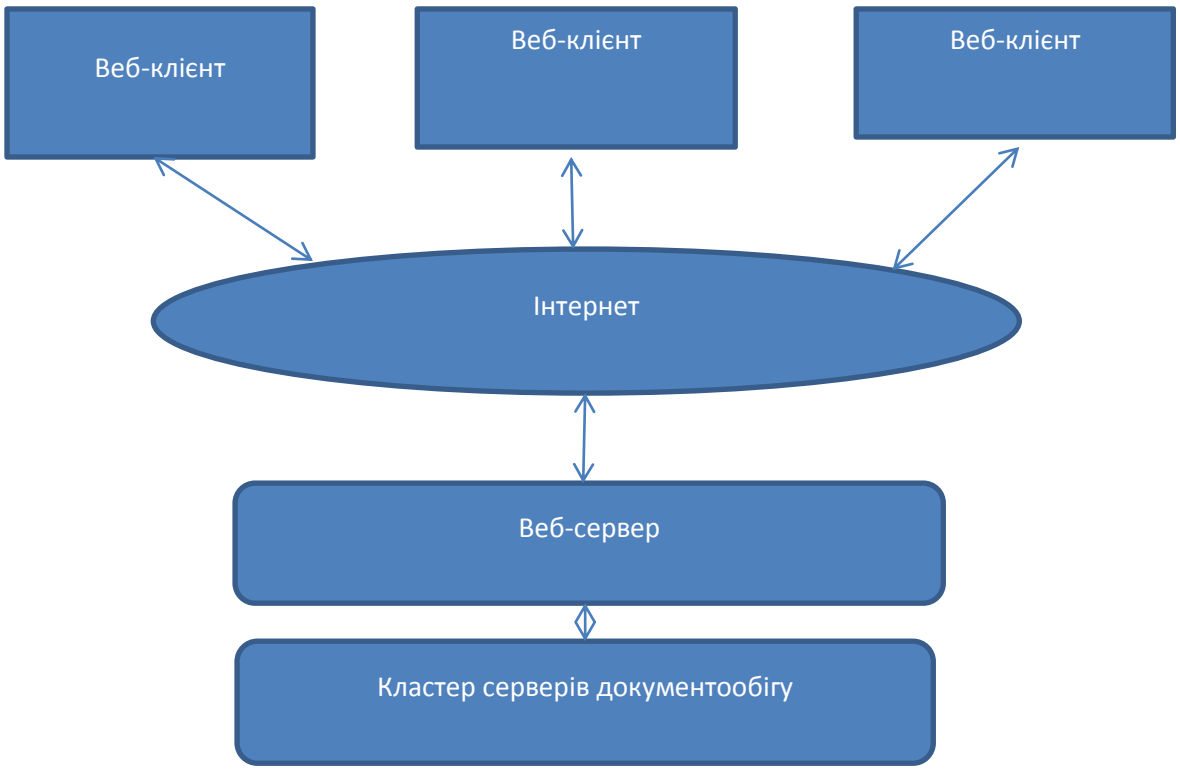


Рис. 1. Схема організації електронного документообігу в хмарі

Хмарні системи дозволяють організовувати повний життєвий цикл документа, починаючи зі сканування і перетворення паперового документа в електронну форму і закінчуючи архівним збереженням. До того ж сканування може здійснюватися як з розпізнаванням і подальшим створенням картки документа з відповідними атрибутами, так і без нього.

