МАГІСТЕРСЬКА РОБОТА

МР. ІПм - 05.00.00.000 ПЗ

Група ІПм-23-1

Бирчак Василь

2024

Івано-Франківський національний технічний університет нафти і газу

Інститут інформаційних технологій

Кафедра інженерії програмного забезпечення

Бирчак Василь Любомирович

(прізвище, ім'я, по батькові)

УДК 004.942

(індекс)

МАГІСТЕРСЬКА РОБОТА

Моделі та методи спільного використання облікових даних на основі специфікації

(назва роботи)

Інженерія програмного забезпечення

(назва освітньої програми)

121 - Інженерія програмного забезпечення

(шифр і назва спеціальності)

Бирчак В.Л.

(підпис, ініціали та прізвище здобувача освітнього ступеня)

Науковий керівник Крихівський Михайло Васильович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Допущено до захисту

Завідувач кафедри

доц. Бандура В.В.

(посада) (підпис) (дата) (ініціали та прізвище)

Нормоконтроль

доц. Вовк Р.Б.

(посада) (підпис) (дата) (ініціали та прізвище)

Робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Івано-Франківськ – 2024

Івано-Франківський національний технічний університет нафти і газу

Інститут інформаційних технологій

Кафедра інженерії програмного забезпечення

Освітній рівень магістр

Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:
Зав. кафедрою **ІПЗ**
доц. В.В. Бандура
“ 04 ” вересня 2024 р.

ЗАВДАННЯ

НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Бирчаку Василю Любомировичу

(прізвище, ім’я, по-батькові)

1. Тема магістерської роботи “ Моделі та методи спільного використання облікових даних на основі специфікації”

керівник проекту (роботи) Крихівський Михайло Васильович, к.т.н., доцент

затверджені наказом закладу вищої освіти від “ 22 ” листопада 2024 р. № 781/7

2. Строк подання студентом проекту (роботи) 15 грудня 2024 р.

3. Вихідні дані до проекту (роботи) Теоретичні концепції та формальні моделі побудови та функціонування інформаційних та програмних технологій певного класу

4. Зміст розрахунково - пояснювальної записки(перелік питань, які потрібно розробити)

1. Аналіз предметної області використання облікових даних
2. Дослідження методів спільного використання облікових даних на основі специфікацій
3. Імплементація моделей та методів для вирішення проблеми використання облікових даних
4. Прикладне застосування пропонованих підходів для системи управління ризиками

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)

1. Загальна архітектура HS2.2 (рис. 1.1)
2. Етапи підключення за допомогою HS 2.2 (рис. 1.2)
3. Загальна блок-схема HS2.2 (рис. 1.3)
4. Структура OMA Device Management Tree (рис. 1.4)
5. Етапи асоціації та автентифікації повторного підключення (рис. 1.5)

6. Консультанти розділів проекту (роботи)

Розділ	Консультант	Підпис, дата
Перевірка на плагіат	доц., к.т.н. Вовк Р.Б.	

7. Дата видачі завдання 04 вересня 2024 р.

Керівник _____
(підпис)

Завдання прийняв до виконання_____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назви етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Підбір і вивчення літератури по темі магістерської роботи	15.09.2024	виконано
2	Аналіз концепцій та алгоритмів предметної області	29.09.2024	виконано
3	Дослідження предметної області використання облікових даних	15.10.2024	виконано
4	Дослідження методів спільного використання облікових даних на основі специфікацій	08.11.2024	виконано
5	Імплементація моделей та методів для вирішення проблеми використання облікових даних	20.11.2024	виконано
6	Прикладне застосування пропонованих підходів для системи управління ризиками	01.12.2024	виконано
7	Затвердження пояснювальної записки роботи завідувачем кафедри	15.12.2024	виконано

Студент – магістр _____
(підпис)

Керівник роботи _____
(підпис)

АНОТАЦІЯ

Магістерська робота: 80 с., 35 рис., 2 табл., 49 джерел.

Тема: Моделі та методи спільного використання облікових даних на основі специфікації

Об'єкт дослідження: є процеси спільного використання сертифікатів у у інформаційних мережах.

Мета роботи: розробка та оцінка підходів до виявлення та запобігання спільному використанню облікових даних у новітніх інформаційних мережах, а також створення алгоритму функціонування системи управління ризиками.

Предмет дослідження: методи та підходи виявлення і запобігання спільному використанню сертифікатів у безпекових середовищах, а також їх вплив на архітектуру та відповідні стандарти.

Результати дослідження

Результати дослідження можуть бути застосовані постачальниками послуг для покращення безпеки мереж HS2.2 через впровадження нових підходів до виявлення та запобігання спільному використанню сертифікатів

Висновок

Запропоновано алгоритм функціонування системи управління ризиками дозволить зменшити ризик несанкціонованого доступу до конфіденційних даних та підвищити рівень захисту інформації в мережах.

СЕРТИФІКАТ ОБЛІКОВИХ ДАНИХ, ІНФОРМАЦІЙНА МЕРЕЖА, ОБМІН ОБЛІКОВИМИ ДАНИМИ, ТОКЕН, АВТЕНТИФІКАЦІЯ, БЕЗПЕКА МЕРЕЖ.

ABSTRACT

Master Thesis: 80 pp., 35 fig., 2 tab., 49 sources.

Thesis Subject: Specification-Based Credential Sharing Models and Methods

Object of research: there are processes of sharing certificates in information networks.

The purpose of the work: the development and evaluation of approaches to the detection and prevention of the sharing of credentials in the latest information networks, as well as the creation of an algorithm for the functioning of the risk management system.

Research subject: methods and approaches to detect and prevent certificate sharing in secure environments, as well as their impact on architecture and relevant standards.

Research results

The research findings can be applied by service providers to improve the security of HS2.2 networks by implementing new approaches to detect and prevent certificate sharing

Conclusion

The proposed algorithm of the risk management system will reduce the risk of unauthorized access to confidential data and increase the level of information protection in networks.

ACCOUNT CERTIFICATE, INFORMATION NETWORK, ACCOUNT EXCHANGE, TOKEN, AUTHENTICATION, NETWORK SECURITY.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

AAA - authentication, authorization and accounting

ANQP - Access Network Query Protocol

SP - service providers

HS – Hotspot

CA - certicate authority

APs - access points

OSU - Online Sign Up

CRLs - certicate revocation lists

MOs - management objects

IMSI - International Mobile Subscriber Identity

IMEI MEID - International Mobile Equipment Identity and Mobile Equipment Indenter

PPS MO - PerProviderSubscription Management Object

FQDN - Fully Qualied Domain Name

RMS - Risk Management System

ВСТУП

Актуальність теми.

З розвитком мережевих технологій та збільшенням обсягу передавання конфіденційних даних, особливо в середовищі HS2.2, проблема захисту сертифікатів від спільного використання стає все більш актуальною. Незаконний доступ до сертифікатів може призвести до порушення безпеки мережі та витоку критично важливої інформації. Тому дослідження ефективних методів виявлення та запобігання спільному використанню сертифікатів є необхідним для підвищення рівня безпеки мереж HS2.2.

Розвиток мереж Wi-Fi точок доступу призвів до появи стандарту Hotspot 2.2 (HS2.2), який забезпечує зручніший та безпечніший доступ до Wi-Fi мереж. Одним із ключових елементів HS2.2 є можливість постачальників послуг надавати пристроям клієнтів облікові дані користувача/пароля або сертифікат облікових даних. Однак, ця функція може створити умови для обміну обліковими даними між пристроями в мережі, що призводить до серйозних ризиків для постачальників послуг та законних користувачів.

Стандарт HS2.2 визначає політику спільного використання лише для облікових даних користувача/пароля. Тому в цій роботі ми розглядаємо проблему обміну сертифікатом облікових даних у мережах HS2.2 та аналізуємо інформацію, необхідну для виявлення та запобігання такому обміну між законними та піратськими мобільними пристроями без SIM-карти.

На основі літературного огляду пропонується шість кандидатів підходів до вирішення проблеми обміну сертифікатом облікових даних. Кожен підхід має свої переваги, обмеження та вплив на мережі HS2.2.

Для ефективного протидії обміну сертифікатом облікових даних у мережах HS2.2 пропонується система управління ризиками, яка поєднує в собі найбільш ефективні підходи. Ця система дозволить постачальникам

послуг виявляти та запобігати обміну обліковими даними, мінімізуючи ризики для їх бізнесу та користувачів.

Зростаюча кількість пристроїв, що підключаються до мереж, та збільшення обсягів конфіденційної інформації, що передається через ці мережі, підвищують вимоги до надійності механізмів автентифікації та захисту даних. У контексті стандарту Hotspot 2.0 (HS2.2), який спрощує підключення користувачів до Wi-Fi-мереж, особливо важливим є питання безпечної передачі облікових даних. Оскільки сертифікати часто використовуються для автентифікації користувачів, їхнє спільне використання або передача від легітимного пристрою до сторонніх піратських пристроїв може призвести до серйозних порушень безпеки мережі.

Зростання кількості випадків несанкціонованого використання сертифікатів вимагає розробки нових ефективних методів виявлення та запобігання спільному використанню облікових даних. Проблема ускладнюється тим, що звичайні підходи до автентифікації, такі як ім'я користувача та пароль, не забезпечують належного рівня захисту в умовах мереж HS2.2. Сертифікати можуть бути легко передані між пристроями, що створює ризик компрометації мережі.

Актуальність теми дослідження полягає в тому, що сьогоднішні механізми автентифікації потребують удосконалення для забезпечення безпеки у середовищах, де сертифікати використовуються як ключові облікові дані. Розробка нових підходів, таких як автентифікація на основі токенів, відбитків пальців пристроїв та віддаленого моніторингу, сприятиме захисту даних і зменшенню ризиків несанкціонованого доступу до ресурсів мереж HS2.2. Таким чином, дослідження цієї проблеми відповідає сучасним викликам у сфері кібербезпеки та управління ідентифікацією в умовах високої динаміки технологічного розвитку.

Мета дослідження - розробка та оцінка підходів до виявлення та запобігання спільному використанню облікових даних у новітніх

інформаційних мережах, а також створення алгоритму функціонування системи управління ризиками.

Об'єкт дослідження - є процеси спільного використання сертифікатів у інформаційних мережах.

Предмет дослідження – методи та підходи виявлення і запобігання спільному використанню сертифікатів у безпекових середовищах, а також їх вплив на архітектуру та відповідні стандарти.

Відповідно до мети роботи було сформовано наступні **задачі:**

- Визначити проблему спільного використання сертифікатів у середовищі HS2.2;
- Розробити та запропонувати кілька підходів для виявлення та запобігання спільному використанню сертифікатів;
- Оцінити вплив запропонованих підходів на архітектуру та стандарт HS2.2;
- Проаналізувати переваги та недоліки кожного підходу;
- Розробити алгоритм функціонування системи управління ризиками для постачальників послуг, яка б інтегрувала запропоновані підходи.

Методи дослідження.

У процесі дослідження використовувалися наступні методи:

- Аналіз і систематизація існуючих підходів до автентифікації на основі сертифікатів.

- Моделювання процесів виявлення та запобігання спільному використанню облікових даних.

- Емпіричний аналіз можливостей технічної реалізації запропонованих підходів.

- Оцінка впливу розроблених рішень на безпеку та продуктивність мереж HS2.2.

Наукова новизна отриманих результатів полягає в розробці нових підходів на основі токенів для виявлення та запобігання спільному використанню облікових даних у мережах HS2.2.

Практичне значення магістерської роботи полягає в тому, що результати дослідження можуть бути застосовані постачальниками послуг для покращення безпеки мереж HS2.2 через впровадження нових підходів до виявлення та запобігання спільному використанню сертифікатів. Запропоновано алгоритм функціонування системи управління ризиками дозволить зменшити ризик несанкціонованого доступу до конфіденційних даних та підвищити рівень захисту інформації в мережах.

Структура магістерської роботи. Робота складається зі вступу, трьох розділів та висновків. Загальний обсяг роботи становить 80 сторінок, і містить 35 рисунків, 2 таблиць, список використаних джерел із 49 найменувань.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ВИКОРИСТАННЯ ОБЛІКОВИХ ДАНИХ

1.1. Особливості спільного використання облікових даних

В останні роки мережі точок доступу Wi-Fi стають все більш популярними як засіб забезпечення доступу пристроїв до послуг пакетної передачі даних, наприклад Інтернету, що пропонуються постачальниками послуг (SP). Hotspot 2.0 (HS2.0) — це новий стандарт, визначений Wi-Fi Alliance для мереж WiFi наступного покоління [18]. Перевага цього нового стандарту полягає в тому, що мобільні пристрої можуть автоматично встановлювати безпечне з'єднання з мережами Wi-Fi. Крім того, HS2.0 забезпечує плавний роумінг в одній мережі Wi-Fi і між мережами Wi-Fi. Наприклад, для тих людей, які пересуваються в міському просторі, їхні мобільні пристрої можуть автоматично отримати безпечний доступ до передплаченої послуги, яку пропонує один конкретний SP через різні мережі точок доступу, у випадку, якщо мережі точок доступу та мобільні пристрої є HS2.0 і ці мережі підтримують цей SP.

Hotspot 2.0 (або HS2.0) - це сучасний стандарт бездротового підключення, який надає більш безпечний та зручний спосіб підключення до громадських Wi-Fi мереж. Він розроблений для того, щоб автоматизувати процес автентифікації та спростити управління доступом до Інтернету.

Основні відмінності Hotspot 2.0 від звичайного Wi-Fi полягають в наступному:

- Автоматична автентифікація: Замість того, щоб вручну вводити логін та пароль кожного разу, коли ви підключаєтесь до нової мережі, HS2.0 дозволяє автоматично ідентифікувати вас за допомогою ваших облікових даних, збережених на вашому пристрої (наприклад, SIM-картці або обліковому записі мобільного оператора).

- Єдина точка доступу: HS2.0 створює єдину точку доступу для всіх мереж, що підтримують цей стандарт. Це означає, що вам не потрібно шукати та підключатися до різних мереж в різних місцях.

- Підвищена безпека: HS2.0 використовує більш надійні методи шифрування для захисту ваших даних під час підключення до громадських мереж.

- Спрощений роумінг: Якщо ви постійно переміщаєтесь між різними місцями, HS2.0 забезпечує плавний перехід між мережами без переривання з'єднання.

Особливості роботи Hotspot 2.0:

- Ви підключаєтесь до мережі: Коли ви входите в зону дії мережі HS2.0, ваш пристрій автоматично виявляє її та починає процес підключення.

- Автентифікація: Ваш пристрій використовує збережені облікові дані для ідентифікації себе в мережі.

- Доступ до Інтернету: Після успішної автентифікації ви отримуєте доступ до Інтернету.

Переваги Hotspot 2.0:

- Зручність: Більш швидке та просте підключення до громадських мереж.

- Безпека: Вищий рівень захисту ваших даних.

- Сумісність: Працює з різними операційними системами та пристроями.

Цей стандарт вже активно використовується багатьма мобільними операторами та провайдерами Wi-Fi послуг в аеропортах, готелях, кафе та інших громадських місцях.

Hotspot 2.2 є подальшим розвитком цього стандарту і пропонує додаткові функції та покращення, такі як підтримка нових протоколів безпеки та більш ефективне використання мережевих ресурсів.

Наразі цей стандарт має дві версії: випуск 1 (HS2.1) і випуск 2 (HS2.2). HS2.1 визначає автоматичний і безпечний процес підключення за умови, що

мобільні пристрої вже мають облікові дані для доступу до послуги підписки певного SP. Але джерело облікових даних невідоме, яке може бути активно налаштовано користувачем, наприклад ім’я користувача/пароль, або пасивно надане SP. Стандарт HS2.2 не лише визначає джерело та тип облікових даних, але й визначає процес надання облікових даних на додаток до HS2.1. Тим не менш, керування обліковими даними після надання рідко обговорюється, наприклад, спільне використання облікових даних. Спільний доступ до облікових даних означає, що облікові дані користувача для послуг передплати, які пропонуються SPs, надаються іншим користувачам або пристроям як добровільно, так і несвідомо. Тобто облікові дані можна отримати з одного пристрою та використати шахрайським шляхом на інших пристроях, що може призвести до зламаних облікових записів. Отже, спільний доступ до облікових даних буде потенційною проблемою для мереж HS2.0.

Крім мереж HS2.0 існують також мережі HS1.0. Мережа HS1.0 є комерційною мережею точок доступу або мережею точок доступу приєднаних порталів [11]. Коли користувачі підключаються до мережі HS1.0, вони перенаправляються на сторінку для оплати, щоб отримати доступ до мереж. Провайдери послуг не надають жодних облікових даних мобільним пристроям, і користувачам не потрібно надавати облікові дані для автентифікації. Іншими словами, у цій мережі точок доступу з оплатою за користування немає облікових даних. Отже, спільний доступ до облікових даних не є проблемою для мереж HS1.0. В цій роботі розглядається лише проблему спільного використання облікових даних стосовно облікових даних, наданих SP, тобто для мереж HS2.2.

