УДК 004.072 DOI: 10.31673/2518-7678.2023.020606

Вишнівський В.В., д.т.н., **Іщеряков С.М.,** к.т.н.,
Аверічев І.М., к.е.н., **Каргаполов Ю.В.**

НОВИЙ ПІДХІД ДО АРХІТЕКТУРИ СИСТЕМ УПРАВЛІННЯ СЕРВІСАМИ В ІНФОРМАЦІЙНИХ СИСТЕМАХ

Vyshnivskyi V.V., Ishcheryakov S. M., Averichev I. M., Kargapolov Y.V. A new approach to the architecture of service management systems in information systems. The article solves the scientific task of researching new principles of service management architecture based on the distribution of subject and identification data flows using the tool of the digital object identification register and the service register. The main task of the integration of information systems related to different areas is the management of the processes of providing services according to customer requests. The risks that have arisen are related to the large amount of data that the integration bus (platform) must pass through itself, including security risks and increased transaction processing costs. An additional factor restraining development is the poorly resolved task of integrating several information systems that use different ecosystems of services with different telecommunication and subject protocols, data request and processing methods, data structures and formats.

Keywords: architecture, computer system, identification, service, service, digital object, information technology

Вишнівський В.В., Іщеряков С.М., Аверічев І.М., Каргаполов Ю.В. Новий підхід до архітектури систем управління сервісами в інформаційних системах. У статті вирішується наукове завдання дослідження нових принципів архітектури управління сервісами на основі розподілу потоків предметних та ідентифікаційних даних із використанням інструменту реєстру ідентифікації цифрових об'єктів та реєстру сервісів. Основним завданням інтеграції інформаційних систем, що стосуються різних областей є управління процесами надання послуг за запитами клієнтів. Виниклі ризики пов'язані з великим обсягом даних, які інтеграційна шина (платформа) повинна пропускати через себе, у тому числі ризики безпеки та збільшення витрат на обробку транзакцій. Додатковим фактором, що стримує розвиток, є завдання, що погано вирішується щодо інтеграції декількох інформаційних систем, що використовують різні екосистеми сервісів з різними телекомунікаційними та предметними протоколами, методами запиту і обробки даних, структурами і форматами даних.

Ключові слова: архітектура, комп’ютерна система, ідентифікація, сервіс, послуга, цифровий об’єкт, інформаційна технологія

Вступ.

Мета ефективного функціонування цифрових об'єктів в інформаційній системі полягає у створенні можливостей для їхньої взаємодії з екосистемою різноманітних сервісів. Це включає як внутрішні сервіси, які оптимізують внутрішні процеси і функції системи, так і зовнішні, які надають або отримують послуги з іншими системами. Сутність цього процесу полягає в створенні мережі взаємозв'язків, де цифрові об'єкти взаємодіють один з одним та з різними сервісами для досягнення конкретних цілей. Отримання та надання послуг стає ключовим аспектом цього складного процесу. Управління взаємозв'язками цифрових об'єктів та сервісів здійснюється не лише через функціональні можливості інформаційної системи. Однак, важливим елементом є також управління ідентифікацією цифрових об'єктів і сервісів. Це включає в себе процеси визначення, перевірки та контролю за ідентичністю цифрових суб'єктів, що забезпечує безпеку та вірогідність взаємодії.

Таким чином, управління взаємозв'язками і ідентифікацією відіграють важливу роль у створенні динамічної та довіренної екосистеми цифрових об'єктів і сервісів в межах інформаційної системи.

Аналіз останніх досліджень і публікацій.

Системи управління сервісами в інформаційних системах є ключовим елементом для ефективної організації та взаємодії функціональних частин системи.