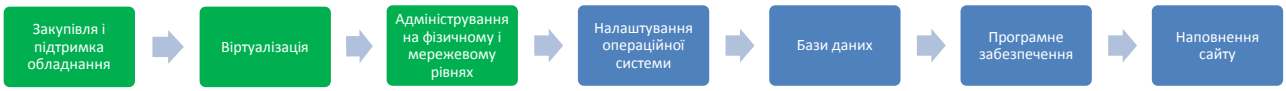
СЕД у хмарі дозволяє зберігати і систематизувати будь-які документи за видами або напрямками, організовувати контекстний пошук, спільну роботу над документами з різними пріоритетами доступу, здійснювати перегляд документів з мобільних пристроїв (планшетів, смартфонів) тощо. Керівник може швидко і легко налаштувати параметри доступу, визначивши, які співробітники можуть переглядати чи редагувати певні документи. Для спрощення роботи з типовими документами передбачено створення шаблонів. Хмарна технологія дозволяє працювати з документом не тільки з офісного комп'ютера, а й, наприклад, з ноутбука, підключеного до бездротової мережі в Internet-кафе.

Використання хмарних технологій підвищує вимоги до безпеки контенту, до технологій обмеження доступу, шифрування даних і застосування електронного підпису.

Хмарні технології з’явились порівняно нещодавно, однак уже мають різні види моделей розгортання (IaaS, SaaS, PaaS), кожна з яких може бути окремо окреслена з точки зору технічного та нормативно-правового застосування.

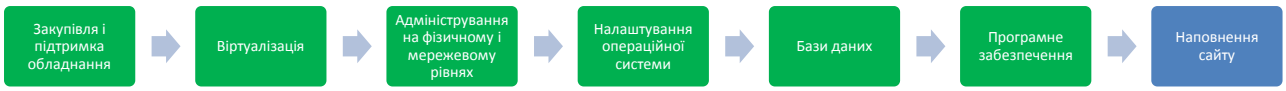
- **IaaS** — Infrastructure as a Service — *інфраструктура як послуга*, наприклад, віртуальні сервери і віртуальна мережа; клієнт може установлювати будь-яке програмне забезпечення.

  Споживачеві надаються засоби обробки даних, зберігання, мереж і інших базових обчислювальних ресурсів, на яких можна розгортати і виконувати довільне програмне забезпечення, включаючи операційні системи і прикладання. Споживач не управляє і не контролює саму хмарну інфраструктуру, але може контролювати операційні системи, засоби зберігання, розроблені прикладання та володіти обмеженим контролем над вибраними мережевими компонентними (наприклад, мережевий екран хоста, керованого споживачем). Вочевидь, модель IaaS має найбільший рівень безпеки за рахунок можливості контролю над ресурсами, але потребує більших затрат на реалізацію.

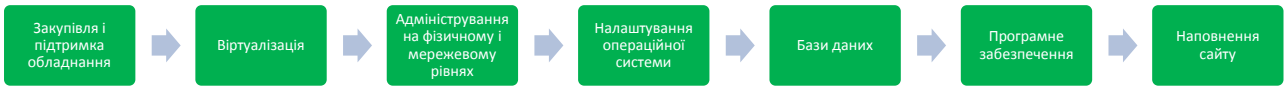


- **PaaS** — Platform as a Service — *платформа як послуга*, наприклад, веб-сервер чи база даних; клієнт здійснює управління додатками, операційною системою керує провайдер.

Споживачеві надаються засоби для розгортання (deploy) на хмарній інфраструктурі створюваних споживачем прикладань, що розробляються з використанням підтримуваних провайдером інструментів і мов програмування. Модель PaaS передбачає більший контроль за процесом обробки даних з боку користувача, але перекладає на нього частину відповідальності та потребує додаткових затрат на розробку прикладань.



- **SaaS** — Software as a Service — *програмне забезпечення як послуга*, наприклад, електронна пошта; клієнт користується додатками, базовими налаштуваннями додатків керує провайдер.



Споживачу надаються програмні засоби – додатки провайдера, що виконуються на хмарній інфраструктурі. Додатки доступні з різних клієнтських пристроїв через інтерфейс «тонкого» клієнта, такий як браузер (наприклад, електронна пошта з web-інтерфейсом). Перевагою такого виду хмарного сервісу є можливість роботи з додатками, що виконуються на хмарній інфраструктурі, не лише із застосуванням «тонких» клієнтів, але і спеціальних клієнтських застосувань, що завантажуються за потреби. Основний недолік – споживач не має можливостей контролювати саму хмарну структуру, на якої виконується прикладання. Але у ряді випадків він може отримати доступ к деяким настройкам конфігурації.