Спільне використання облікових даних є загальновідомою проблемою, яка існує та вивчається протягом багатьох років [5, 8, 30]. Загалом спільний доступ до облікових даних означає повторне використання облікових даних одного користувача в загальній підписці іншим користувачем і отримання доступу до відповідної служби. Пара ID користувача та пароля є обліковими

даними, які зазвичай використовуються. Загалом існує два основних типи спільного використання облікових даних для кожного облікового запису користувача.

Перший тип — випадковий обмін. Випадковий обмін означає, що користувачі охоче діляться обліковими даними з кимось іншим, зокрема друзями та членами родини. У цьому випадку кожен SP повинен мати чітку політику щодо кількості користувачів або пристроїв, яким дозволено використовувати однакові облікові дані, інакше облікові дані можуть бути надані несанкціонованим способом. Наприклад, Netflix має план передплати, який дозволяє одночасно використовувати кілька пристроїв одному абоненту. На відміну від Netflix, Spotify дозволяє використовувати сімейний план, за яким одна підписка може бути спільною з 6 членами однієї сім'ї. Однак випадковий обмін може призвести до значних негативних наслідків для цих постачальників послуг (SP). Для Netflix, як наслідок спільного доступу до облікових даних, один абонент може заплатити, поділитися паролем, необхідним для доступу до послуги з кимось іншим (і вартість підписки зазвичай коштує); обидва разом можуть одночасно отримати доступ до Netflix, якщо вони не перевищують ліміт пристроїв для спільного використання. Шкода для Netflix у цьому конкретному випадку полягає в тому, що у них є один платний передплатник і один не платить. Клієнти, які не платять, отримують неавторизований доступ замість того, щоб отримати законну підписку. Для «Сімейного» плану Spotify визначення сім'ї є розпливчастим, і ніщо не заважає користувачам вільно ділитися підпискою далі. Крім того, SP може бути важко відрізнити випадковий обмін інформацією від зловмисного.

Другий тип — це крадіжка облікових даних. Як випливає з назви, викрадення облікових даних означає, що невідомі законним користувачам облікові дані викрадаються зловмисниками за допомогою різних методів, таких як фішинг, зловмисне програмне забезпечення, атаки грубою силою тощо [25]. Об'єднуючи облікові дані для служб підписки, зловмисники

можуть отримати незаконний доступ до служб або продати дійсні облікові записи групі людей, що створює величезний ризик як для постачальників послуг, так і для законних користувачів. Таким чином, як реальне занепокоєння постачальників послуг, незалежно від випадкового обміну з друзями чи облікових даних, викрадених зловмисниками, обмін обліковими даними може призвести до втрати доходу, втрати потенційних клієнтів і, відповідно, вплинути на репутацію бренду.

Спільне використання облікових даних для HS2.2

В даний час запропоновано деякі апаратні та програмні рішення для запобігання загальному обміну обліковими даними, наприклад генерування унікального пароля [1], використання токена під час процесу автентифікації [19, 29], застосування систем біометричної автентифікації для отримання унікальної ідентифікації користувача інформація [21] тощо. Однак немає відповідних дослідницьких робіт щодо спільного використання облікових даних у середовищі HS2.2.

У стандарті HS2.2 облікові дані імені користувача/паролю та облікові дані сертифіката є єдиними двома типами облікових даних, які можуть надаватися SP. SP має можливість дозволити спільний доступ до імені користувача/паролю між декількома пристроями для доступу до спільного сервера в мережах HS2.2. З цією метою разом із ім’ям користувача/паролем мобільному пристрою також надається відповідна конфігурація підписки, яка вказує, чи облікові дані імені користувача/паролю можна використовувати лише на мобільному пристрої, який підписаний, або також на інших мобільних пристроях користувача [18]. На відміну від спільного використання облікових даних імені користувача/пароля, стандарт HS2.2 не вказує опцію для SP, щоб дозволити спільний доступ до облікових даних, якщо використовуються сертифікати. Однак це не означає, що незаконний обмін обліковими даними на основі сертифікатів не відбуватиметься в мережах HS2.2. Тому існує потреба вивчити проблему спільного

використання облікових даних сертифіката та знайти відповідне рішення для цього нового стандарту.

1.2. Загальна архітектура та опис стандарту HS 2.2

В даній роботі основна увага буде зосереджена лише на Hotspot 2.0 Release 2. І HS2.1, і HS2.2 забезпечують вибір мережі та функції безпечного доступу. HS2.2 зосереджується на покращенні взаємодії з користувачем під час підключення до мережі гарячої точки, надаючи ще дві функції на додаток до HS2.1, а саме онлайн-реєстрацію (OSU) і налаштування політики. Наприклад, іноді людям, які виходять за межі початкового діапазону підключеної мережі та хочуть зберегти безперервне бездротове мережеве з'єднання, потрібно вручну вибрати нову мережу SP, надати деяку особисту інформацію, наприклад адресу електронної пошти, і зареєструвати обліковий запис, щоб отримати доступ до знову Інтернет. Однак, застосовуючи стандарт HS2.2, весь описаний вище трудомісткий процес може бути автоматично виконаний самим мобільним пристроєм.

У стандарті HS2.2 мобільні пристрої можна розділити на дві категорії. Один — це пристрої, що підтримують SIM-карту, з обліковими даними SIM-карти, а інший — це пристрої, що не підтримують SIM-карту, які мають облікові дані для імені користувача/пароля та облікові дані сертифіката. Програмне ім'я користувача/пароль можна відносно легко отримати або скопіювати з пристрою. Апаратні облікові дані SIM важко отримати або скопіювати з пристрою. У цій роботі я зосереджусь на мобільних пристроях без SIM-карти.

На рисунку 1.1 зображено загальну архітектуру HS2.2, яка дозволяє декільком провайдерам надавати послуги мобільним пристроям без SIM-карти на одній і тій же мережі гарячої точки. Архітектуру в основному можна розділити на дві частини: мережа гарячих точок і мережі SP.

Як показано на рисунку нижче, кожна мережа точки доступу складається з однієї або кількох точок доступу (AP) для забезпечення бездротового підключення до мобільних пристроїв, маршрутизаторів і сервера AAA (автентифікації, авторизації та обліку) для локальної автентифікації або ретрансляції повідомлень між мобільними пристроями та Сервер SPs AAA. Для кожної мережі точок доступу є оператор точки доступу, який відповідає за розгортання та роботу мережі. Оператор точки доступу може бути тією ж організацією, що й SP. У специфікації HS2.2 існує сервер Access Network Query Protocol (ANQP), який знаходиться на точці доступу з підтримкою Passpoint для надання мобільним пристроям інформації про цю мережу точки доступу та SP, які вона підтримує.

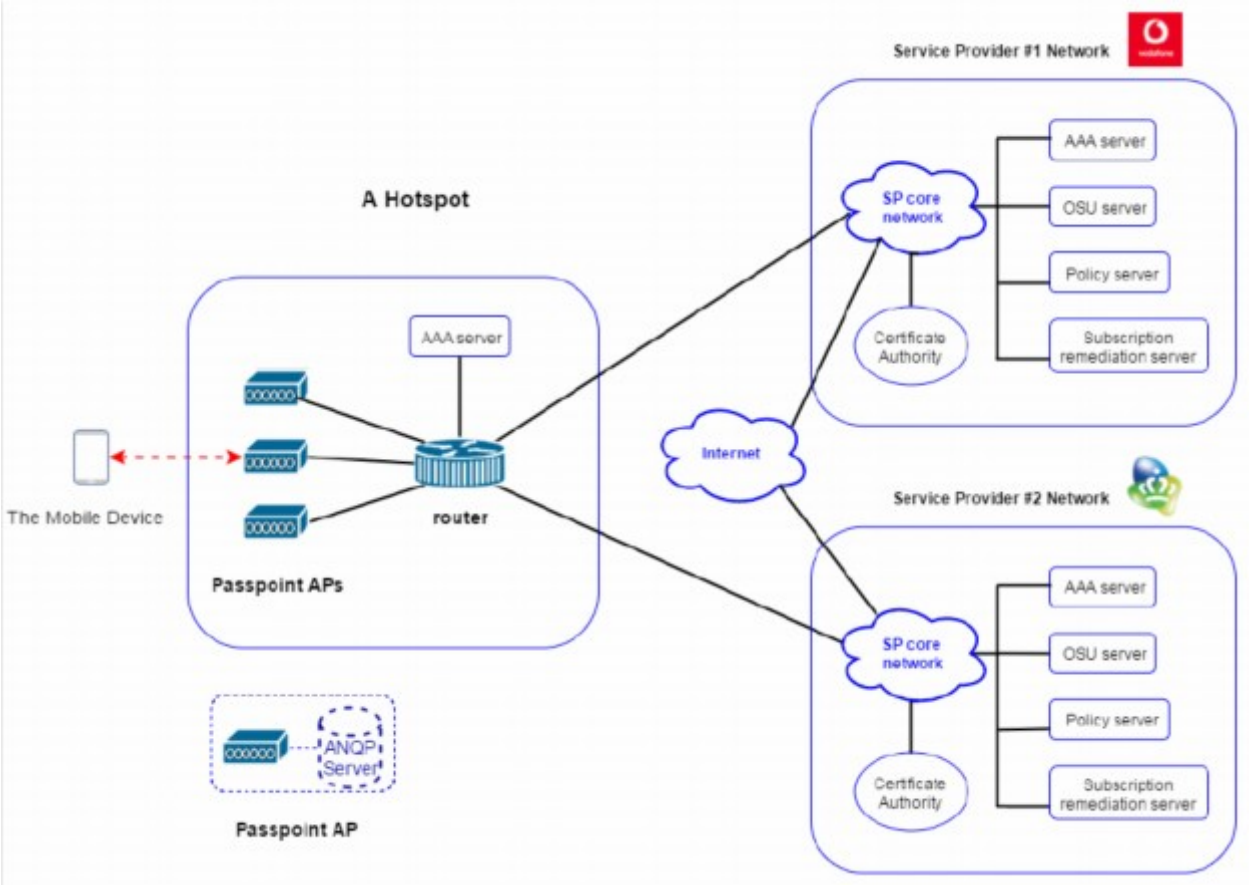

Рис. 1.1. Загальна архітектура HS2.2

Кожна мережа SPs має:

- сервер OSU

– Сервер OSU відповідає за реєстрацію нових абонентів і надання їм облікових даних у процесі OSU. Відповідно до реєстраційної інформації, наданої користувачем, сервер OSU визначає тип облікових даних для надання: ім’я користувача/пароль або сертифікат. У разі надання імені користувача/пароля сервер OSU безпосередньо створить і надішле цей тип облікових даних на пристрій. Крім того, сервер OSU зв’яжеться з центром сертифікації (CA), щоб видати сертифікат клієнта, а потім надіслати сертифікат на пристрій.

• сервер AAA

– Сервер AAA відповідає за облік використання послуги та автентифікацію передплатників. Перш ніж клієнтський пристрій отримає доступ до мережі, йому необхідно виконати взаємну автентифікацію із сервером AAA SP на етапі безпечного доступу.

• сервер політики

– Під час процесу OSU сервер політики відповідає за надання інформації про політику для вибору та оновлення мережі. Інформація про політику відноситься до політики домашнього SP, включаючи переваги або пріоритет партнерів з роумінгу, політику для оновлення, IP-протокол, номер порту тощо. На етапі безпечного доступу сервер політики може оновити політику інформації на мобільний пристрій відповідно до потреб ІП.

• сервер відновлення підписки

– Сервер відновлення підписки відповідає за оновлення наданої інформації, наприклад, оновлення облікових даних і підписки, а також виправлення проблем, відомих SP, наприклад, закінчення терміну дії сертифіката та прострочений рахунок. На етапі захищеного доступу сервер AAA може попросити мобільний пристрій зв’язатися з сервером відновлення підписки з метою відновлення підписки. Виправлення підписки можна розділити на два типи: активне оновлення та пасивне виправлення. Активне оновлення стосується процесу, ініційованого мобільним пристроєм. Пасивне виправлення стосується процесу, ініційованого SP.

• CA (certicate authority)

– CA — це «набір комп’ютерного обладнання, програмного забезпечення та людей, які ним керують» [18]. CA виконує такі функції:

1. Видача сертифікатів. CA використовує інформацію, надану пристроями, для створення сертифікатів і підписання сертифікатів. Детальний процес буде описано в розділі 3.2.

2. Ведення інформації про статус виданих сертифікатів, наприклад, прострочений або відкликаний

3. Видача списків відкликаних сертифікатів (CRL), щоб вказати стани відкликаних сертифікатів, які той самий ЦС видав раніше.

4. Періодична публікація CRL.

Окрім сервера AAA, усі інші сервери називаються серверами підписки. На основі цієї архітектури, наприклад, через цю мережу точки доступу можна отримати доступ до кількох постачальників послуг, таких як Vodafone і KPN. Щоб підписатися на послугу, запропоновану одним із них, мобільний пристрій спочатку підключається до однієї з точок доступу до мережі hotpsot, а потім зв’язується з мережею SP через цю точку доступу.

1.3. Особливості підключення послуг за допомогою стандарту HS 2.2

Коли користувач хоче вперше підписатися на послугу, яку пропонує один призначений SP, його мобільний пристрій має пройти всі чотири етапи, як показано на рисунку 1.2, щоб отримати облікові дані від цього SP. Чотири етапи описані нижче.

1) Виявлення: мобільний пристрій вибирає точку доступу, з якою зв’язується, і запитує інформацію про SP для вибору мережі.

2) Реєстрація: користувач надає особисту інформацію та створює обліковий запис у вибраному SP.

3) Надання: SP встановлює облікові дані та інформацію про політику підписки та надає їх на мобільний пристрій. На цьому етапі можна надати лише ім’я користувача/пароль і облікові дані сертифіката.

4) Безпечний доступ: мобільний пристрій взаємно автентифікується та зв’язується з мережею SPs, а потім успішно отримує доступ до послуги підписки.

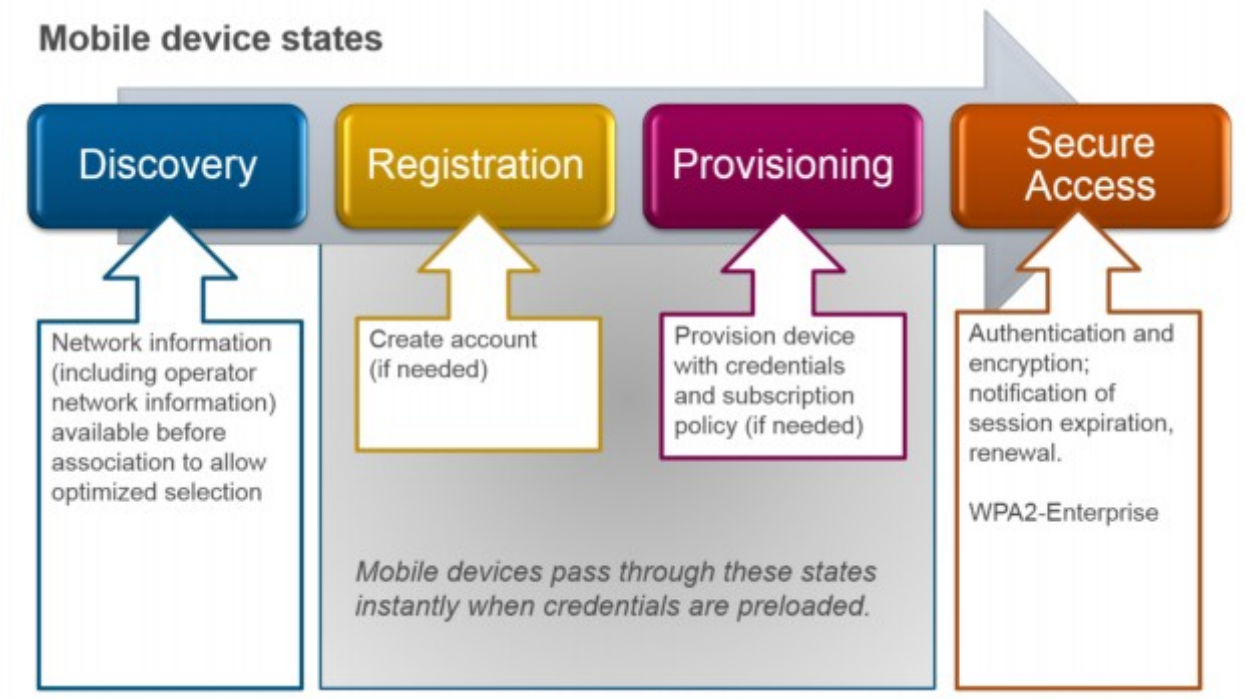

Рис. 1.2. Етапи підключення за допомогою HS 2.2

Поєднання етапів реєстрації та надання називається процесом OSU (Online Sign Up). Кожного разу, коли він повторно підключається до мережі, мобільний пристрій має пройти лише етапи виявлення та безпечного доступу, оскільки пристрій уже отримав облікові дані раніше. Процес OSU завжди пропускається, якщо є дійсні підготовлені облікові дані, за винятком того, що мобільний пристрій використовує клієнтські сертифікати, які попередньо підготовлені іншим SP або під час виробничого процесу. На рисунку 1.2 показано весь процес першої підписки, яка дозволяє мобільному пристрою мати автоматичний безпечний доступ до мережі Wi-Fi.