Вона включає такі аспекти: [1-9] :

- архітектура мікросервісів: Замість монолітної структури, інформаційні системи використовують архітектуру мікросервісів, де окремі функції реалізуються невеликими та автономними сервісами;
- реєстрація та виявлення сервісів: Система управління дозволяє реєструвати, виявляти та взаємодіяти із сервісами в автоматизований спосіб, забезпечуючи їх доступність та стабільність;
- моніторинг та керування відмовами: Забезпечення стійкості системи шляхом моніторингу стану сервісів, автоматичного виявлення відмов та перехоплення їх впливу;
- оркестрація та управління ресурсами: Керування життєвим циклом сервісів, включаючи їхню розгортку, масштабування та відключення, для оптимального використання ресурсів;
- безпека та ідентифікація: Захист інформації та забезпечення автентифікації сервісів та користувачів для забезпечення безпеки взаємодії;
- міжсервісна комунікація: Забезпечення ефективної комунікації між різними сервісами, часто використовуючи стандартні протоколи та інтерфейси;
- автоматизоване ведення журналів та аналіз даних: Збір та аналіз логів для моніторингу, виявлення помилок та оптимізації функціонування системи;
- загальна мета систем управління сервісами полягає в створенні гнучких, масштабованих та надійних інформаційних систем, які забезпечують високу продуктивність та зручність управління.

Постановка наукового завдання.

Практика розробки інформаційних систем, що стосуються різних областей – інтелектуальних транспортних систем, переносимості абонентських номерів, адміністрування кодів IMEI, систем індустріального IoT, систем IoT для житлового та сільського господарства тощо. Основним завданням інтеграції є управління процесами надання послуг за запитами клієнтів. Виниклі ризики пов'язані з великим обсягом даних, який інтеграційна шина (платформа) повинна пропускати через себе, у тому числі ризики безпеки та збільшення витрат на обробку транзакцій. Додатковим фактором, що стримує розвиток, є завдання, що погано вирішується щодо інтеграції декількох інформаційних систем, що використовують різні екосистеми сервісів з різними телекомунікаційними та предметними протоколами, методами запиту і обробки даних, структурами і форматами даних.

Як показує практика, прийнята архітектура управління сервісами не дає рішення.

У статті вирішується наукове завдання дослідження нових принципів архітектури управління сервісами на основі розподілу потоків предметних та ідентифікаційних даних із використанням інструменту реєстру ідентифікації цифрових об'єктів та реєстру сервісів.

Метою роботи є підвищення ефективності управління ідентифікацією цифрових об'єктів на основі нових принципів побудови архітектури мультисервісних систем.

Виклад основного матеріалу дослідження

Управління сервісами є функцією, спрямованою на досягнення цілей технологічних (основних та допоміжних) процесів системи. Ця функція може бути складною чи простою залежно від ступеня складності процесу. Але вона завжди має на увазі, що для її реалізації необхідно мати сам сервіс або послуги і клієнта чи клієнтів.

Складність процесу управління починає зростати у разі, якщо клієнт вступає за допомогою системи (її інструментальних засобів) у відносини з багатьма сервісами, а той самий сервіс стає необхідним для багатьох клієнтів.

Можна розглядати такі випадки взаємодії між клієнтами та сервісами:

1) **1 клієнт : 1 сервіс**, як представлено на схемі нижче (рисунок 1)

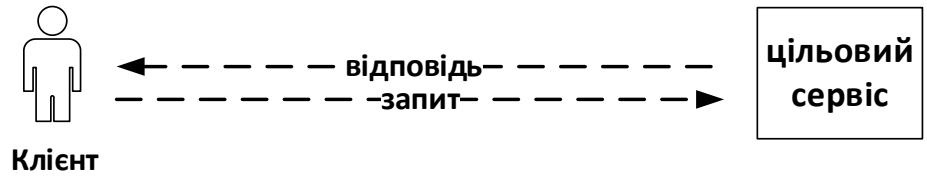

Рисунок 1. Спрощене уявлення про обробку запиту та відповіді між клієнтом та сервісом для випадку 1 : 1

2) **M клієнтів : 1 сервіс**, як представлено на схемі нижче (рисунок 2)