Загалом, архітектура клієнт-сервер є одним із архітектурних шаблонів програмного забезпечення та є домінуючою концепцією у створенні розподілених мережних застосунків і передбачає взаємодію та обмін даними між ними. Вона передбачає такі основні компоненти:

набір серверів, які надають інформацію або інші послуги програмам, які звертаються до них;

набір клієнтів, які використовують сервіси, що надаються серверами;

мережа, яка забезпечує взаємодію між клієнтами та серверами.

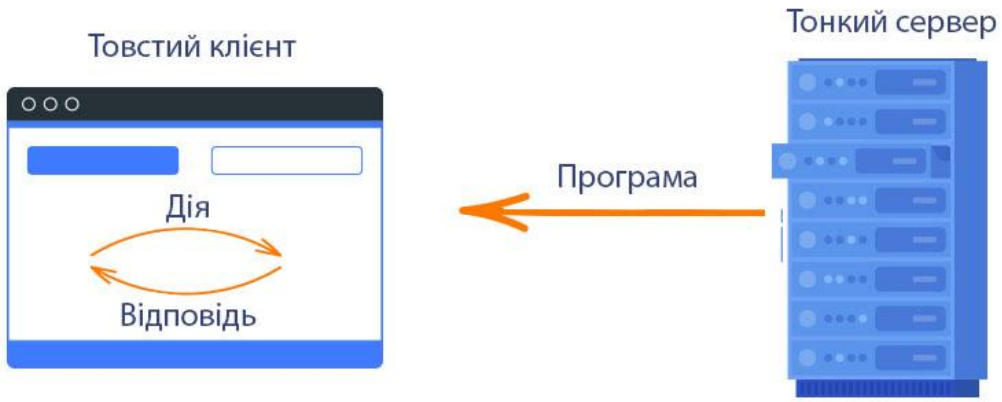
У комп'ютерних технологіях термін «клієнт» має на увазі програмне або апаратне забезпечення, яке взаємодіє з сервером для отримання інформації або виконання певних дій. Клієнт є важливою частиною клієнт-серверної архітектури. Прикладами клієнтів можуть бути web-браузери. Вони виступають web-клієнтами і відправляють запити на web-сервер, отримуючи у відповідь потрібну web-сторінку.

Клієнтів у моделі клієнт-серверної архітектури можна розділити на 2 типи: тонкі та товсті. Також існують архітектури, які об'єднують можливості і тонких, і товстих клієнтів – гібридні клієнти.

Що таке товстий клієнт

Товстий клієнт – клієнт, який проводить запитувані користувачем операції незалежно від центрального сервера. Центральний сервер в такому варіанті архітектури може використовуватися як сховище даних, обробка та надання яких переноситься на робочу машину клієнта.

Товстим клієнтом є робоча станція або ПК, які працюють під управлінням власної операційної системи і мають повний необхідний набір програмного забезпечення для реалізації завдань користувача.



Плюси товстих клієнтів:

- широка функціональність;
- режим з багатьма користувачами;
- робота в режимі оффлайн;
- висока швидкодія;
- мінімізація залежності від дорогих та складних серверів.