Рисунок 1.3 просто ілюструє обмін повідомленнями між мобільним пристроєм, мережею точки доступу та мережею SP протягом усього робочого процесу.

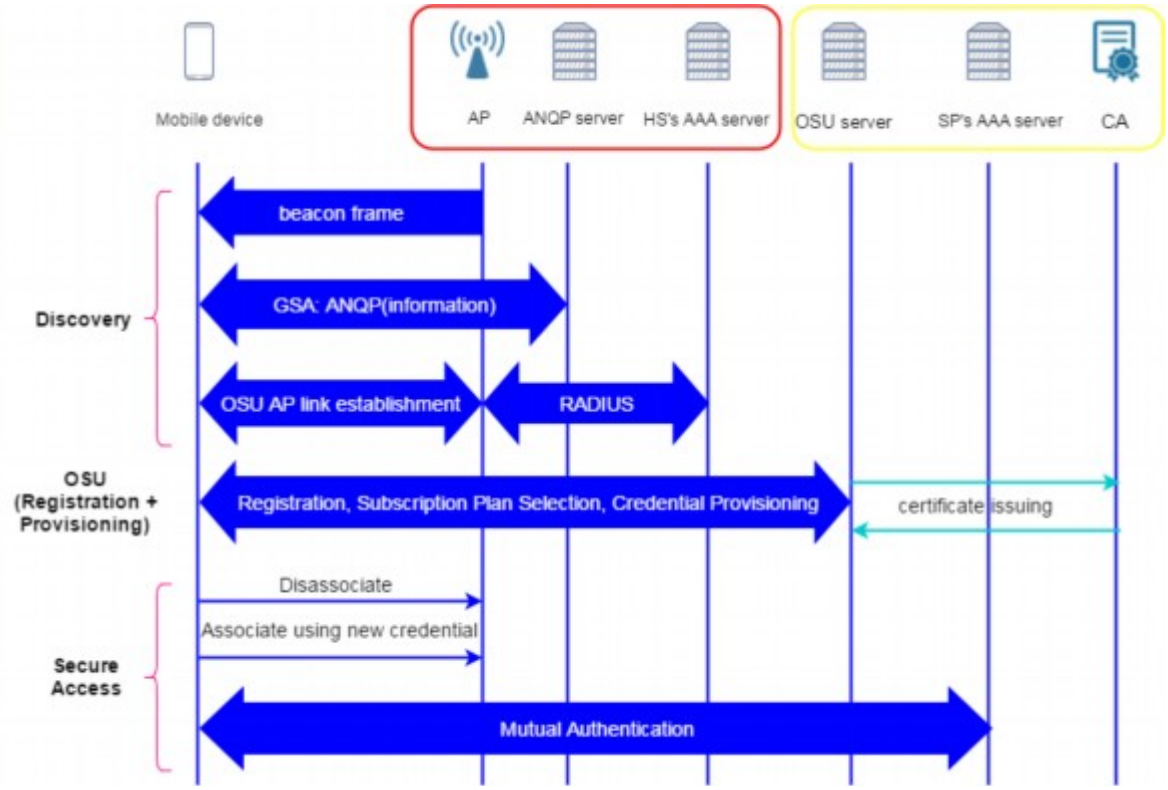

Рис. 1.3. Загальна блок-схема HS2.2

Як показано на схемі, червоний прямокутник позначає мережу точки доступу, а жовтий — мережу SP. На етапі виявлення, оскільки в одній точці доступу може бути більше однієї фізичної точки доступу, мобільний пристрій без SIM-карти без облікових даних спочатку повинен знайти відповідну точку доступу для асоціації та вибрати SP, який підтримує OSU, запитуючи інформацію за допомогою сервер ANQP. Механізм мережі шифрування рівня 2 (OSEN) із автентифікацією лише на сервері OSU захищає безпечний зв'язок і зв'язок між мобільним пристроєм і точкою доступу. Асоціація OSEN складається з відкритої автентифікації та асоціації 802.11, анонімної автентифікації EAP-TLS і протоколу 4-сторонньої автентифікації (4WHS). По суті, мобільний пристрій спочатку підключається до сервера AAA мережі для локальної автентифікації, а потім установлює

ключі шифрування з точкою доступу для захисту з'єднання Wi-Fi перед підключенням до сервера OSU мережі SP.

На етапі реєстрації мобільний пристрій підключається до сервера OSU і підписується на призначений SP через мережу точки доступу. Користувачеві потрібно вибрати план передплати, надати контактну інформацію, платіжну інформацію та ідентифікаційну інформацію свого пристрою, яка міститься в об'єктах управління (MO) DevInfo та DevDetail. Як зазначено в специфікації керування пристроями OMA, DevInfo MO та DevDetail MO є двома стандартними об'єктами в дереві керування пристроями OMA [2]. На рисунку 1.4 показана загальна структура OMA Device Management Tree.

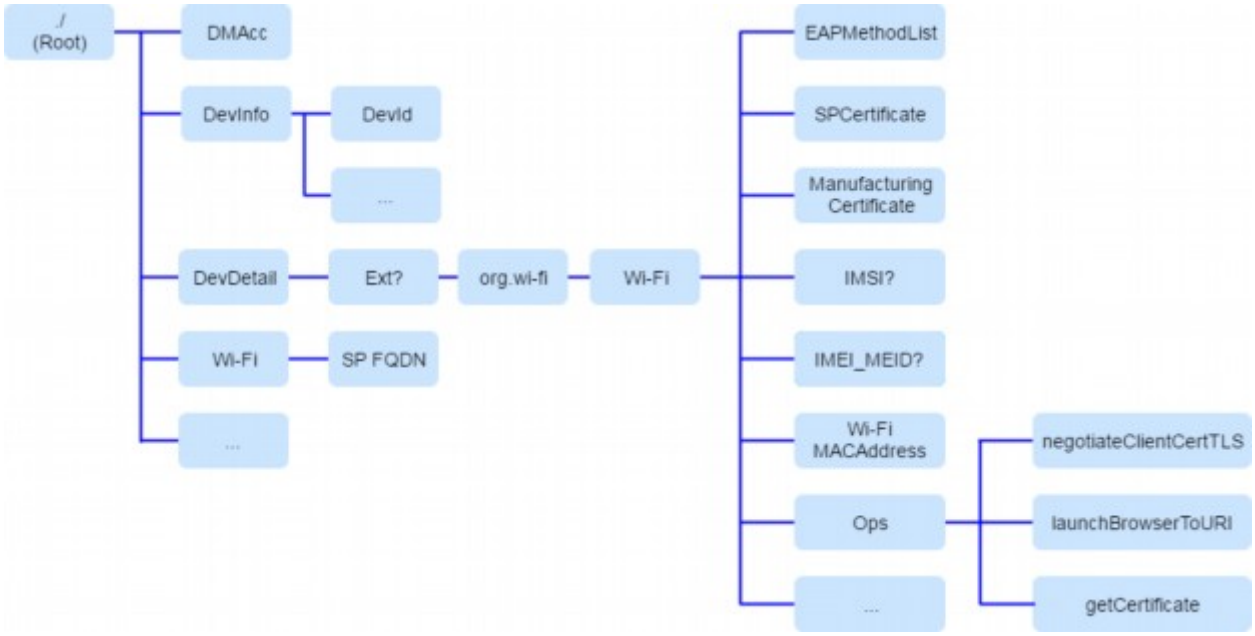

Рис. 1.4. Структура OMA Device Management Tree

Знак питання на малюнку означає нуль або одне повторення. Ідентифікаційна інформація мобільного пристрою, наприклад:

- DevId: кінцевий вузол DevInfo MO, що містить глобальний унікальний ідентифікатор пристрою.
- IMSI (International Mobile Subscriber Identity): кінцевий вузол DevDetail MO, що містить IMSI мобільного пристрою. IMSI — унікальний

номер для ідентифікації користувача стільникової мережі. Він зберігається на SIM-картці.

- IMEI MEID (міжнародний ідентифікатор мобільного обладнання та ідентифікатор мобільного обладнання): кінцевий вузол DevDetail MO, що містить IMEI або MEID мобільного пристрою. IMEI — це ідентифікатор обладнання для мобільного пристрою 3GPP (3rd Generation Partnership Project) або мобільного пристрою з подвійним режимом 3GPP2/3GPP [18]. MEID — це ідентифікатор обладнання для мобільного пристрою 3GPP2 [18].
- MAC-адреса Wi-Fi: кінцевий вузол DevDetail MO, що містить MAC-адресу мобільного пристрою.

Зауважимо, що кінцеві вузли IMSI та IMEI MEID є необов'язковими для мобільних пристроїв без SIM-карт; ці вузли є обов'язковими для мобільних пристроїв, що мають SIM-карти [18]. Між мобільним пристроєм і сервером SPs OSU дозволений лише трафік HTTPS.

На етапі підготовки на основі інформації, наданої користувачем, сервер OSU визначає тип облікових даних для надання (або ім'я користувача/пароль, або сертифікат клієнта кожного разу), установлює інформацію про підписку, що міститься в об'єкті керування підпискою PerProvider (PPS MO), і доставляє цю інформацію призначеному користувачеві. Наданий PPS MO буде додано до дерева керування пристроями OMA як кінцевий вузол під кореневим вузлом. PPS MO складається з кількох кінцевих вузлів, що містять інформацію про Home SP, політику підписки, інформацію про керування та облікові дані. Наприклад, PPS MO містить такі кінцеві вузли:

• UpdateIdentifier: UpdateIdentifier — це число, встановлене SP для визначення версії PPS MO. Він скорочено називається PPS MO ID. Значення PPS MO ID змінюватиметься щоразу, коли змінюється будь-яка інформація в PPS MO.

• Policy: цей внутрішній вузол містить політику Home SP, встановлену сервером політики. Зміст політики представлено в наступному розділі.

• SubscriptionUpdate: цей внутрішній вузол містить інформацію, пов’язану з оновленнями підписки та виправленням підписки, наприклад UpdateInterval і UpdateMethod. UpdateInterval — це параметр, який визначає частоту, яку мобільний пристрій має перевіряти в SP для оновлення конфігурації підписки. UpdateMethod — це параметр, який визначає метод, що використовується для оновлення підписки, наприклад, OMA DM або SOAP XML.

• HomeSP: цей внутрішній вузол містить інформацію, пов’язану з Home SP, таку як SSID (ідентифікатор набору послуг), зрозуміле ім’я, FQDN (повне кваліфіковане ім’я домену) тощо.

• SubscriptionParameters: цей внутрішній вузол містить інформацію, пов’язану зі службою підписки, таку як дата створення підписки, дата закінчення терміну дії, накопичені ліміти використання даних (мегабайти) або час (хвилини).

• Облікові дані: цей внутрішній вузол містить інформацію, пов’язану з обліковими даними підписки, наприклад дату створення облікових даних, термін дії, ім’я користувача/пароль або цифровий сертифікат тощо. Зауважте, що цифровий сертифікат тут є відбитком облікових даних сертифіката.

У разі надання облікових даних сертифіката мобільний пристрій згенерує пару відкритий/приватний ключ, а ЦС підпише відкритий ключ для видачі сертифіката клієнта на мобільний пристрій. За бажанням сервер політики надає інформацію про політику на мобільний пристрій, яка також буде додана до PPS MO. Тому після процесу OSU інфраструктура SPs зберігає інформацію про абонентів у власному реєстрі, а мобільний пристрій має PPS MO, сертифікат сервера OSU, сертифікат довірчого кореневого ЦС сервера AAA SPs та/або сертифікат клієнта та пару відкритих/приватних ключів. у разі надання облікових даних сертифіката. Для ясності, облікові дані імені користувача/паролю посилаються на PPS MO, що містить ім’я користувача/пароль, інформацію про підписку та інформацію про політику.

Облікові дані сертифіката стосуються комбінації PPS MO та сертифіката клієнта. Сертифікат клієнта зберігається на мобільному пристрої.

На етапі безпечного доступу, використовуючи надані облікові дані, ідентифікаційну інформацію пристрою та/або цифровий підпис у разі надання облікових даних сертифіката, мобільний пристрій зв'язується та взаємно автентифікується з сервером AAA SPs і може успішно отримати доступ до служби, яку користувач підписався.

1.4. Етапи повторного під'єднання користувачів та методи автентифікації

На рисунку 1.5 показано кроки асоціації та автентифікації авторизованого клієнтського пристрою, коли він повторно підключається до мережі SP.

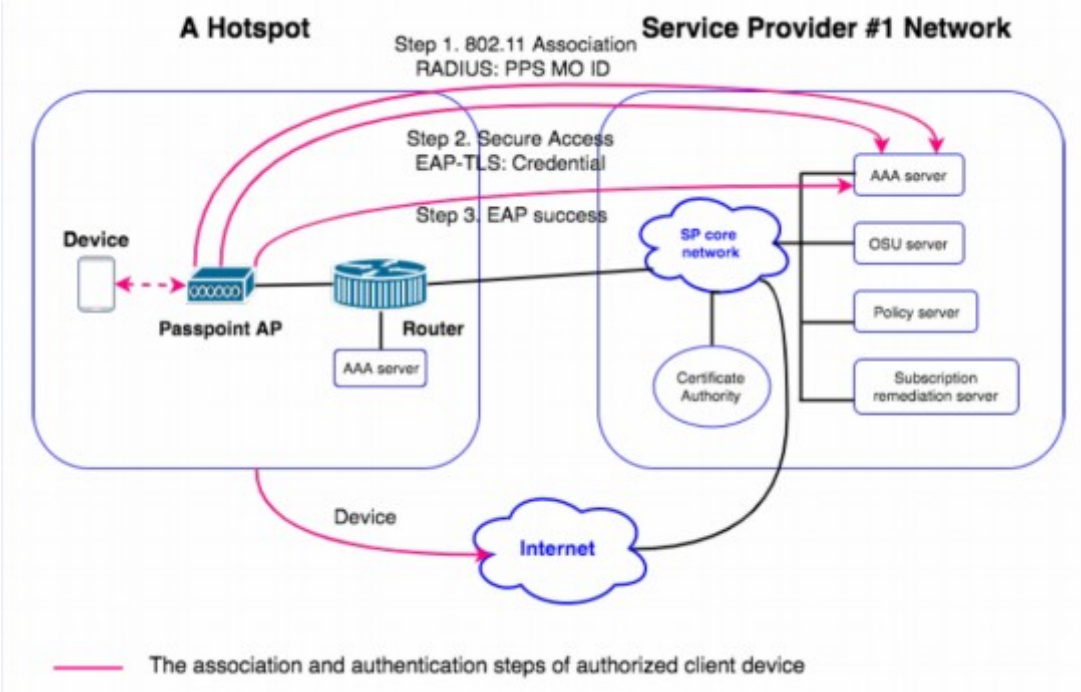

Рис. 1.5. Етапи асоціації та автентифікації повторного підключення

Як згадувалося вище, у будь-який момент, коли він повторно підключається, мобільному пристрою потрібно лише пройти через етапи

виявлення та безпечного доступу, якщо є дійсні облікові дані, надані призначеним SP раніше. На етапі виявлення мобільний пристрій використовує інформацію про підписку та політику, налаштовану в PPS MO, повторно підключаючись до мережі Home SP через мережу точки доступу. Як показано на рисунку 1.5, крок 1, пристрій підключається до точки доступу, а потім через точку доступу підключається до сервера AAA мережі SP. У цей момент точка доступу передає PPS MO ID пристрою на сервер AAA за допомогою протоколу RADIUS. Після цього мобільний пристрій безпосередньо переходить у стан безпечного доступу, пропускаючи процес OSU. Використовуючи ідентифікаційну інформацію пристрою та облікові дані імені користувача/паролю або облікові дані сертифіката з цифровим підписом, мобільний пристрій може пройти взаємну автентифікацію та знову отримати доступ до послуги підписки, як показано на кроці 2 і 3. Під час асоціації та автентифікації з SPs AAA на сервері SP може вимагати від мобільного пристрою зв’язатися з сервером відновлення підписки для оновлення конфігурації підписки, угоди про надання послуг, платіжної інформації тощо.

На рисунку 1.5 показано кроки асоціації та автентифікації авторизованого клієнтського пристрою, коли він повторно підключається до мережі SP.