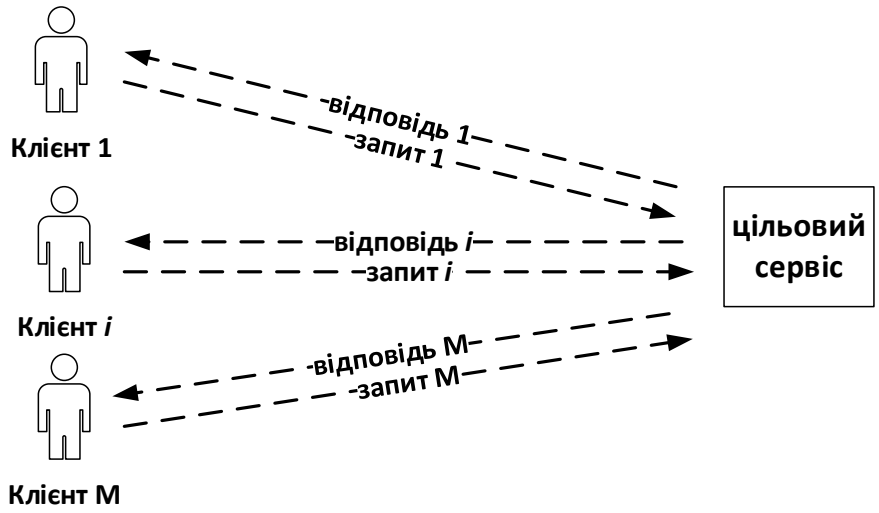

Рисунок 2. Спрощене уявлення про обробку запиту та відповіді між клієнтом та сервісом для випадку M : 1

3) **1 клієнт : N сервісів**, як представлено на схемі нижче (рисунок 3)

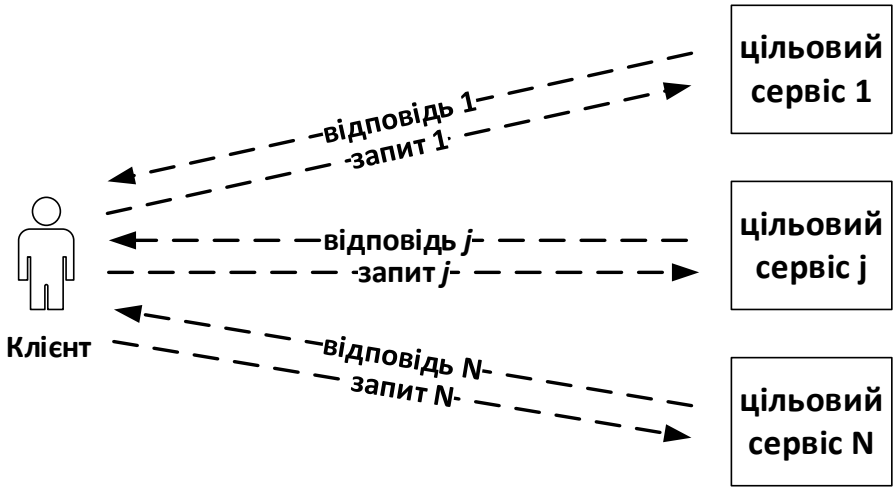

Рисунок 3. Спрощене уявлення про обробку запиту та відповіді між клієнтом та сервісом для випадку 1 : N

4) **M клієнтів : N сервісів**, як представлено на схемі нижче (рисунок 4)

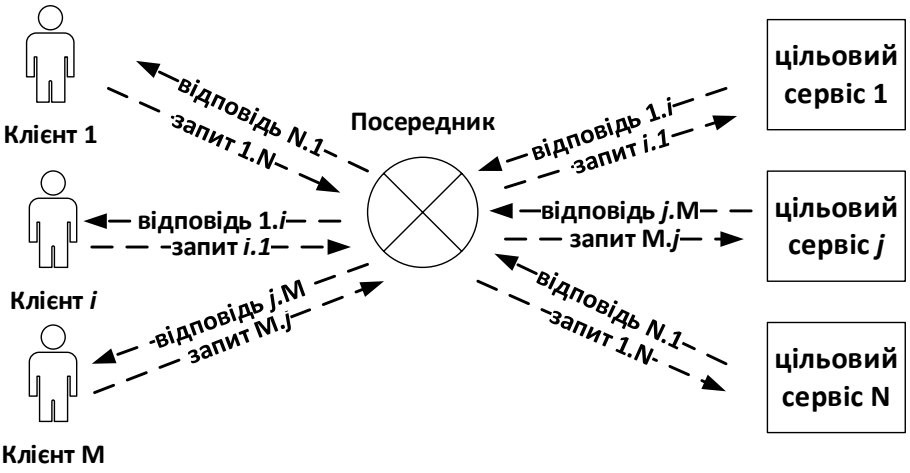

Рисунок 4. Спрощене уявлення про обробку запиту та відповіді між клієнтом та сервісом для випадку M : N

Для формування загальної моделі взаємодії між клієнтом та сервісами розглядатимемо загальний випадок M клієнтів: N сервісів. У цьому варіанті взаємодія між клієнтами та сервісами потребує позиції посередника, який визначає та керує чергою звернень клієнтів до сервісів та відповідей, що надходять від сервісів до клієнтів.