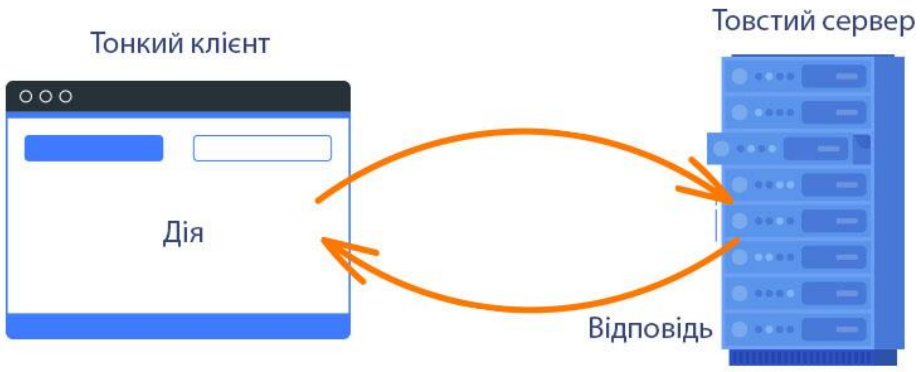
Недоліки:

- кожна робоча машина потребує постійного обслуговування;
- індивідуальне оновлення апаратного забезпечення кожного клієнта до рівня додатків, які будуть використовуватися;
- можливість виникнення проблем з віддаленим доступом до даних;

- великі розміри дистрибутивів;
- залежність від платформи, для якої клієнт був розроблений.

Що таке тонкий клієнт

Тонкий клієнт – тип клієнта, який переносить завдання з обробки даних на сервер, не використовуючи свої обчислювальні можливості для їх реалізації. Обчислювальні ресурси такого клієнта дуже обмежені, їх повинно бути досить лише для запуску необхідного мережевого додатку, використовуючи, наприклад, web-інтерфейс.



Одним із прикладів використання тонкого клієнта є ПК із встановленим web-браузером, який використовується для роботи з web-додатками. Особливість тонких клієнтів – використання термінального режиму. В такому випадку, термінальний сервер використовується для відправки і отримання даних користувача, що і є головною відмінністю від незалежної обробки даних в товстих клієнтах.

Крім використання програмного варіанту тонкого клієнта, існують також апаратні рішення тонких клієнтів. Це пристрої, які можуть не мати свого жорсткого диска, використовують спеціальну локальну операційну систему, основне завдання якої – встановити зв'язок з сервером, за допомогою якого будуть вирішуватися завдання користувача.

Плюси тонких клієнтів:

- менше обслуговування апаратного обладнання та програмного забезпечення користувачів;

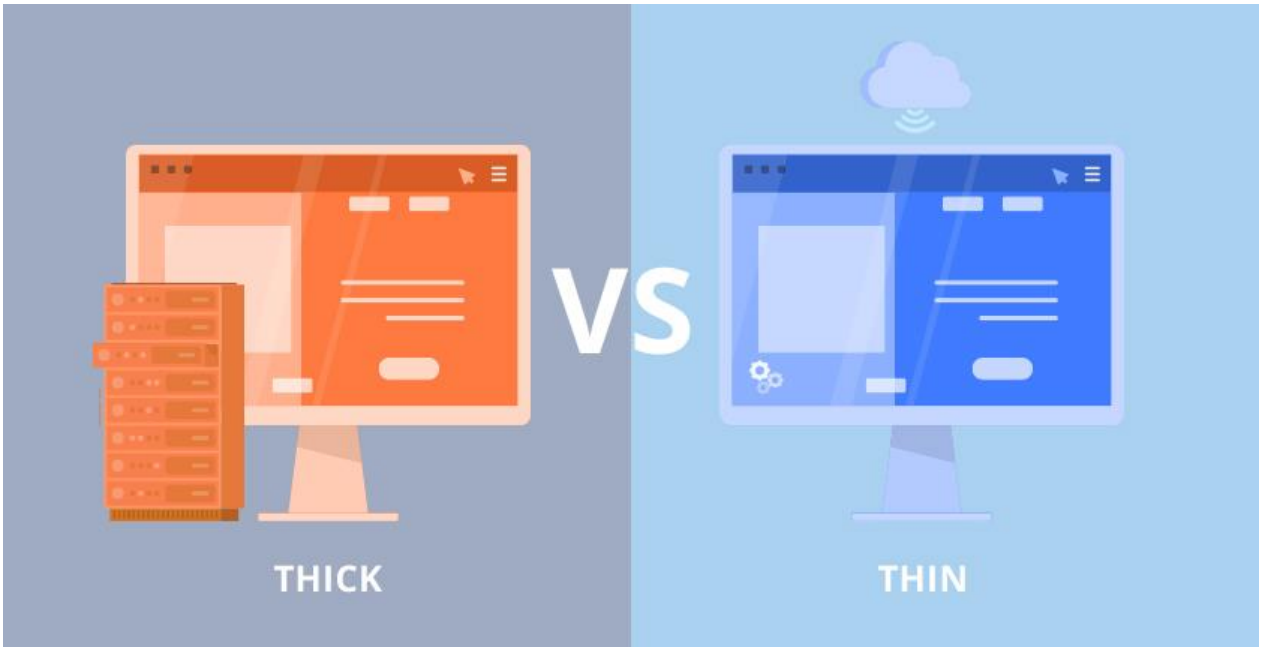
- зниження ризику несправностей, оскільки файли і додатки зберігаються на центральному сервері;
- менше вимог до апаратного обладнання у порівнянні з товстими клієнтами.