1.4.1. Безпечна автентифікація та відкрита автентифікація

У HS2.2 автентифікацію між мобільним пристроєм і точкою доступу можна розділити на дві групи: безпечну автентифікацію та відкриту автентифікацію. Безпечна автентифікація включає анонімну автентифікацію та взаємну автентифікацію між клієнтським пристроєм і сервером автентифікації. Обидва типи засновані на методах розширюваного протоколу автентифікації (EAP). Використання різних методів EAP, таких як EAP-TLS, EAP-TTLS, EAP-SIM тощо, пов’язане з типом облікових даних. Для тих мобільних пристроїв, які мають облікові дані типу сертифіката,

використовуватиметься EAP-TLS. Для тих мобільних пристроїв, які мають облікові дані типу імені користувача та пароля, використовуватиметься EAP-TTLS. Анонімна автентифікація EAP-TLS означає, що лише сервер автентифікується на клієнтському пристрої (автентифікація на стороні сервера). Клієнтський пристрій не може пройти автентифікацію на сервері AAA (автентифікація на стороні клієнта), оскільки на пристрої немає облікових даних, які можна надати серверу AAA для перевірки. Тому анонімна автентифікація EAP-TLS використовується лише на етапі виявлення для першої підписки. Взаємна автентифікація означає, що слід виконувати як автентифікацію на стороні сервера, так і автентифікацію на стороні клієнта. На даний момент мобільний пристрій уже отримав облікові дані від сервера OSU і може надати облікові дані серверу AAA для перевірки. Отже, взаємна автентифікація використовується на етапі безпечного доступу як для першої підписки, так і для повторного підключення.

Рисунок 1.6 ілюструє потік повідомлень взаємної автентифікації між мобільним пристроєм і сервером AAA SP. На початку автентифікації ідентифікаційна інформація користувача клієнтського пристрою запитуватиметься сервером AAA SP1. Відповідно до протоколу автентифікації EAP-TLS ідентифікація визначається з полів subject або subjectAltName у сертифікаті пристрою [27]. У стандарті HS2.2 інформація, включена в поле теми або розширення subjectAltName, отримується з DevInfo і DevDetail MO [18]. Таким чином, ідентичність, необхідна серверу AAA, виводиться з інформації в DevInfo та DevDetail MO. Після отримання посвідчення сервер AAA запитує клієнтський пристрій почати розмову EAP-TLS за допомогою пакета EAP-Request. Далі клієнтський пристрій відповідає серверу AAA пакетом EAP-Response, що містить повідомлення привітання клієнта. Перш ніж клієнтський пристрій надсилає свій сертифікат клієнта на сервер AAA для перевірки, він спочатку автентифікує сервер AAA за допомогою сертифіката сервера AAA та повідомлення про перевірку, підписаного закритим ключем сервера AAA (автентифікація на стороні

сервера), а потім сервер AAA автентифікує пристрій за допомогою сертифіката клієнта та повідомлення перевірки, підписаного закритим ключем клієнтського пристрою (автентифікація на стороні клієнта). Наприкінці автентифікації EAP-TLS клієнтський пристрій має домовитися з сервером AAA, щоб узгодити набір шифрів, включаючи алгоритм шифрування, алгоритм хешування тощо.

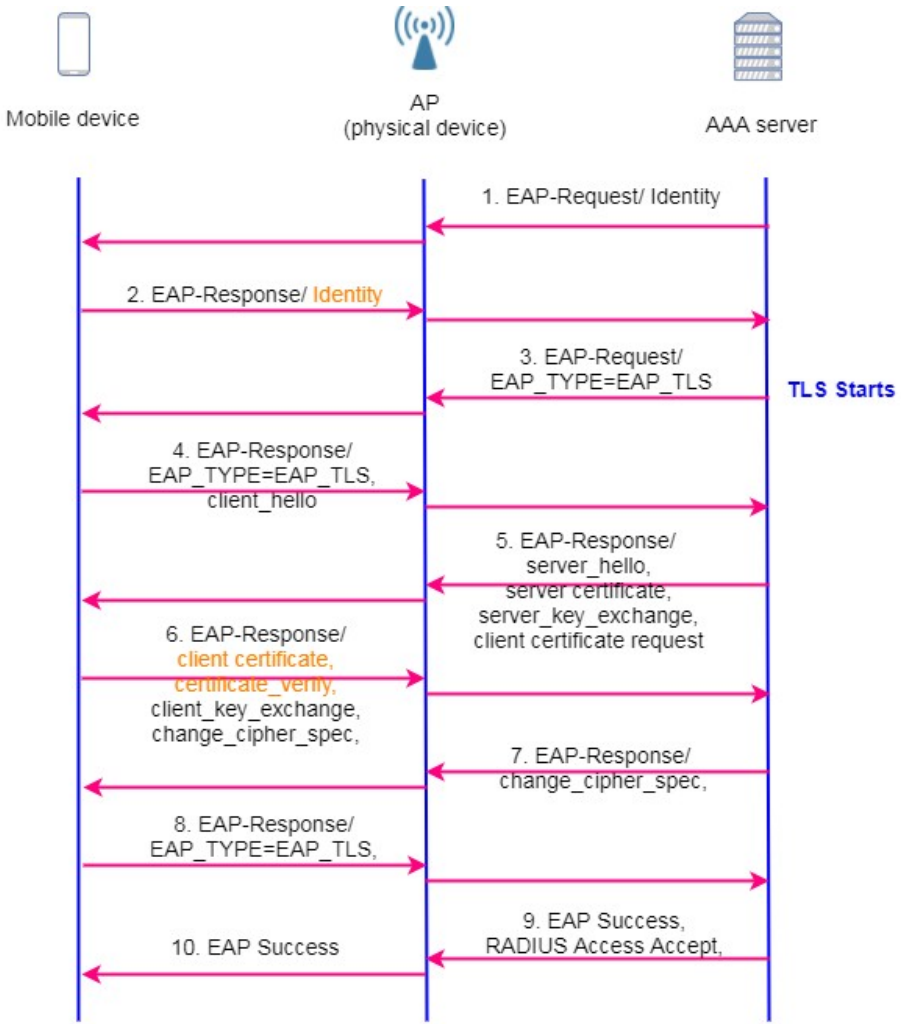

Рис. 1.6. Потік повідомлень взаємної автентифікації EAP-TLS

Окрім безпечної автентифікації, точка доступу має можливість використовувати відкриту автентифікацію з клієнтським пристроєм. З'єднання між клієнтським пристроєм і точкою доступу буде відкритим і незашифрованим. Рисунок 1.7 ілюструє робочий процес відкритої

автентифікації 802.11 і асоціації між клієнтським пристроєм і точкою доступу.

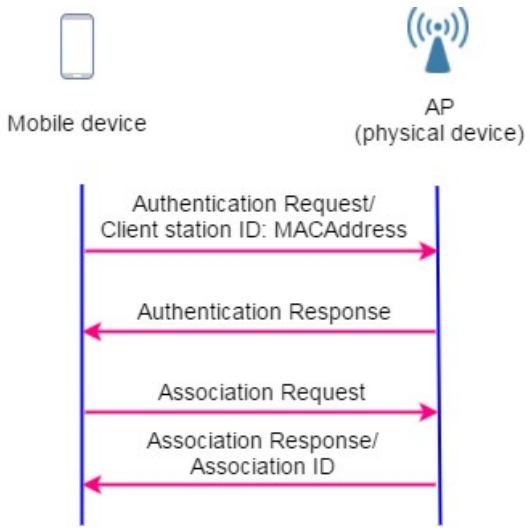

Рис. 1.7. Відкрита автентифікація 802.11 і асоціація між клієнтським пристроєм і точкою доступу

Клієнтський пристрій спочатку надішле до точки доступу запит на автентифікацію зі своєю MAC-адресою як ідентифікатором пристрою. Далі точка доступу надішле відповідь про автентифікацію, вказуючи на те, що автентифікація пройшла успішно чи невдало. Потім клієнтський пристрій надішле запит на асоціацію до точки доступу, і у відповідь точка доступу відповість ідентифікатором асоціації в разі успіху.

1.4.2. Локальна та віддалена автентифікація

Аутентифікацію між мобільним пристроєм і сервером AAA також можна розділити на дві інші групи: локальна автентифікація та віддалена автентифікація. Локальна автентифікація означає, що клієнтський пристрій автентифікується на сервері AAA локальної мережі точок доступу. Віддалена автентифікація означає, що клієнтський пристрій автентифікується на сервері AAA мережі SP. Під час етапу виявлення, відповідно до ідентифікаційної інформації, отриманої від мобільного пристрою, точка доступу може визначити, що пристрій має продовжити локальну автентифікацію або

віддалену автентифікацію. Якщо ідентифікатор мобільного пристрою є в списку локальних користувачів мережі гарячої точки, він повинен виконати локальну автентифікацію за допомогою сервера AAA мережі гарячої точки. Якщо ідентифікатор мобільного пристрою відсутній у цьому списку, він повинен виконати віддалену автентифікацію за допомогою сервера AAA мережі SP, і на цьому етапі сервер AAA мережі точки доступу діє як наскрізний, пересилаючи повідомлення між пристроєм і AAA SP.

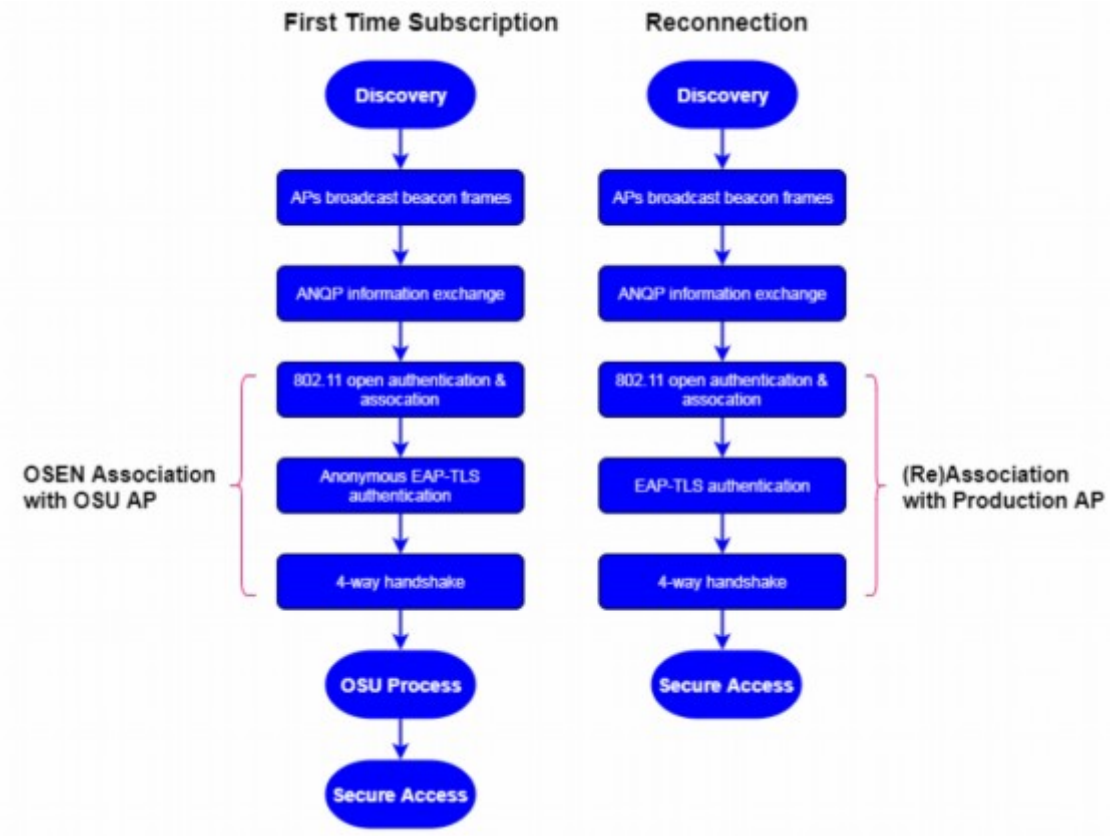

Рис. 1.8. Робочі процеси локальної автентифікації під час першої підписки та повторного підключення

На рисунку 1.8 показано робочі процеси локальної автентифікації під час першої підписки (лівий процес) і повторного підключення (правий процес). Обидва процеси локальної автентифікації використовують безпечні методи автентифікації. Порівнюючи процеси безпечної автентифікації, існує три відмінності між цими двома з’єднаннями:

- Безпечний метод автентифікації
- Об’єкт для з’єднання
- Намір з'єднання

По-перше, для безпечного методу автентифікації лівий процес використовує анонімну автентифікацію EAP-TLS, тоді як правий процес використовує взаємну автентифікацію EAP-TLS. Як пояснювалося вище, використання автентифікації EAP-TLS залежить від того, чи надано клієнтському пристрою облікові дані від SP.

По-друге, для об’єкта з’єднання в лівому процесі клієнтський пристрій зв’язується з точкою доступу OSU, тоді як у правому процесі клієнтський пристрій зв’язується з робочою точкою доступу. У стандарті HS2.2 одна точка доступу з підтримкою HS2.2 повинна мати дві різні віртуальні точки доступу – точку доступу OSU і робочу точку доступу. AP OSU використовується лише для підключення до сервера OSU, тоді як робоча AP використовується для надання мережевого доступу до автентифікованого мобільного пристрою. Вибір об’єкта з’єднання пов’язаний із наміром з’єднання. Останнє - це намір підключення. У лівому процесі клієнтський пристрій перейде до процесу OSU після завершення локальної автентифікації. Оскільки клієнтський пристрій має підписатися на сервер OSU у мережі SP, щоб отримати облікові дані підписки. Тому клієнтський пристрій спочатку має підключитися до точки доступу OSU. У правильному процесі, щоб отримати доступ до мережевої служби, клієнтський пристрій підключиться до робочої точки доступу та безпосередньо перейде до етапу безпечного доступу після завершення локальної автентифікації.

Таблиця 1.1.

Методи автентифікації, які використовуються для першої підписки та повторного підключення

Методи автентифікації	Перша підписка	Повторне підключення
Локальна аутентифікація	Відкрита/Анонімна	Відкрите/взаємне
Віддалена аутентифікація	Взаємна	Взаємне

У таблиці 1.1 наведено методи автентифікації, які використовуються для першої підписки та повторного підключення. У мережах HS2.2 точка доступу має два варіанти локальної автентифікації: відкрита автентифікація та безпечна автентифікація. Якщо використовується безпечна автентифікація, для першої підписки між клієнтським пристроєм і сервером AAA мережі гарячої точки можна виконувати лише анонімну автентифікацію, як описано вище. Що стосується повторного підключення, між ними повинна бути виконана взаємна аутентифікація. Для віддаленої автентифікації клієнтський пристрій має взаємно автентифікуватися з сервером AAA мережі SP незалежно від першої підписки чи повторного підключення.

Висновки до розділу

В даному розділі проведено всебічний аналіз стандарту HS2.2. Зокрема, детально розглянуто теоретичні засади, що передували його розробці, а також практичні аспекти впровадження та функціонування мереж, побудованих на основі цього стандарту. Описано поетапний процес розгортання мережі HS2.2 та детально проаналізовано кожен з етапів, включаючи налаштування обладнання, конфігурацію мережі та встановлення безпекових політик. Особлива увага приділена опису робочого процесу мереж HS2.2, зокрема механізмам автентифікації, авторизації та обліку користувачів.

РОЗДІЛ 2. ДОСЛІДЖЕННЯ МЕТОДІВ СПІЛЬНОГО ВИКОРИСТАННЯ ОБЛІКОВИХ ДАНИХ НА ОСНОВІ СПЕЦИФІКАЦІЙ

2.1. Принципи надання сертифікатів в мережах HS2.2

Для аналізу можливостей спільного використання облікових даних на основі сертифікатів в мережах HS2.2 необхідно детально розглянути структуру та функціональне призначення сертифікатів. Згідно зі стандартом HS2.2, клієнтські сертифікати, як правило, представлені у форматі X.509v3 і базуються на криптографічних парах ключів RSA. Сертифікат X.509v3 є цифровим документом, який зв'язує відкритий ключ з відповідним суб'єктом та містить цифрову підпис для забезпечення автентичності.

Структура сертифіката X.509v3 передбачає наявність таких обов'язкових полів (таблиця 2.1):

- tbsCertificate: містить основну інформацію про сертифікат, включаючи ідентифікатори суб'єкта та емітента, відкритий ключ, період дії та інші атрибути.

- signatureAlgorithm: визначає алгоритм, який використовується для створення цифрового підпису сертифіката.

- signatureValue: власне цифровий підпис, який обчислюється на основі вмісту поля tbsCertificate за допомогою приватного ключа емітента сертифіката.

Таблиця 2.1.

Основні поля сертифіката відкритого ключа

Basic Certificate fields	Content	
1.tbsCertificate	subject	
	issuer	
	validity	
	subjectPublicKeyInfo	
	...	
	extensions	subjectAltName
		...
2.signatureAlgorithm	algorithm	
3.signatureValue	digital signature	

У HS2.2 пара відкритий/приватний ключ генерується мобільним пристроєм під час процесу OSU (рис. 2.1, крок 4) і зберігається в пристрої. З одного боку, відкритий ключ можна використовувати для шифрування повідомлень, а відповідний закритий ключ потім використовувати для розшифровки таких зашифрованих повідомлень. З іншого боку, приватний ключ можна використовувати для створення цифрового підпису, який потім можна перевірити відповідним відкритим ключем. Оскільки приватний ключ прив'язаний до певного суб'єкта та не може вільно передаватись, дійсний цифровий підпис можна використовувати для перевірки автентичності цього суб'єкта під час процесу автентифікації EAP-TLS.