Проте на рисунках 1 - 4 дано спрощене уявлення про процеси взаємодії клієнтів та сервісів. Необхідно розглядати випадки, коли виконання цільового сервісу потрібує виконання ланцюжка сервісів, передбачених логікою технологічного процесу у системі. Наприклад, щоб клієнт у вигляді громадського транспортного засобу типу автобус при зверненні до сервісу забезпечення пріоритетного проїзду отримав рішення, інтелектуальна транспортна система, згідно з технологічним процесом, має забезпечити послідовне виконання попередньо сервісів, пов'язаних з моніторингом параметрів транспортних потоків та моніторингом руху громадського транспорту.

Таким чином, клієнт може бути в ситуації, коли сценарій системи виконання цільового сервісу вимагає попереднього виконання ланцюжка сервісів, пов'язаних з цільовим сервісом.

При цьому необхідно брати до уваги, що сценарії системи можуть передбачати виконання цільового сервісу залежно від обставин часу, місця та ситуації, в яких знаходиться клієнт. Це означає, що частина пов'язаних сервісів можуть не виконуватися.

Беручи до уваги наявність множинності потоків даних та сценарної варіативності для різних обставин, в яких знаходиться клієнт, визначальну роль для архітектури системи набуває здатність позиції посередника забезпечувати:

1) пряму командна взаємодія «клієнт-сервіс»[1], оскільки це полегшує управління і знімає навантаження з посередника,

2) децентралізацію управління потоками даних, що використовуються в ході технологічного процесу,

3) відмову від необхідності підтримки централізованих баз даних, крім баз даних сервісів, що використовуються власне сервісами,

4) єдиний простір автентифікації та авторизації прав доступу для клієнтів.

Класична загальноприйнята архітектура інформаційної системи, в якій є безліч цифрових об'єктів (клієнтів), які викликають послуги з метою виконання запропонованої функціональності має три основні складові:

1) Цифрові об'єкти, тобто клієнти в нашому сенсі,
2) Сервіси зі своїми базами даних
3) Керуюча платформа, що виконує функції посередника між клієнтами та сервісами та має свою централізовану базу даних, що дозволяє агрегувати дані, що надходять у рамках системи.

Модель цифрового об'єкта можна представити як набір власних атрибутів цифрового об'єкта та набором атрибутів, які визначає користувач (власник), який вводить ці цифрові об'єкти як актора в систему.

До власних атрибутів відносяться:

- ідентифікатор цифрового об'єкту,
- атрибути власника,
- атрибути життєвого циклу.

До атрибутів, що визначаються користувачем (власником), відносяться:

- властивості цифрових об'єктів, що включають ім'я цифрового об'єкта, його мережеві адреси, предметні протоколи, які повинен дотримуватись цифровий об'єкт під час технологічних циклів та загальний опис цифрового об'єкта,
- структури та формати вхідних та вихідних даних, що описують цифровий об'єкт,
- функції та умови роботи цифрового об'єкта, що визначають допустимі значення даних для забезпечення роботи цифрового об'єкта, мінімальний набір даних, необхідних для функціонування цифрового об'єкта та телекомунікаційні протоколи, за допомогою яких може виконувати свої запити,
- вимоги до безпеки, що містять відомості про сертифікати, що використовуються для роботи цифрового об'єкта та центри сертифікації.

Модель сервісу можна визначити як набір атрибутів:

- ідентифікацію сервісу в інформаційній системі,
- відомості про оператора сервісу, який розробив сервіс та займається його експлуатацією та підтримкою,
- назва сервісу в системі,
- протоколи, які використовує сервіс для підтримки надання послуг,
- параметри аутентифікації, що визначають можливість роботи з сервісом та авторизації, що визначають права доступу для роботи з даними сервісу,
- мережеві адреси серверів, на яких розміщено програмне забезпечення сервісу,
- пов'язані сервіси, які беруть участь у сценарних ланцюжках виконання сервісу,
- структури та формати даних, що використовуються сервісом,

а також як набір функціональних операцій у вигляді методів, які необхідно використовувати для виконання сервісу.