Недоліки:

- загальна точка відмови: у разі збою на сервері будуть охоплені всі користувачі;
- неможливість працювати без підключення до мережі;
- при великому обсязі роботи з відео і аудіо даними (особливо створення та редагування) централізація тонких клієнтів може сильно знизити продуктивність центрального сервера.

Різниця між тонкими і товстими клієнтами

Основною відмінністю між тонкими і товстими клієнтами є спосіб обробки інформації. Товстий клієнт здійснює роботу з даними використовуючи свої апаратні і програмні можливості, а з сервером пов'язується тільки для отримання даних з бази. Тонкі клієнти, в свою чергу, використовують забезпечення центрального сервера для обробки інформації, надаючи лише необхідний інтерфейс для роботи користувача. Тому в якості тонких клієнтів можуть бути використані вже застарілі ПК.



Порівняння товстих і тонких клієнтів за основними характеристиками

Незалежність – товсті клієнти працюють незалежно від центрального сервера і використовують свої ресурси. Тонкі клієнти майже повністю залежні від центрального сервера та доступних на ньому ресурсів.

Ресурсовитратність – товсті клієнти використовують більше локальних ресурсів, оскільки самі виконують всі функції. Локальні ресурси тонких клієнтів призначені лише для створення сесії зв'язку з сервером.

Збереження даних – дані користувачів товстих клієнтів зберігаються локально на робочій машині.

Підключення до мережі – товсті клієнти можуть працювати в оффлайн режимі, тонкі клієнти ж потребують постійного доступу до інтернету.

Розгортання – товсті клієнти потребують великих затрат, тому що потрібно індивідуально оновлювати кожну робочу машину під конкретні завдання користувача. Тонкі клієнти не такі витратні, тому можуть використовуватися дуже прості по апаратному забезпеченню ПК. Головною витратою при роботі з тонкими клієнтами буде високопродуктивний сервер та його налаштування.

Безпека – підвищені проблеми з безпекою можуть виникнути при роботі з товстими клієнтами через великий ризик втрати даних на індивідуальних робочих машинах. Тонкі клієнти вважаються більш безпечними, тому що дані зберігаються на сервері.

Незважаючи на очевидні переваги, основним стримуючим фактором використання хмарних сервісів є проблема забезпечення безпеки та низький рівень довіри до постачальників хмарних послуг. Не менш вагомою проблемою є поточне законодавство України, яке виключає розміщення та обробку важливих даних за її межами. Деякі провайдери (наприклад, Google, Symantec) декларують послуги розміщення своїх ресурсів в відповідній країні, але такі гарантії є скоріше виключенням для постачальників послуг, та й перевірити це практично неможливо. Крім того, стримуючим фактором є питання прозорості діяльності сервіс-провайдерів, труднощі з оцінкою фінансової ефективності використання хмарних сервісів та інтеграції різних хмарних сервісів між собою

та з нехмарними сервісами, неготовність керівних органів до використання хмарних сервісів, труднощі міграції на хмарні технології та від одного хмарного провайдера до іншого і ін.