2.1.1. Реєстрація сертифіката HS2.2

На етапі забезпечення, після того як буде визначено, що сертифікат має бути наданий як облікові дані, сервер OSU інструктує мобільний пристрій розпочати процес реєстрації сертифіката клієнта, надсилаючи команду виконання getCertificate, як показано на рисунку 2.1.

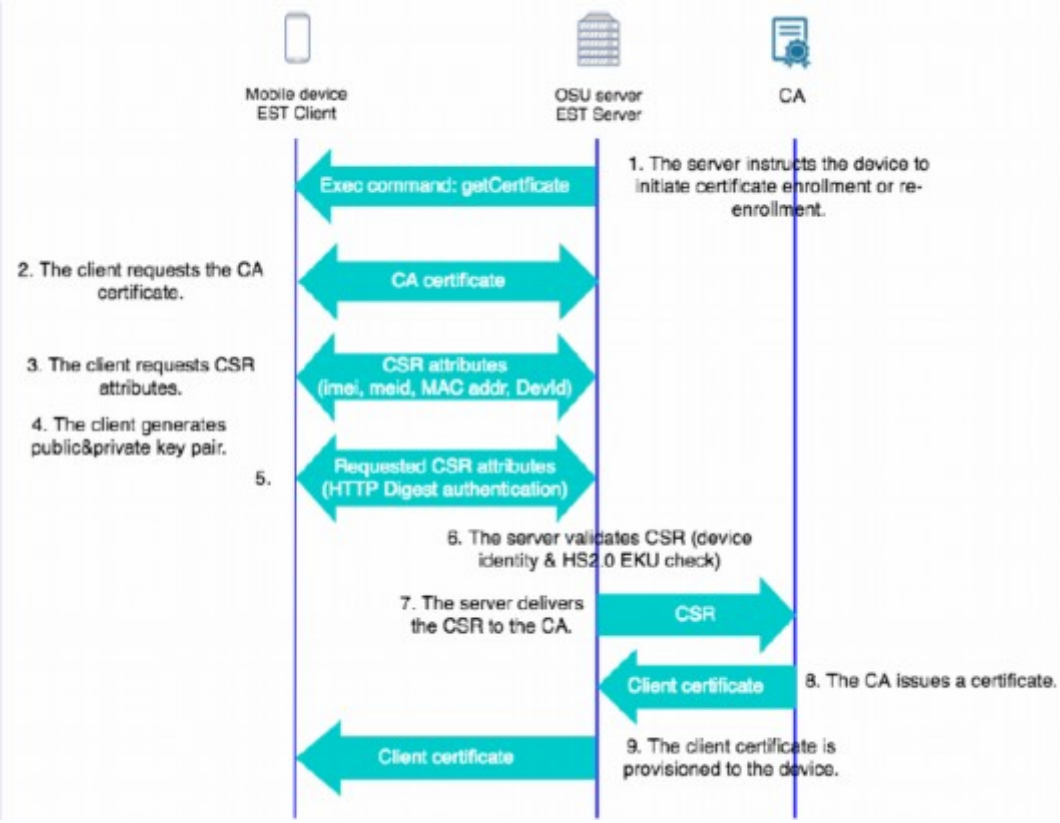

Рис. 2.1. Потік повідомлень про процес реєстрації сертифіката клієнта

Реєстрація сертифіката для клієнтів використовує протокол Enrollment over Secure Transport (EST). Мобільний пристрій діє як клієнт EST, а сервер OSU діє як сервер EST. Щоб запросити сертифікат клієнта від CA, мобільному пристрою необхідно спочатку надати свій ідентифікатор пристрою, наприклад DevId, адресу Wi-Fi MAC, IMSI, IMEI MEID, та свій відкритий ключ серверу OSU. Ця інформація про ідентифікацію пристрою міститься як у DevInfo MO, так і в DevDetail MO в дереві керування пристроєм OMA.

Потім сервер OSU перевіряє ідентифікацію пристрою для CA, оскільки він вже отримав DevInfo та DevDetail MO під час початкового асоціювання. Після того, як ідентифікація пристрою буде перевірена, сервер OSU передасть запит на сертифікат клієнта з інформацією про ідентифікацію та відкритий ключ CA. Як останній крок CA видає сертифікат клієнта та надсилає його на мобільний пристрій через сервер OSU. Сформований відповідно до X.509V3, сертифікат клієнта містить інформацію про відкритий ключ, ідентифікацію пристрою та цифровий підпис CA.

2.1.2. Використання спільного доступу до облікових даних

Незважаючи на ризики, пов’язані зі спільним використанням облікових даних, для деяких постачальників послуг, щоб бути більш привабливими та підвищити лояльність клієнтів, спільний доступ до імені користувача/паролю вважається прийнятним використанням, як-от Netflix і Spotify. Як згадувалося в першому розділі, SP може надати облікові дані імені користувача/паролю для пристрою замість сертифіката.

Стандарт HS2.2 містить опцію спільного використання облікових даних імені користувача та пароля. Для цього в стандарті визначено параметр AbleToShare; цей параметр входить до MO ППС. Після того, як SP встановлює для цього параметра значення true, весь PPS MO, який також включає ім’я користувача/пароль, має бути спільно використаний кількома мобільними пристроями в загальній службі підписки, щоб увімкнути

спільний доступ до облікових даних. Оскільки цей тип спільного використання облікових даних є законним, ці мобільні пристрої можуть отримати весь PPS MO безпосередньо з сервера OSU.

У наведеному нижче тексті передбачається, що SP надає сертифікат для пристроїв. На відміну від спільного використання облікових даних імені користувача/пароля, стандарт HS2.2 не визначає опцію для SP, щоб дозволити спільний доступ до облікових даних, якщо використовуються сертифікати; зокрема, стандарт не визначає жодних параметрів, конфігурації підписки чи політики для цього. Крім того, стандарт визначає, що пристрої генерують власні пари ключів і що значення закритого ключа в такій парі невідоме SP. Іншими словами, SP не може дозволити кільком пристроям використовувати однакові облікові дані.

Оскільки сертифікати є загальнодоступними, сертифікат одного пристрою може використовуватися іншими пристроями під час процесу автентифікації. Однак, не знаючи значення відповідного закритого ключа, інший пристрій не зможе успішно пройти процес автентифікації. Крім того, використання сертифікатів і закритих ключів певною мірою вважається більш безпечним, ніж використання імені користувача/пароля. Якщо облікові дані сертифіката можна спільно використовувати та повторно використовувати на кількох пристроях, тоді ризик для безпеки та конфіденційності значно підвищиться. Крім того, для пристроїв без SIM-карти секрети та інша важлива інформація, як-от закриті ключі, не будуть захищені апаратним забезпеченням, що також створює ризики як для авторизованих користувачів, так і для постачальників відповідних послуг.

Таким чином, необхідно вивчити спільний доступ до облікових даних сертифіката в мережах HS2.2. У наступному розділі на основі двох варіантів використання буде з’ясовано всю необхідну інформацію для спільного використання, а також буде проаналізовано технічну можливість спільного використання облікових даних сертифіката в мережах HS2.2.

2.2. Методи спільного використання облікових даних сертифікату

2.2.1. Випадки використання

Як приклад, припустимо, що є два користувачі, користувач А та користувач В. І користувач А, і користувач В мають два мобільні пристрої, як показано на рисунку 2.2.

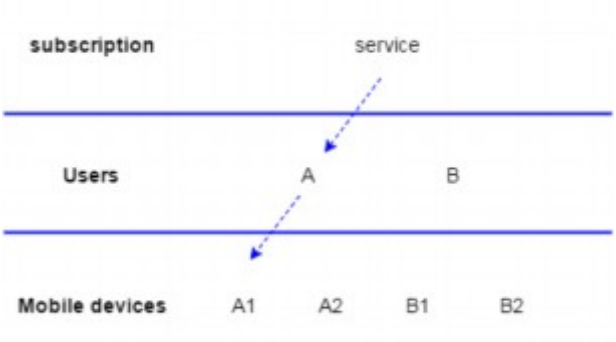

Рис. 2.2. Приклад спільного використання облікових даних сертифіката

Усі ці чотири мобільні пристрої не є SIM-пристроями (пристрій А1, пристрій А2, пристрій В1 і пристрій В2). Користувач А зареєстрував свій мобільний пристрій А1 у пакеті оновлень 1 і успішно отримав доступ до своєї служби передплати за допомогою облікових даних сертифіката, наданого пристрою А1. Пристрій А2, пристрій В1 і пристрій В2 не підписані на одну послугу з SP1. Тепер перший випадок використання — випадковий обмін, що користувач А бажає поділитися обліковими даними сертифіката пристрою А1 з іншими трьома мобільними пристроями, щоб дозволити їм отримати доступ до тієї самої підписки, що й пристрій А1. Як пояснювалося вище, специфікація не підтримує спільний доступ до сертифікатів та/або закритих ключів. Як наслідок, користувач А вважається зловмисником, якщо він/вона ділиться сертифікатом або закритим ключем пристрою А1 з іншим пристроєм (таким як пристрій А2, пристрій В1 та/або пристрій В2).

Другий випадок використання — це викрадення облікових даних, тобто облікові дані сертифіката пристрою А1 викрадаються користувачем В, а

потім використовуються на іншому пристрої. На цьому етапі користувач B вважається зловмисником. Незалежно від випадкового обміну чи викрадення облікових даних, пристрій A1 є законним пристроєм, тоді як пристрій A2, пристрій B1 і пристрій B2 є піратськими пристроями. Після того, як уся необхідна інформація буде витягнута з пристрою A1 і скопійована на піратські пристрої, кожен із мобільних пристроїв A1, A2, B1 і B2 може видати себе за законний пристрій A1 і отримати доступ до послуги, що надається SP1. З точки зору SP1, він не може виявити справжню особу пристрою або користувача, який видає себе за іншу особу, якщо він/вона незаконно використовує дубльовані облікові дані сертифіката, щоб отримати доступ до служби підписки на законні пристрої.

2.2.2. Аналіз процесів обміну інформацією

Для обох випадків використання інформація, яку потрібно отримати з пристрою A1, ідентична. Незалежно від того, чи надано облікові дані добровільно, чи їх несвідомо вкрадено, мета спільного доступу завжди полягає в тому, щоб дозволити іншим мобільним пристроям імітувати пристрій A1, тобто успішно пройти через процес повторного підключення (етапи виявлення та безпечного доступу) з використанням посвідчення пристрою A1. Таким чином, процес повторного підключення є ключовим моментом для визначення інформації, необхідної для спільного використання облікових даних. Перший варіант використання буде використано для наступного аналізу.

• Відкриття

На початку процесу повторного підключення пристрій A1 сканує мережі, що підтримують HS2.2, і використовує повідомлення ANQP для вибору мережі. На основі наданої конфігурації підписки пристрій A1 порівнює інформацію ANQP, отриману від мережі гарячої точки, з політикою Home SP і параметрами, специфічними для підписки, що зберігаються в PPS.

Спільний доступ до PPS MO необхідний, щоб увімкнути спільний доступ до облікових даних сертифіката, оскільки конфігураційна інформація в PPS MO необхідна для того, щоб інші мобільні пристрої могли підключатися до того самого SP і використовувати ту саму послугу підписки, що й користувач A.

Після виявлення та вибору мережі пристрій A1 підключається до робочої точки доступу та виконує локальну автентифікацію за допомогою сервера AAA мережі HS. Як зазначено в таблиці 1.1, існує два варіанти локальної автентифікації: один — відкрита автентифікація, інший — безпечна або взаємна автентифікація. У наведеному нижче тексті передбачається, що точка доступу використовує відкриту автентифікацію для повторного підключення (рис. 1.6).

На початку відкритої автентифікації та асоціації пристрій A1 спочатку надішле запит на автентифікацію зі своєю MAC-адресою як ідентифікатором пристрою до точки доступу. Далі точка доступу надішле відповідь про автентифікацію, вказуючи на те, що автентифікація пройшла успішно чи невдало. Потім пристрій A1 надішле запит на асоціацію до точки доступу, і у відповідь точка доступу відповість ідентифікатором асоціації в разі успіху. Після завершення відкритої автентифікації та асоціації пристрій A1 зв’яжеться з робочою точкою доступу та автоматично перейде до етапу безпечного доступу. Під час цього відкритого процесу автентифікації пристрій A1 має лише надати точці доступу свою MAC-адресу. Таким чином, у разі випадкового обміну, користувач A також повинен поділитися MAC-адресою пристрою A1 з іншими пристроями, щоб допомогти їм пройти етап виявлення. MAC-адреса мобільного пристрою є параметром, який міститься в DevDetail MO. Зауважимо, що просто мати спільну MAC-адресу недостатньо, піратські пристрої (пристрій A2, пристрій B1 і пристрій B2) повинні маскувати свої MAC-адреси під MAC-адресу законного пристрою A1. Загалом мобільні пристрої можуть змінити свою MAC-адресу двома

способами: зміна адреси в пам'яті мережевої карти або за допомогою програмного забезпечення.

• Безпечний доступ

Пропускаючи весь процес OSU, пристрій A1 переходить до взаємної автентифікації з сервером AAA мережі SP1. Коли пристрій A1 успішно виконає взаємну автентифікацію, він матиме доступ до мережі. На рисунку 1.6 пояснюється робочий процес взаємної автентифікації EAP-TLS.

На етапі захищеного доступу мобільний пристрій виконує віддалену автентифікацію за допомогою сервера AAA SP, тоді як точка доступу обмінюється повідомленнями між пристроєм і сервером AAA SP. На початку автентифікації пристрій A1 має надати свої DevInfo MO та DevDetail MO як ідентифікаційну інформацію пристрою серверу AAA SP1 (рисунок 1.6, крок 2). Тому і DevInfo MO і DevDetail MO пристрою A1 є інформацією, якою користувач A повинен поділитися.

Після отримання посвідчення сервер AAA та пристрій A1 виконають автентифікацію на стороні сервера та автентифікацію на стороні клієнта відповідно (рис. 1.6, крок 5, 6). Пристрій A1 спочатку автентифікує сервер AAA за допомогою сертифіката сервера AAA та повідомлення перевірки, підписаного закритим ключем сервера AAA, а потім сервер AAA автентифікує пристрій A1 за допомогою сертифіката клієнта та повідомлення перевірки, підписаного закритим ключем пристрою. A1.

Сертифікати та підписи на стороні сервера не потребують перевірки, щоб встановити незаконне з’єднання. Таким чином, у разі випадкового обміну, як сертифікат клієнта, так і приватний ключ слід отримати з пристрою A1. Зауважте, що наприкінці автентифікації EAP-TLS не потрібно ділитися додатковою інформацією для узгодження набору шифрів (кроки 7-9 на рис. 1.6).

Усю необхідну інформацію, необхідну для іншого мобільного пристрою, щоб зв’язатися з мережею SP1 і завершити взаємну

автентифікацію, уже визначено. Підсумовуючи, щоб імітувати пристрій A1, є п’ять елементів, якими потрібно надати спільний доступ, а саме:

1. PPS MO;
2. сертифікат клієнта;
3. закритий ключ;
4. DevInfo MO;
5. DevDetail MO.

як показано на рисунку 2.3. Перші два надаються SP, тоді як інші генеруються або належать самому мобільному пристрою.

Рис. 2.3. Представлення інформації для спільного використання облікових даних сертифіката

2.2.3. Технічна можливість спільного використання облікових даних сертифіката

Для підтвердження можливості спільного використання облікових даних сертифіката недостатньо лише знати інформацію, якою потрібно поділитися. Отримання цієї інформації з пристрою A1 також має бути технічно можливим. У пристроях без SIM-карти інформація не буде захищена апаратними засобами. У результаті ця інформація зберігається в програмному забезпеченні, яке зазвичай пропонує нижчий рівень безпеки. Отже, зловмисники зазвичай можуть отримати інформацію, наприклад, через інтерфейси або за допомогою шкідливих програм. Отже, можна зробити висновок, що можливість спільного використання облікових даних сертифіката дійсно існує в мережах HS2.2.

Висновки до розділу

Розділ присвячено комплексному дослідженню проблематики спільного використання облікових даних на основі сертифікатів у контексті стандарту HS2.2. В перших двох підрозділах представлено детальний опис структури сертифікатів клієнтів та процедури їх реєстрації в мережах HS2.2. Далі обґрунтовано актуальність дослідження можливостей спільного використання облікових даних сертифікатів та проаналізовано мотивацію такого підходу. Також проведено аналіз інформаційних потреб для реалізації спільного використання та оцінено технічну можливість отримання необхідних даних з авторизованих пристроїв.