Модель платформи, що управляє, можна представити як набір процесів, що визначають:

- порядок та політики виконання запитів цифрових об'єктів до сервісів,
- моніторинг виконання сервісів,
- порядок та політики отримання цифровими об'єктами відповідних відповідей,

- політики управління потоками даних.

Загалом загальноприйнятий підхід до архітектури інформаційних систем можна визначити рисунком 5.

Наявність централізованої бази даних одна із основних умов роботи елемента архітектури «Посередник», оскільки інформація у ній визначає правила адресації до сервісів і структури даних, які використовуються сервісами.

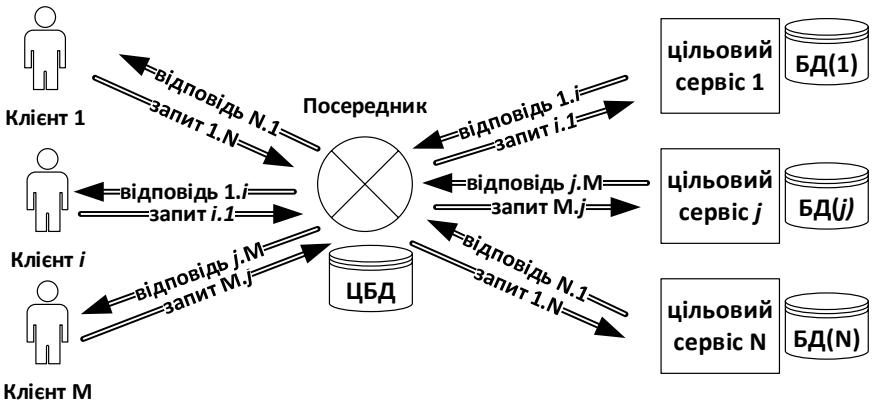

Рисунок 5. Загальноприйнятий підхід до архітектури інформаційних систем

Посередник виконує функцію інтеграції та об'єднує дані з різних джерел для створення їхньої консолідованої версії. Такими джерелами є бази даних сервісів. Централізована база даних робить дані доступними для багатьох цифрових об'єктів без зниження якості даних і може істотно знижувати витрати на маршрутизацію запитів (наприклад, варіанти з dns-cache таблицями).

Наприклад, якщо звернутися до практики створення інформаційних систем управління послугою переносимості абонентських номерів, то згідно з RFC 3482 розглядається 4 схеми організації роботи:

1) All Call Query (ACQ) з централізованою БД,
2) Query on Release (QoR) з централізованою БД,
3) Call Dropback (Return to Pivot (RTP)) з можливістю використання централізованої БД,
4) Onward Routing (OR) з можливістю використання централізованої БД.

Розглядаючи переносимість абонентських номерів як випадок надання послуг клієнтам ($M >> 10^3$) з можливістю залучення N сервісів, в умовах мереж наступного покоління та застосування технологій блокчейн слід зазначити, що згідно з Рекомендацією ITU-T Y.2342 «*… можливості прикладного рівня NGNe-BC надають користувачам певні послуги, що працюють через NGNe-BC, такі як переносимість мобільних номерів (MNP), розрахунок роумінгу тощо. Рівень додатків є джерелом для читання/запису даних з NGNe-BC, а також приводом для виконання стимулів, якщо це застосовано. Зазвичай рівень програми реалізує логіку сервісу та завершує обробку даних.*

Потрібно забезпечити можливість визначення формату даних, що зберігаються у NGNe-BC.

Потрібно забезпечити можливість координації обробки даних, що зберігаються в NGNe-BC та іншій централізованій базі даних в одній сервісній архітектурі.

Необхідно забезпечити можливість керування секретним ключем та алгоритмом шифрування для забезпечення конфіденційності даних.

Потрібно надати інтерфейс користувача або гаманець між користувачем та реєстром NGNe-BC, включаючи операції з обліковим записом, автентифікацію, операції з активами та дані особистої конфіденційності, надані, якщо застосовано.»