Ефективним шляхом вирішення проблеми безпеки зберігання інформації є шифрування даних. Провайдер, що надає доступ до даних, повинен шифрувати інформацію клієнта, а також у випадку відсутності необхідності подальшого зберігання, оперативно її видаляти. Зашифровані дані при передачі повинні бути доступні тільки після аутентифікації. Для забезпечення її більш високої надійності використовуються токени та сертифікати. Тоді дані будуть захищені навіть у випадку доступу через ненадійні вузли.

Віртуальні мережі повинні бути розгорнуті із застосуванням надійних технологій (наприклад, VPN, VLAN і VPLS). Часто провайдери ізолюють дані користувачів один від одного за рахунок зміни даних коду в єдиній програмному середовищі. Але даний підхід має певні ризики, пов'язані з небезпекою знайти дірку в нестандартному коді, що дозволяє отримати доступ до даних. У випадку можливої помилки в коді користувач може отримати дані іншого.

Технології електронного підпису

Для електронного документа здійснюється обчислення електронного підпису за криптографічними алгоритмами та розрахунок геш-функції.

Колектив авторів Дронюк І., Шкодин А., Барабаш І., Закала М. стверджують, що «алгоритми шифрування поділяють на симетричні, асиметричні та геш-функції». Симетричні криптосистеми використовують спосіб шифрування, в якому для шифрування і дешифрування застосовується однаковий криптографічний ключ. До асиметричних криптосистем належить криптографічна система з відкритим ключем. Перевага асиметричних шифрів над симетричними полягає у тому, що не треба передавати секретний ключ. Крім симетричних та асиметричних криптосистем, широко застосовують геш-функції. Геш-функції використовують для перевірки цілісності документа у

випадку його підписання не автором чи створювачем, однак розраховані за геш-функціями геш-значення не доводять авторство документа.

Існує значна кількість алгоритмів гешування з різними характеристиками (обчислювальна складність, криптостійкість тощо). Серед основних алгоритмів гешування найпоширенішими є CRC, SHA-1, SHA-2 (SHA-224, SHA-256, SHA-384, SHA-512), RIPEMD-128, RIPEMD-160, RIPEMD-256, RIPEMD-320, MD2, MD4, MD5, Tiger, Whirlpool [13]. Гешування застосовується для порівняння даних. Якщо у двох масивах геш-значення різні, масиви гарантовано розрізняються; якщо однакові - масиви, швидше за все, однакові. У загальному випадку, однозначної відповідності між вихідними даними і геш-значенням немає в силу того, що кількість значень геш-функцій менша, ніж варіантів вхідного масиву, існує безліч масивів, які дають однакові геш-значення - так звані колізії.

Як бачимо, гешування може використовуватися для перевірки електронного документа, однак не доводить авторство документа та не гарантує однозначної відповідності навіть у випадку співпадання геш-значень файлів електронного документа. Саме з цієї причини з метою надійного захисту інформації може використовуватися гешування в поєднанні з асиметричними алгоритмами кодування даних.

2020 року запроваджено новий Національний стандарт ДСТУ 9041:2020. Його повна назва: ДСТУ 9041:2020. Інформаційні технології. КРИПТОГРАФІЧНИЙ ЗАХИСТ ІНФОРМАЦІЇ. Алгоритм шифрування коротких повідомлень, що ґрунтується на скручених еліптичних кривих Едвардса.

Цей алгоритм призначений для шифрування коротких (до 616 біт) повідомлень для будь-яких алгоритмів шифрування, в тому числі визначених національними стандартами України.