РОЗДІЛ 3. ІМПЛЕМЕНТАЦІЯ МОДЕЛЕЙ ТА МЕТОДІВ ДЛЯ ВИРІШЕННЯ ПРОБЛЕМИ ВИКОРИСТАННЯ ОБЛІКОВИХ ДАНИХ

3.1. Представлення методу автентифікації на основі токенів

В цьому розділі будуть запропоновані та обговорені два підходи на основі токенів. Спочатку буде представлено концепцію токена та токен-базованої автентифікації. Потім показано, як загальний метод на основі токенів використовувався раніше та яку проблему вирішує цей метод. Після цього розглянемо конкретний випадок HS2.2, щоб пояснити застосування методу на основі токенів у середовищі HS2.2 та запропонувати два підходи.

Токени можна класифікувати на дві групи: токени на основі апаратного забезпечення та токени на основі програмного забезпечення. Токени на основі апаратного забезпечення відносяться до фізичних пристроїв, таких як брелоки, смарт-карти, токени USB тощо. Токени на основі програмного забезпечення відносяться до фрагмента даних, створеного сервером і містить інформацію для ідентифікації конкретного користувача. Токени, які будуть обговорюватися нижче, є токенами на основі програмного забезпечення. Токени на основі апаратного забезпечення не будуть розглядатися. Як і сертифікати, токени зазвичай створюються сервером з часом закінчення терміну дії, після чого вони стануть недійсними, і потрібно буде отримати нові токени від цього сервера. Крім того, токени містять достатньо даних для ідентифікації конкретного користувача.

Токен-базована автентифікація - це механізм, який використовує токени, надані сервером, для швидкої та безпечної ідентифікації користувача або пристрою клієнта, що широко застосовується у веб-додатках. Зазвичай система токен-базованої автентифікації дозволяє користувачам входити в систему зі своїм ім'ям користувача/паролем, щоб отримати токен(и) для майбутньої автентифікації з сервером автентифікації системи. Після отримання токенів користувачам потрібно лише передавати токен для

доступу до конкретного ресурсу або послуги. Іншими словами, токен(и) замінюють ім'я користувача/пароль і стають новими обліковими даними.

3.1.1. Специфікація підходу автентифікації на основі токенів

На рисунку 3.1 показано базовий обмін повідомленнями між клієнтським пристроєм і веб-сервером, коли використовується аутентифікація на основі токенів.

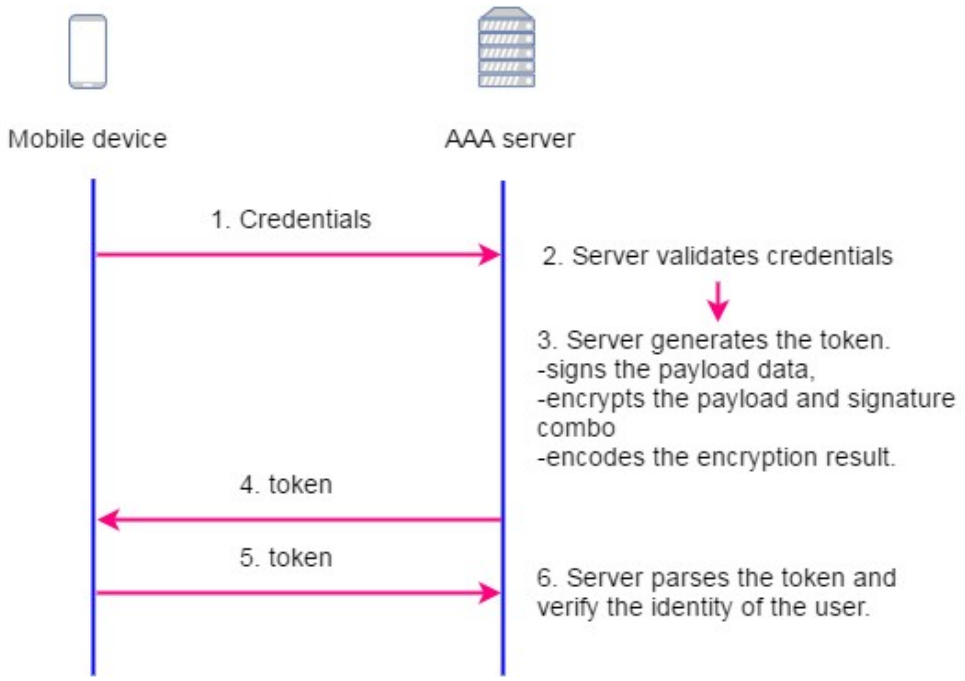

Рис. 3.1. Підхід автентифікації на основі токенів, що використовується у веб-додатках

Розглянемо основні кроки даного підходу:

Крок 1. Клієнтський пристрій спочатку надсилає свої облікові дані для автентифікації на сервер AAA.

Крок 2. Сервер AAA перевіряє та підтверджує облікові дані.

Крок 3. Якщо облікові дані дійсні, сервер AAA згенерує токен. Корисне навантаження токена містить ідентифікаційну інформацію користувача та метадані токена, наприклад час закінчення терміну дії. Після серії

шифрованих операцій сервер AAA генерує криптографічно підписане повідомлення як токен.

Крок 4. Сервер AAA надсилає токен на клієнтський пристрій.

Крок 5. Клієнтський пристрій зберігає токен і надсилає його на сервер AAA в наступних запитах. Токен замінює облікові дані для автентифікації користувача.

Крок 6. Сервер AAA аналізує токен, перевіряє дійсність токена та перевіряє особу користувача шляхом порівняння бази даних відомих користувачів.

У традиційній веб-автентифікації програма покладається на інформацію користувача, що ввійшов у систему, яка зберігається на сервері, для перевірки та автентифікації користувача. Сервер повинен створювати та зберігати записи автентифікації для ідентифікації користувача, що призводить до навантаження на серверне сховище. Аутентифікація на основі токенів призначена для вирішення цієї проблеми. Серверу не потрібно зберігати записи автентифікації, оскільки сам токен містить інформацію користувача з метою перевірки, що може знизити навантаження на сервер.

Використання токенів розширює надійність автентифікації. Інформація, пов’язана з користувачем і токеном, закодована в самому токені та захищає від підробки за допомогою надійного криптографічного підпису. Крім того, термін дії токена закінчується та стає недійсним через встановлений проміжок часу.

Щоб отримати нові токени, користувачеві потрібно буде ще раз пройти автентифікацію на сервері. Сервер може визначати термін дії токена та частоту оновлення, наприклад, одноразовий використаний токен. Час дії токена може вплинути на спільний доступ між авторизованим і неавторизованим пристроями та змусити їх часто синхронізуватися. Іншими словами, використання токенів може дозволити SP виявити та запобігти проблемі спільного використання облікових даних сертифіката.

3.1.2. Випадок застосування HS2.2

Повертаючись до випадку використання спільного доступу, описаного в другому розділі, користувач А бажає поділитися обліковими даними сертифіката пристрою А1 зі своїм іншим пристроєм і пристроями користувача В (пристроєм А2, пристроєм В1 і пристроєм В2). Через унікальність пари відкритий/приватний ключ, незаконний спільний доступ до облікових даних сертифіката має здійснюватися між кількома пристроями незалежно від користувача. Таким чином, варіант використання змінено на те, що користувач А бажає поділитися обліковими даними сертифіката свого пристрою А з іншим пристроєм В. На рисунку 3.2 показано оригінальний результат обміну обліковими даними сертифіката між авторизованим пристроєм А та неавторизованим пристроєм В.

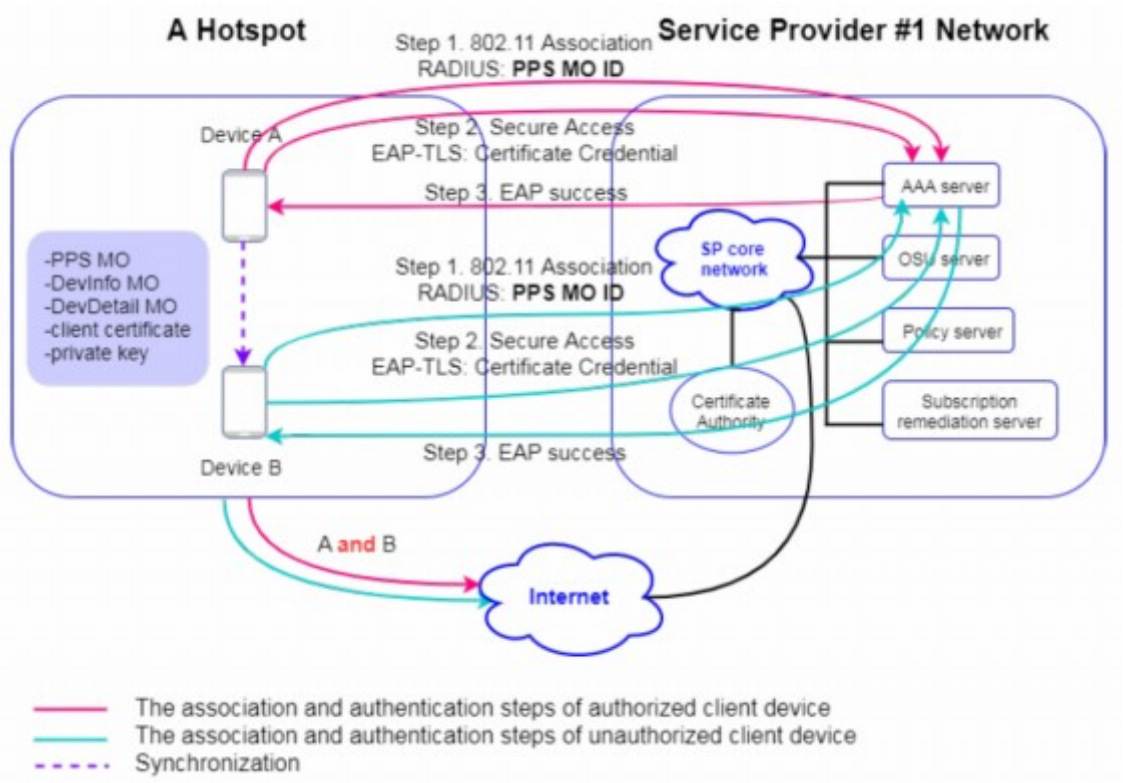

Рис. 3.2. Спільне використання облікових даних оригінального сертифіката між авторизованими та неавторизованими пристроями

Червоні лінії вказують на спрощену асоціацію та кроки автентифікації авторизованого пристрою А, як і на рисунку 1.5. Перш за все, пристрій А

локально автентифікуватиметься та зв’яжеться з точкою доступу мережі гарячої точки. Далі через мережу точки доступу пристрій А зв’яжеться з мережею SP і взаємно автентифікуватиметься на сервері AAA за допомогою облікових даних сертифіката. Після цього сервер AAA надішле на пристрій повідомлення про успішне завершення EAP, яке вказує на успішну взаємну автентифікацію, після чого пристрій А отримає доступ до мережі. Після цього, як проаналізовано в другому розділі, користувач А ділиться всією необхідною інформацією (PPS MO, сертифікат клієнта, закритий ключ, DevInfo MO та DevDetail MO) із пристроєм В. За допомогою цієї інформації пристрій В може слідувати тій самій асоціації. і кроки автентифікації, як показано синіми лініями, і мають доступ до мережі як пристрій А. Зауважимо, що перші два кроки, показані на схемі, пов’язані з наступними двома підходами на основі токенів відповідно.

3.2. Підхід інтенсифікації процедур відновлення та оновлення підписок

Час від часу підписка користувача може вимагати виправлення або оновлення. Виправлення підписки — це процес, який SP використовує на етапі безпечного доступу для зміни інформації, включеної в PPS MO. SP визначає, коли необхідне виправлення підписки та яку інформацію потрібно змінити. Як зазначено в першому розділі, існує два типи виправлення підписки: активне оновлення та пасивне виправлення. Для активного оновлення стандарт визначає параметр UpdateInterval, включений у PPS MO. На основі цього параметра під час повторного підключення на етапі безпечного доступу мобільний пристрій буде активно підключатися до сервера відновлення підписки, а потім з певною частотою оновлюватиме конфігурацію підписки на цьому сервері. Якщо SP запитує пасивне виправлення, сервер AAA SP сповістить мобільний пристрій про необхідність виправлення, а потім пристрій підключиться до сервера

виправлення підписки. Щоб прийняти рішення щодо пасивного виправлення, SP повинен знати поточну версію PPS MO, яка зберігається на мобільному пристрої. Для цього стандарт також визначає параметр PPS MO ID. Значення PPS MO ID змінюватиметься щоразу, коли змінюється будь-яка інформація в PPS MO. Тобто новий ідентифікатор MO PPS замінюватиме старий кожного разу, коли активне оновлення або пасивне виправлення відбувається на етапі безпечного доступу.

З метою виявлення SP повинен зберігати кілька ідентифікаторів MO PPS у базі даних конкретного користувача. Якщо інший пристрій використовує старе значення PPS MO ID для асоціації 802.11, тоді, порівнюючи базу даних користувача, SP може виявити, що пристрій мав оновити свій PPS MO ID, і викликати підозру щодо спільного використання облікових даних сертифіката. У результаті виправлення підписки піратський пристрій змушений синхронізувати ідентифікатор PPS MO ID, щоб продовжувати користуватися послугою. Повертаючись до випадків використання, розглянутих у другому розділі, незалежно від випадкового обміну чи викрадення облікових даних, після того, як усю необхідну інформацію буде отримано з пристрою A, зловмисник може передбачити активну частоту оновлення за допомогою параметра UpdateInterval, щоб уникнути порушення синхронізації. Однак пасивне відновлення, ініційоване SP, не є передбачуваним. Тому перший запропонований підхід полягає у використанні PPS MO ID як токена та збільшенні частоти пасивного виправлення до на з’єднання.

На рисунках 3.3 і 3.4 представлені дві діаграми послідовності, які ілюструють етап безпечного доступу з використанням першого підходу та без нього. На рисунку 3.4 повідомлення, виділені жирним шрифтом, вказують на додатковий робочий процес, якщо SP вимагає виправлення підписки. На обох діаграмах під час процесу асоціації 802.11 ідентифікатор MO PPS буде надано точці доступу, а потім передано на сервер AAA SP у повідомленні запиту доступу RADIUS. Це також перший крок, показаний на

рисунку 3.2. Тоді під час автентифікації EAP сервер AAA може використовувати цей параметр, щоб визначити, чи потрібне пасивне виправлення чи ні.

- Без використання першого підходу інтенсифікації

Зазвичай SP рідко потребує пасивного виправлення, за винятком випадків збільшення частоти виправлення або виникнення проблем, наприклад, закінчення терміну дії сертифіката або прострочений рахунок. Зверніть увагу, що в стандарті HS2.2 термін дії сертифіката клієнта становить два роки. Таким чином, припустимо, що частота не збільшується і SP не вимагає пасивного виправлення, тоді після успішної взаємної автентифікації з сервером AAA пристрій A отримає доступ до мережі, як показано на рисунку 3.3.

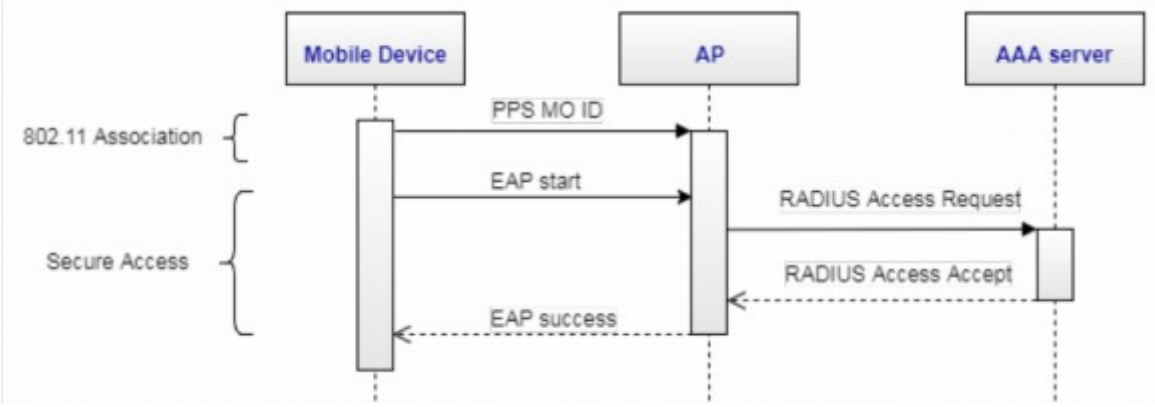

Рис. 3.3. Діаграма послідовності етапу безпечного доступу без використання першого підходу інтенсифікації

- З використанням першого підходу інтенсифікації

Припустимо, що SP вимагає пасивного виправлення, сервер AAA міститиме запит сповіщення WNM (Wireless Network Management) у повідомленні RADIUS Access Accept і надсилатиме його до точки доступу, як показано на рисунку 3.4 . Потім точка доступу передасть це повідомлення на пристрій A після надсилання повідомлення про успішне завершення EAP. PPS MO пристрою A містить URL-адресу сервера виправлення підписки, щоб

пристрій А міг підключитися до передбачуваного сервера виправлення підписки, ідентифікованого за URL-адресою. Пристрій А ініціює підключення до сервера відновлення підписки за допомогою HTTPS. Після цього пристрій А спочатку автентифікує сервер виправлення підписки за допомогою сертифіката сервера виправлення підписки та повідомлення перевірки, підписаного закритим ключем сервера виправлення підписки. Оскільки пристрій А має облікові дані сертифіката, він використовує цей сертифікат і повідомлення перевірки, підписане його закритим ключем, для автентифікації на стороні клієнта. Після цього сервер відновлення підписки оновить ідентифікатор PPS MO ID. Отримавши оновлений ідентифікатор PPS MO ID і замінивши старий у PPS MO, пристрій А роз'єднає робочу точку доступу, а потім повторно зв'яже його за допомогою оновленого PPS MO ID.