Тобто архітектура інформаційних систем з використанням елемента «Посередник», що може виражатися у вигляді інтеграційної шини, передбачає використання управління потоками даних через «Посередника», а узагальнений принцип, який використовується в архітектурі інформаційних систем, полягає в тому, що:

1) *i*-тий клієнт надсилає на адресу «Посередника» запит на виконання *j*-того сервісу, який можна позначити як «запит *i.j*»,

2) «Посередник» використовує інформацію централізованої бази даних, визначає за якою адресою знаходиться сервер, на якому обслуговується *j*-тий сервіс і надсилає дані запиту на його адресу,

3) *j*-ий сервіс отримавши від «Посередника» запит *i*-того клієнта виконує запит і відправляє «відповідь *j.i*» на його адресу через «Посередника».

4) «Посередник» отримує відповідь і знаючи адресу *i*-того клієнта надсилає йому результат.

Таким чином, управління виконанням сервісами в інформаційних системах визначається через функціонал «Посередника» з прогоном через нього всіх даних, пов'язаних як із запитом, так і відповіддю.

Подібна схема, незважаючи на загальне застосування, має суттєвий недолік – «Посередник» отримує потенційний доступ до всіх даних клієнта та сервісу, тому що ці дані проходять через нього.

У цьому випадку виникають питання – якого роду дані проходять через «Посередника» і які небезпеки можуть бути пов'язані з цими даними?

У еталонній моделі, наведеній у Рекомендації MCE-T Y.4000, передбачається, що дані є єдиним масивом, а управління ідентичністю визначає ідентифікацію цифрового об'єкта з точки зору його аутентифікації та авторизації прав доступу до ресурсів інформаційної системи. В цьому випадку через «Посередника» проходить весь масив даних про клієнта, включаючи предметні, автентифікаційні та ідентифікаційні, а також весь масив даних про результати роботи сервісу на запит клієнта, включаючи предметні, авторизаційні та ідентифікаційні. Таке становище становить серйозну загрозу як клієнта, так сервісу. Дані про клієнта можуть потрапити в треті руки в повному обсязі, сервіс може бути скомпрометований і можливість визначити сегмент інформаційної системи, де стався витік важко, оскільки це може статися на сервері сервісу, у клієнта, а може статися у «Посередника».

Виходом із ситуації може бути новий підхід до архітектури систем управління сервісами в інформаційних системах, що ґрунтується на принципі відсікання «Посередника» від будь-якого виду даних, які можуть бути використані між клієнтом та сервісом, крім ідентифікаційних. Така архітектура представлена на рисунку 6.

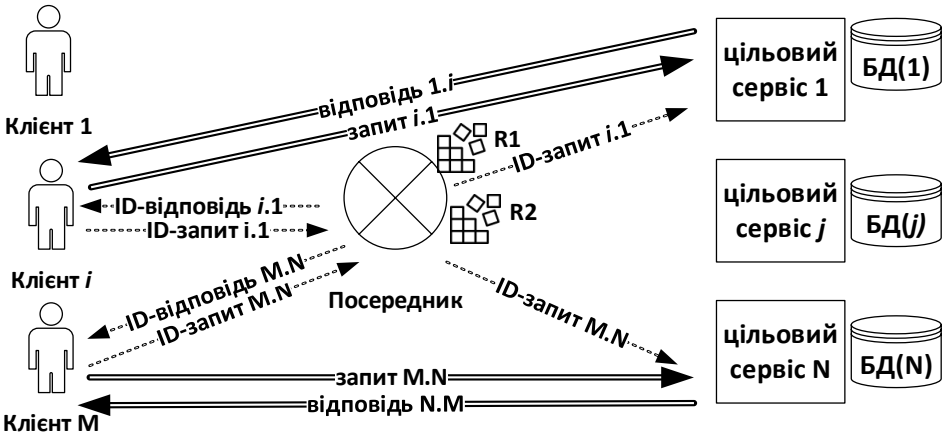

Рисунок 6. Нова архітектура систем управління сервісами в інформаційних системах

Нова архітектура використовує принцип розподілу управління даними, які проходять через «Посередника». Це лише ідентифікаційні дані про цифровий об'єкт (клієнт) та сервіс. Отримання послуг відповідно до сервісу, що викликається клієнтом, відбувається безпосередньо між клієнтом і сервером сервісу, «Посередник» ніякої участі в цьому процесі не бере. Також архітектура не передбачає наявність у «Посередника» централізованої бази даних, у рамках використання якої виникає небезпека формування cache-даних, що проходять через середовище «Посередника». Замість централізованої бази даних «Посередник» має два реєстри – реєстр ідентифікації цифрових об'єктів (R1) та реєстр сервісів (R2).