Як і стандарт цифрового підпису ДСТУ 4145:2002, новий алгоритм використовує криптографічні перетворення у групі точок еліптичних кривих, використовуючи замість кривих у формі Вейєрштрасса найновітніші розробки у галузі еліптичної криптографії – криві у формі Едвардса. Це дає

суттєві переваги у швидкодії більш ніж у 3 рази. Новий стандарт розроблений з урахуванням усіх найсучасніших вимог до стійкості криптографічних алгоритмів.

Стандарт ДСТУ-9041 узгоджений з усіма діючими в Україні національними стандартами. Новиною стандарту є його сфера застосування – інкапсуляція ключів, найсучасніший математичний апарат, а також новий алгоритм генерації псевдовипадкових послідовностей, який, на відміну від аналогічного алгоритму генерації з ДСТУ 4145, використовує виключно національні криптографічні алгоритми національних стандартів та не містить посилань на відповідні пост-радянські стандарти, термін дії яких вже практично вичерпався.

Контейнери ЕД

Контейнер визначено як “файл, який відповідає специфікації формату ZIP та вимогам ISO/IEC 21320- 1:2015”. Крім того, наголошено, що файли, електронні цифрові підписи та/або печатки, накладені на відповідні “файли, папки, метадані, що містяться в контейнері, структуруються у відповідній ієрархії, яка визначається форматом даних контейнера”. У свою чергу формат даних такого контейнера має відповідати вимогам національного стандарту ДСТУ ETSI EN 319 162-1:2016 (ETSI EN 319 162-1:2016, IDT). Отже агентство з питань електронного урядування України вважає за доцільне застосовувати європейський стандарт контейнерів асоційованих підписів (Associated Signature Containers, ASiC), в якому визначено стандартизоване використання типів контейнерів для встановлення спільного способу асоціювання файлів, що містять об'єкти даних, з файлами, що містять цифрові підписи та/або підтвердження часу.

ASiC — це контейнер даних, що містить набір об'єктів-файлів та пов'язаних цифрових підписів та/або підтверджень часу з використанням формату ZIP. Будь-який контейнер ASiC має внутрішню структуру, що включає:

• кореневу папку для всього вмісту контейнера, можливо, включаючи папки, що відображають структуру вмісту;

• папку «META-INF» в кореневій папці для файлів, що містять метадані щодо вмісту, включаючи пов'язані з ним файли підписів та підтверджень часу.

Відокремлені підписи або підтвердження часу застосовуються таким чином, щоб не порушувалася цілісність даних, якщо файли вилучаються з контейнеру ZIP. Отже підписи та підтвердження часу, що використовуються в ASiC, можна верифікувати щодо об'єктів-файлів, до яких вони застосовуються, коли вони знаходяться поза структурою контейнера (наприклад, якщо вони розміщені в локальному сховищі).

ІТ в архівній справі

Інформаційні технології значно полегшують роботу архівістів. Так, з допомогою інформаційних технологій легше стало знайти потрібні документи, зареєструвати при їх прийнятті тощо. Відповідно й зростає якість наданих послуг. Впровадження інформаційних технологій виступає сьогодні як одна з необхідних умов удосконалення функціонування архівних установ. Її впровадження надає можливість державним архівним установам аналізувати зміни, що відбуваються у такій складній системі, як суспільство, поліпшувати планування, облік і контроль в її діяльності.

Інформатизація архівної справи базується за наступними напрямами:

- впровадження системи електронного документообігу;

- створення захищеної архівної інформаційної автоматизованої системи та централізованого електронного довідкового апарату;

- запровадження архівних електронних послуг;

- оцифрування документів Національного архівного фонду України.

Загалом впровадження інформаційних технологій в діяльність архівних установ, його результативність залежить від багатьох чинників. Основні проблеми, з якими стикаються сучасні державні архівні установи при впровадженні інформаційних технологій це:

нестача фінансових ресурсів;

відсутність належного комп’ютерного та програмного забезпечення;

відсутність у працівників навичок та вмінь роботи із сучасними інформаційними технологіями;

небажання працівників вчитись новому.