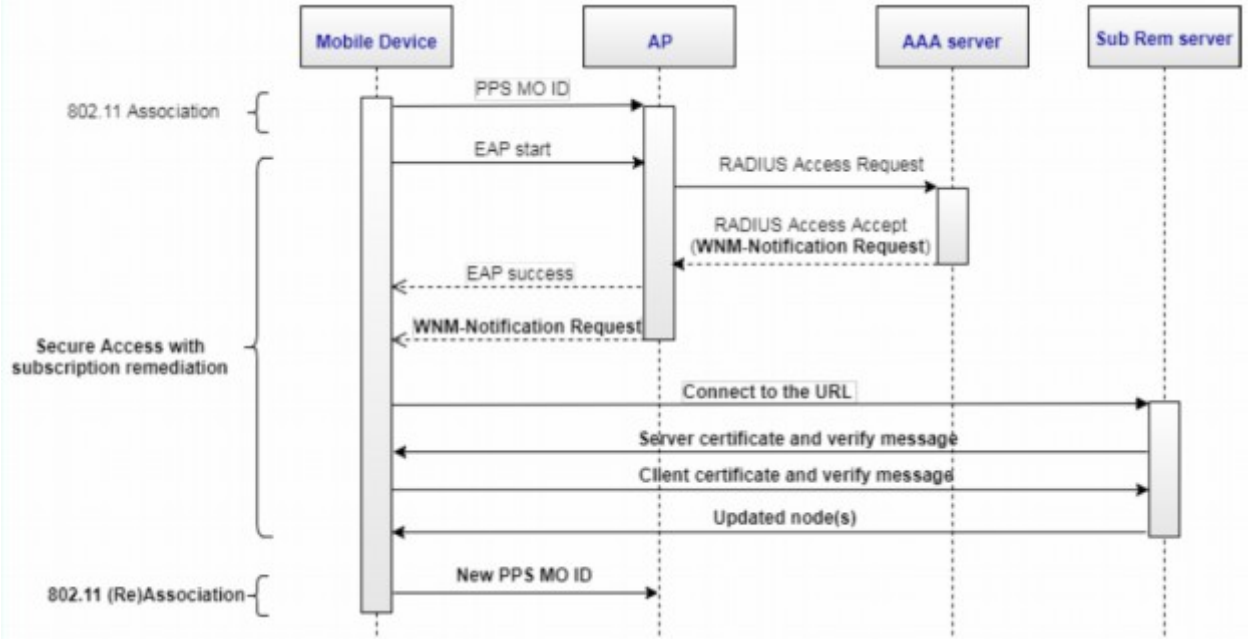

Рис. 3.4. Діаграма послідовності етапу безпечного доступу при використанні першого підходу інтенсифікації

Проведемо аналіз підходу. На рисунку 3.5 показано результат випадкового обміну між пристроями А та В, коли пристрою А часто потрібно проходити процес виправлення підписки. Припустімо, що пристрій В

отримав всю необхідну інформацію з пристрою A, але ще не використав цю інформацію для доступу до мережі.

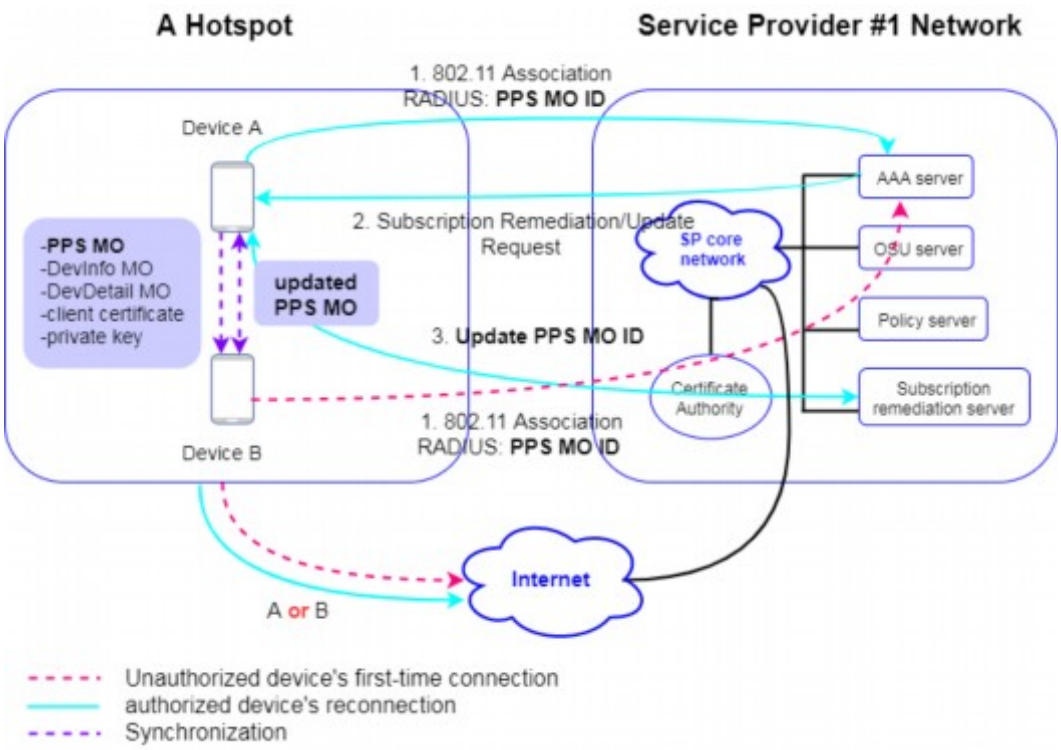

Рис. 3.5. Полегшення спільного використання облікових даних сертифіката на основі процесу виправлення підписки

Є три ситуації:

1) Якщо пристрій A оновлює свій PPS MO ID перед пристроєм B, а пристрій B ще не синхронізував останній PPS MO ID. Пристрій A отримає новий PPS MO ID, тоді як пристрій B має лише стару версію, як показано на рисунку 3.6. SP може виявити наявність спільного використання облікових даних сертифіката, якщо пристрій B все ще використовує старе значення PPS MO ID під час асоціації 802.11. У цьому випадку лише пристрій A може отримати доступ до послуги мережі.

2) Якщо пристрій B оновлює свій PPS MO ID перед пристроєм A, а пристрій A ще не завершив синхронізацію. Пристрій B отримає новий PPS MO ID, тоді як пристрій A має лише стару версію, як показано на рисунку 3.6. SP може виявити наявність спільного використання облікових даних

сертифіката, якщо пристрій А все ще використовує старе значення PPS MO ID під час асоціації 802.11. У цьому випадку лише пристрій В може отримати доступ до послуги мережі.

3) Якщо і пристрій А, і пристрій В містять оновлений ідентифікатор PPS MO ID. У цьому випадку SP не може виявити спільний доступ до облікових даних сертифіката. Як контрзахід у цій ситуації SP може збільшити частоту оновлення до на з'єднання, щоб запобігти одночасному використанню між пристроями А та В. Навіть якщо користувач А бажає надати спільний доступ до нового PPS MO, або пристрій А, або пристрій В можуть отримати доступ до послуга мережі.

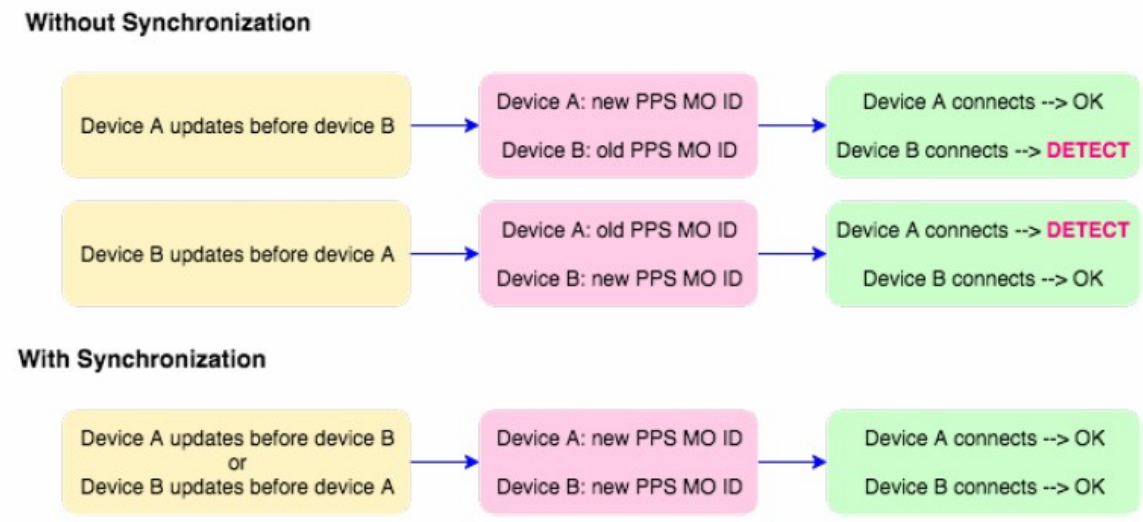

Рис. 3.6. Обставини, коли SP може і не може виявити наявність спільного використання облікових даних сертифіката

З одного боку, використання цього підходу може дозволити SP виявляти наявність спільного використання облікових даних сертифіката, коли клієнтський пристрій (пристрій А або пристрій В) підключається до мережі до завершення синхронізації між пристроєм А і пристроєм В, як показано на рис. 3.6 . З іншого боку, якщо SP вимагає пасивного виправлення кожного разу, коли клієнтський пристрій підключається до своєї мережі, цей підхід можна використовувати для запобігання одночасному використанню

кількох пристроїв. Іншими словами, кілька пристроїв, які імітують один і той самий пристрій, не можуть отримати доступ до однієї служби підписки одночасно. Цей підхід також має свої обмеження: SP не може визначити, який пристрій скасувати асоціацію або де-автентифікацію після виявлення наявності спільного доступу, оскільки цей підхід не може дозволити SP виявити ідентифікатор пристрою.

Припустімо, що SP збільшує частоту виправлення підписки на кожне підключення, тоді використання цього підходу матиме такий вплив на архітектуру HS2.2:

1) Структурна складність: на етапі безпечного доступу, коли сервер AAA перевіряє значення PPS MO ID, йому знадобиться інший сервер токенів або сам сервер AAA, щоб відстежувати ідентифікатори PPS MO для кожного мобільного пристрою. У разі потреби іншого сервера цей компонент необхідно додати до мережі SP.

2) Затримка: час автентифікації буде збільшено. Час автентифікації Ta визначається як:

$$T\,a_1 = ta_1 + t_s \tag{3.1}$$

де t_s — середній час, необхідний для завершення одного процесу виправлення підписки, ta_1 — початковий середній час, необхідний для завершення взаємної автентифікації, Ta_1 — новий середній час автентифікації для одного клієнтського пристрою на з’єднання. Кожного разу, коли клієнтський пристрій підключається до мережевої служби, користувач повинен чекати Ta_1 для автентифікації.

3) Вартість: зі збільшенням частоти виправлення з’являться додаткові витрати на обчислення для SP. Вартість обчислення C визначається як:

$$C = c_{sr} * n_{rq} \tag{3.2}$$

де c_{sr} – обчислювальна вартість для оновлення одного PPS MO ID, n_{rq} – середня кількість запитів на повторне підключення від мобільних пристроїв на день, C вказує загальну обчислювальну вартість процесу виправлення підписки на день. Таким чином, вартість обчислень SP зростатиме лінійно разом із кількістю запитів на повторне підключення.

Насправді цей підхід змінює робочий процес стандарту HS2.2. Перед використанням цього підходу клієнтський пристрій рідко вимагає виконання процесу виправлення підписки. Після використання цього підходу щоразу, коли пристрій підключатиметься, йому потрібно буде зв’язуватися з сервером відновлення підписки на етапі безпечного доступу.

3.3. Підхід створення нових токенів

Посилаючись на приклад веб-додатку, другий запропонований підхід полягає в генерації нових токенів для процесу автентифікації. Рисунок 3.7 ілюструє нещодавно розроблену першу підписку. Порівняно з рисунком 3.2 архітектура та робочий процес були змінені. Цей підхід реалізує службу токенів на основі стандарту HS2.2. Служба токенів використовує API (інтерфейс програмування додатків), який дозволяє SP генерувати та маніпулювати токенами, прив’язаними до певного мобільного пристрою. Через службу токенів сервери мережі SP можуть створювати нові токени, перевіряти статус токенів, недійсні токени тощо. На відміну від прикладу веб-додатку, токени не замінять сертифікат клієнта як облікові дані для автентифікації. На етапі безпечного доступу клієнтський пристрій має пройти взаємну автентифікацію з сервером AAA за допомогою облікових даних сертифіката та токенів. На рисунках 3.8 і 3.9 представлені дві діаграми послідовності, які ілюструють процес OSU і процес взаємної автентифікації з використанням другого підходу та без нього. На рисунку 3.9 повідомлення, виділені жирним шрифтом, вказують на етапи створення та перевірки токенів.

• Без використання другого підходу (створення токенів)

Як було описано в попередньому розділі, під час процесу OSU, коли SP отримує ідентифікатор пристрою та вирішує надати сертифікат як облікові дані, сервер OSU повідомить клієнтський пристрій про початок процесу реєстрації сертифіката клієнта. Потім клієнтський пристрій надсилає свої ідентифікаційні атрибути пристрою та відкритий ключ на сервер OSU. Після підтвердження ідентифікації пристрою атрибути, сервер OSU передає їх і відкритий ключ центру сертифікації для генерації сертифіката клієнта. Потім видається сертифікат клієнта та надсилає його на сервер OSU. На останньому етапі початкового процесу OSU клієнтський пристрій отримує свій сертифікат і PPS MO від сервера OSU. На початковому етапі захищеного доступу клієнтський пристрій повинен лише надати свій сертифікат і повідомлення перевірки, підписане закритим ключем, на сервер AAA для взаємної автентифікації. Наприкінці взаємної автентифікації сервер AAA надішле клієнтському пристрою повідомлення про успішне завершення EAP, щоб вказати на успішну автентифікацію.

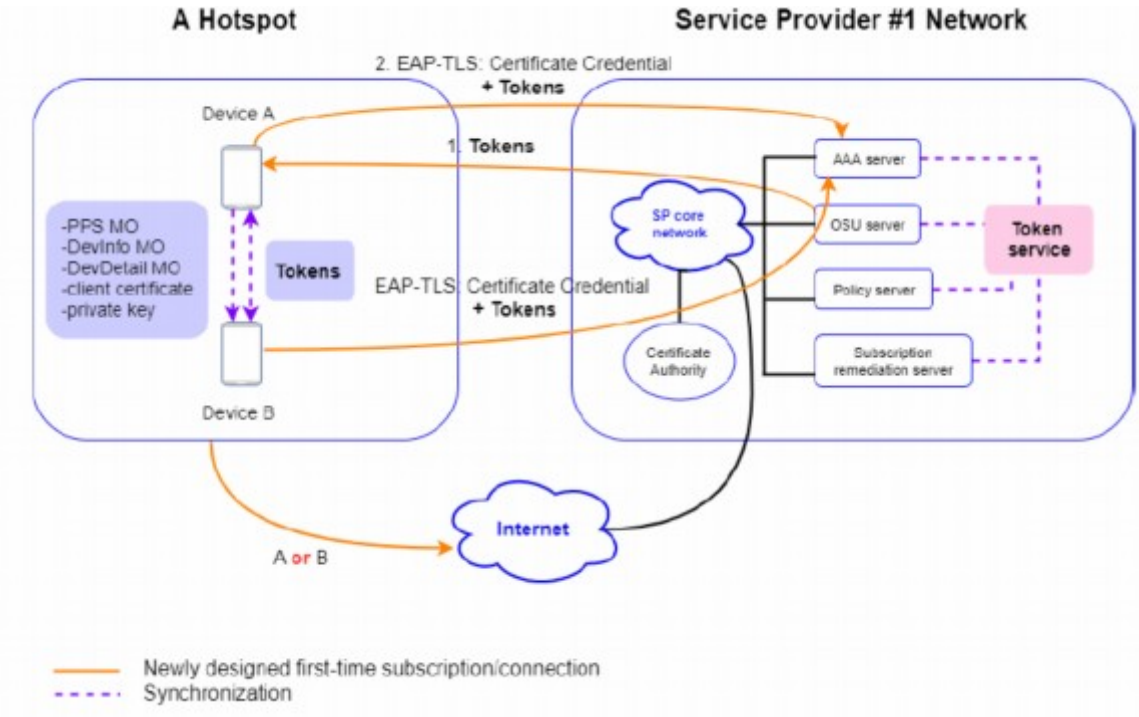

Рис. 3.7. Полегшення спільного використання облікових даних сертифіката на основі щойно згенерованих токенів

- З використанням другого підходу

У новій розробленій першій підписці отримання токена від сервера OSU буде останнім кроком процесу OSU. Як показано на малюнку 4.9 , сервер OSU викличе службу токенів, щоб створити токен 1, а потім надіслати на клієнтський пристрій для подальшої взаємної автентифікації з сервером AAA. Токен 1 — це частина даних, підписана закритим ключем сервера OSU. Корисне навантаження токена 1 містить ідентифікацію пристрою та метадані токена, наприклад час закінчення терміну дії, налаштований SP. На новому етапі безпечного доступу, окрім сертифіката клієнта, клієнтський пристрій також має надати попередньо згенерований токен для перевірки та автентифікації. Тому наприкінці взаємної автентифікації сервер AAA не надсилатиме клієнтському пристрою повідомлення про успішне завершення EAP, оскільки серверу AAA потрібно перевірити токен 1.