М-ий клієнт звертається до «Посередника» з ідентифікаційним «ID-запитом M.N» з метою отримати дані про знаходження М-го сервісу. «Посередник» у реєстрі R1 ідентифікує цього клієнта та у реєстрі R2 ідентифікує необхідний клієнту сервіс. Після цього повертає ці дані на адресу клієнта – «ID-відповідь M.N», а також повідомляє N-ий сервіс, про те, що до нього буде звертатися М-ий клієнт. М-ий клієнт безпосередньо робить «запит M.N» і безпосередньо отримує «відповідь N.M». «Посередник» із схеми отримання даних власне сервісу виключено.

Висновки

Матеріал, викладений у статті, дозволяє сформулювати базу створення нових принципів архітектури систем управління сервісами інформаційних систем. Ці принципи ґрунтуються на поділі предметних, аутентифікаційних, авторизаційних та ідентифікаційних даних, які у свою чергу мають бути визначені та розмежовані у моделях цифрових об'єктів та моделі сервісів.

Управління ідентифікаційними даними здійснюється за допомогою двох реєстрів – реєстру ідентифікації цифрових об'єктів та реєстру сервісів, які включені до інформаційної системи. В цьому випадку відпадає необхідність використання централізованих баз даних, що значно знижує ризики порушення інформаційної безпеки та підвищує швидкість та гнучкість отримання клієнтами необхідних сервісів.

Нові принципи архітектури можуть бути використані для проектування інтеграційних шин інформаційних систем.

Список використаної літератури:

1. Recommendation ITU-T Y.4403 (07/2012). Functional requirements and architecture of the next generation network for support of ubiquitous sensor network applications and services.

2. Recommendation ITU-T X.1252 (04/2021). Baseline identity management terms and definitions.

3. Recommendation ITU-T Y.2342 (12/2019). Scenarios and capability requirements of blockchain in next generation network evolution

4. ETSI TS 103 486: "CYBER; Identity Management and Discovery for IoT

5. ETSI TS 132 362 V16.0.0 (2020-08) Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; Telecommunication management; Entry Point (EP) Integration Reference Point (IRP); Information Service (IS) (3GPP TS 32.362 version 16.0.0 Release 16)

6. RFC 3482. Number Portability in the Global Switched Telephone Network (GSTN): An Overview

7. RFC 7642. System for Cross-domain Identity Management: Definitions, Overview, Concepts, and Requirements. September, 2015

8. 3GPP TS 24.382 V13.1.0 (2016-06) Technical Specification. 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Mission Critical Push To Talk (MCPTT) identity management; Protocol specification (Release 13).

9. 3GPP Specification #: 33.924. Identity management and 3GPP security interworking; Identity management and Generic Authentication Architecture (GAA) interworking. Technical Report. (Release 9)

Автори статті

Вишнівський Віктор – доктор технічних наук, професор, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.

Іщеряков Сергій – кандидат технічних наук, доцент, Державний університет інформаційно-комунікаційних технологій, Київ, Україна

Аверічев Ігор – кандидат економічних наук, доцент, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.

Каргаполов Юрій – старший викладач, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.

Authors of the article

Vyshnivskyi Viktor - Doctor of Science (technic), Professor, State University of Information and Communication Technologies, Kyiv, Ukraine.

Ishcheryakov Serhiy – PhD (technic), associate professor, State University of Information and Communication Technologies, Kyiv, Ukraine.

Averichev Ihor - PhD (economics), associate professor, State University of Information and Communication Technologies, Kyiv, Ukraine.

Karhapolov Yuriy - Senior Lecturer, State University of Information and Communication Technologies, Kyiv, Ukraine.