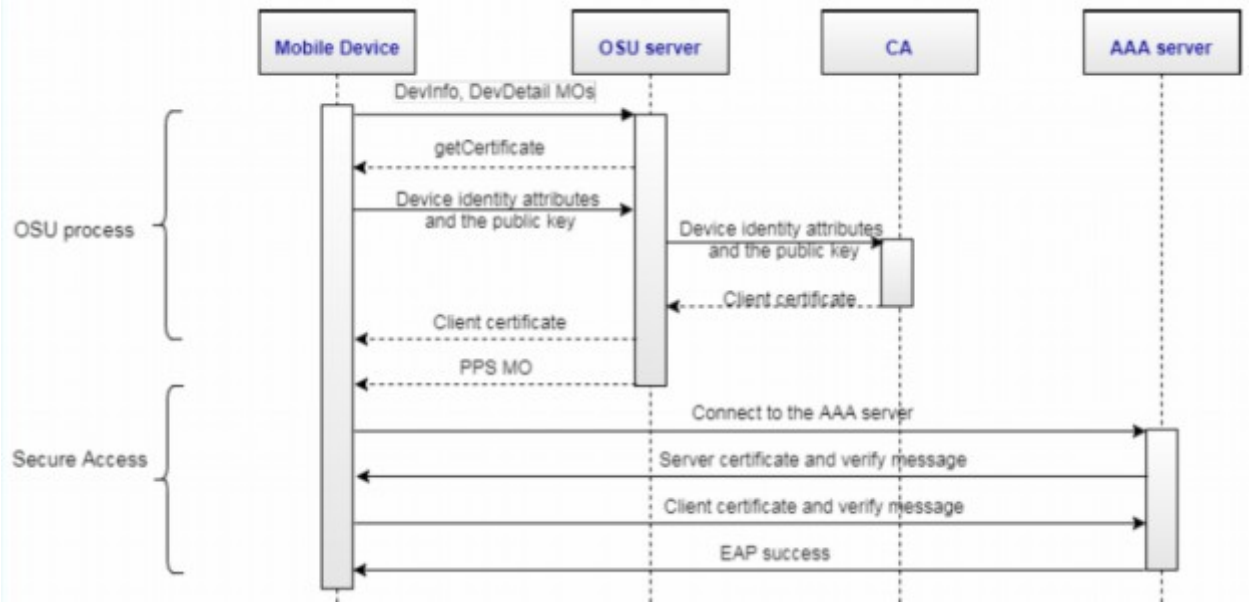

Рис. 3.8. Діаграма послідовності процесу OSU і безпечного доступу без використання другого підходу (створення токенів)

Припустімо, що сервер AAA отримує відкритий ключ, який відповідає закритому ключу, який використовується для підпису токена, від сервера

OSU. Сервер AAA може використовувати цей відкритий ключ для перевірки цифрового підпису токена 1. У цьому випадку сервер AAA може підтвердити, що токен 1 був згенерований сервером OSU того самого SP.

Якщо термін дії токена 1 закінчився, сервер AAA надішле клієнтському пристрою повідомлення про відхилення доступу. Якщо термін дії токена 1 не минув, тоді сервер AAA викличе службу токенів, щоб перевірити, чи можна знайти токен 1 у посиланні бази даних токенів на клієнтський пристрій. Далі, якщо токен 1 не знайдено в базі даних, тоді сервер AAA надішле клієнтському пристрою повідомлення про відхилення доступу. Якщо токен 1 можна знайти в базі даних, тоді сервер AAA перевірить статус цього токена. Далі, якщо статус недійсний, сервер AAA надішле клієнтському пристрою повідомлення про відхилення доступу. Якщо статус дійсний, то сервер AAA надішле клієнтському пристрою повідомлення про успіх. Наприкінці сеансу сервер AAA викличе службу токенів для недійсного токена 1 і згенерує інший токен 2. Токен 2 буде надіслано на клієнтський пристрій для наступної автентифікації.

Повертаючись до рисунка 3.7, він також ілюструє результат випадкового обміну між пристроями A та пристроями B, якщо пристрій A використовує токен як ще один обліковий запис для отримання доступу до мережевої служби. Припустімо, що пристрій A уже завершив нещодавно розроблену першу підписку та отримує облікові дані сертифіката та токен 2, оскільки токен 1 використовувався під час взаємної автентифікації. Крім того, пристрій B отримав всю необхідну інформацію з пристрою A. Є дві ситуації:

1) Якщо пристрій A спочатку повторно підключається до мережі за допомогою облікових даних сертифіката та токена 2. На цьому етапі токен 2 стає недійсним, оскільки його можна використати лише один раз. Якщо пристрій B все ще використовує токен 2 для автентифікації, тоді SP може виявити, що облікові дані були надані спільно. Таким чином, лише пристрій A може отримати доступ до послуги мережі.

2) Якщо пристрій В спочатку повторно підключається до мережі за допомогою облікових даних сертифіката та токена 2. Пристрій А не може отримати доступ до мережевої служби, якщо пристрій В не надає спільний доступ до нового згенерованого токена 3.

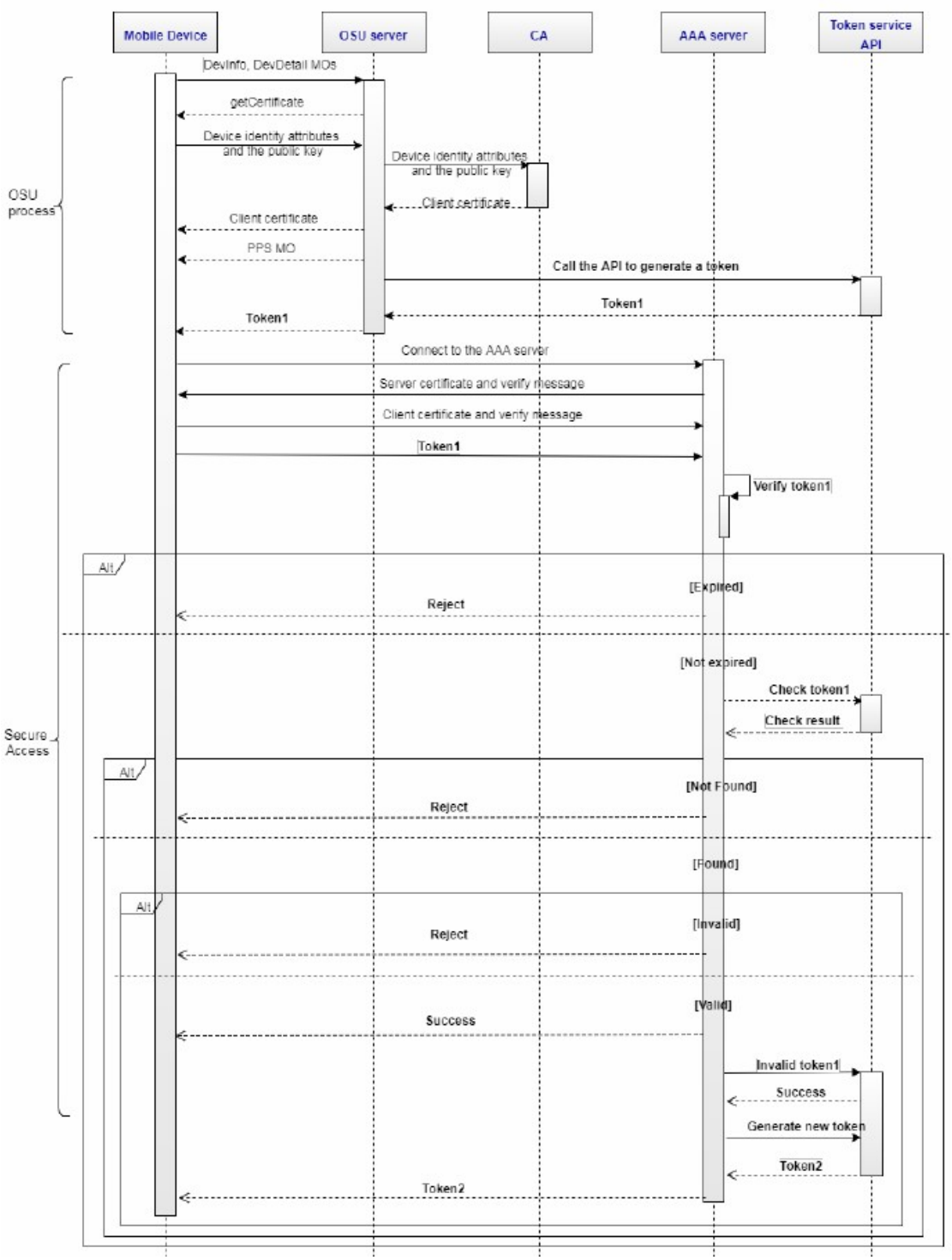

Рис. 3.9. Діаграма послідовності підходу до створення нового токена під час повторного з’єднання з пристроєм А.

В цьому випадку пристрій A може отримати доступ до мережевої послуги, але пристрій A і пристрій B не можуть отримати доступ до послуги одночасно.

Подібно до першого підходу, з одного боку, цей підхід також можна використовувати, щоб дозволити SP виявляти проблему спільного використання облікових даних, коли затримка синхронізації перевищує час між подвійним підключенням одного пристрою до мережі. З іншого боку, цей підхід може запобігти одночасному використанню кількох пристроїв, навіть якщо токени були синхронізовані.

Цей підхід має наступні впливи на архітектуру HS2.2:

1) Структурна складність: деякі компоненти потрібно додати до мережі SP, наприклад API токенів, базу даних токенів і окремий сервер токенів. Сервер токенів використовує API токенів, щоб генерувати токени для клієнтського пристрою, а потім зберігати ці токени в посиланні бази даних токенів на пристрій. Сервер токенів також відповідає за відстеження згенерованих токенів і зміну статусу токенів у базі даних токенів. Наприклад, у процесі OSU, коли сервер OSU викликає службу токенів, сервер токенів надсилає згенерований токен 1 на пристрій A через сервер OSU. Водночас сервер токенів зберігатиме токен 1 у базі даних токенів пристрою A. На етапі безпечного доступу сервер токенів порівнює стан токена, отриманий від клієнтського пристрою, із станом токена, що зберігається в базі даних токенів цього пристрою, а потім надсилає результат перевірки на сервер AAA.

2) Затримка: час OSU і час автентифікації буде подовжено через час для створення та автентифікації токенів. Розширений час можна розділити на дві частини: час генерації та час автентифікації. Наведений вище текст припускає, що сервер щоразу створює лише один токен для клієнтського пристрою. На практиці SP має можливість генерувати кілька токенів для одного клієнтського пристрою. У великомасштабній мережі кількість токенів N, які потрібно згенерувати на день, визначається так:

$$N = n_d * n_t \text{ , } (n_t \geq 1) \quad (3.3)$$

де n_d — середня кількість підключених клієнтських пристроїв, які потребують надання нових токенів на день, n_t — кількість токенів на пристрій. Припустимо, що n_t є фіксованим числом, тоді N буде зростати лінійно разом із кількістю підключених пристроїв, яким потрібні нові токени. Час генерації t_g для SP визначається як:

$$t_{min1} \leq t_g \leq t_{max1 * N} \quad (3.4)$$

де t_{min1} — мінімальний час для генерації токена, а t_{max1} — максимальний час для генерації токена. Оскільки сервер може обробляти кілька запитів на генерацію одночасно, максимальний час генерації має дорівнювати щонайбільше $t_{max1 * N}$, а мінімальний час генерації має бути принаймні рівним t_{min1}. Зауважимо, що генерація токенів може відбуватися офлайн у пакетному режимі. Сервер обчислює N токенів один раз перед тим, як їх надати. Ініціалізація займе час, а також перевірка токенів. Час автентифікації t_{a2} для SP визначається як:

$$t_{min2} \leq t_{a2} \leq t_{max2 * N} \quad (3.5)$$

де t_{min2} — мінімальний час для автентифікації токена, а t_{max2} — максимальний час для автентифікації токена. Оскільки сервер може обробляти кілька запитів на аутентифікацію одночасно, максимальний час генерації має дорівнювати щонайбільше $t_{max2 * N}$, а мінімальний час генерації має бути принаймні рівним t_{min2} . Таким чином, загальний час T на день для SP визначається як:

$$T = t_g + t_{a2} \quad (3.6)$$

Тому на T впливає nd і обчислювальна потужність сервера. Незалежно від обчислювальної потужності сервера T буде збільшуватися разом із nd. Як наслідок, у великомасштабній мережі, якщо є мільйони пристроїв, яким потрібно отримати токени одночасно, це спричинить величезну затримку для SP.

3) Вартість: використання цього підходу призведе до додаткових витрат на обчислення для SP. Вартість обчислення C визначається як:

$$C = c_g * N \quad (3.7)$$

де c_g – середня вартість обчислення одного токена. Тому C буде лінійно зростати разом із nd . Як наслідок, у великомасштабній мережі, якщо існують мільйони пристроїв, які потребують одночасного отримання токенів, це призведе до величезних витрат на обчислення для SP.

4) Пропускна здатність: використання цього підходу займе пропускну здатність мережі та вплине на мережевий трафік між підключеними пристроями та серверами. Ширина смуги B 1 визначається як:

$$B_1 = s_p * n_{rq} / 24 \text{ год} * 3600 \text{ сек} \quad (3.8)$$

де b_{rq} є середнім споживанням пропускної здатності передач під час генерації токена та автентифікації токена для кожного запиту на повторне підключення. n_{rq} – середня кількість запитів на повторне підключення від мобільних пристроїв за день. Таким чином, B_1 вказує на середнє збільшення використання пропускної здатності сервера, а B_1 зростатиме разом із кількістю запитів на повторне підключення. У випадку мільйонів запитів за короткий проміжок часу збільшення споживання пропускної здатності може бути значно значним, що може призвести до зниження пропускної здатності та довшого часу відповіді на кожен запит і навіть перевантаження мережі.

Цей підхід також впливає на стандарт HS2.2, особливо на робочий процес у стандарті. При використанні цього підходу процес OSU і етап безпечного доступу буде змінено. Порівнюючи малюнок 4.8 і малюнок 4.9 , процес генерації токена додається до процесу OSU, а процес автентифікації токена додається до етапу безпечного доступу.

Для двох підходів на основі токенів вони використовують різні параметри, PPS MO ID і токен, але досягають тієї самої мети. У таблиці 4.1 наведено відмінності та схожість між двома підходами на основі токенів. Перш за все, перший підхід використовує параметр PPS MO ID і процес виправлення підписки, який існував у мережах HS2.2, тоді як другий підхід створює нову службу токенів поверх мереж HS2.2. По-друге, задіяні етапи різні. Перший підхід використовується лише на етапі безпечного доступу, тоді як другий підхід бере участь у OSU та етапі безпечного доступу, оскільки токени спочатку генеруються сервером OSU. Крім того, використання параметра відрізняється. Оскільки SP може визначати частоту оновлення, один ідентифікатор PPS MO можна використовувати кілька разів і навіть один раз. Токен призначений для використання лише один раз. Що стосується подібності, обидва ці два підходи змушують зловмисника синхронізувати облікові дані сертифіката з певною частотою, а також запобігають одночасному з’єднанню з використанням тієї самої особи чи облікових даних сертифіката.

3.4. Прикладне застосування пропонованих підходів для системи управління ризиками

Risk Management System (RMS) — це процес, який дозволяє користувачеві ідентифікувати, кількісно оцінити, контролювати ризик і передбачити вплив ризику. Підхід RMS існував протягом багатьох років і застосовувався в депозитарних установах, страхових компаніях, фірмах з цінних паперів, комерційних банках та інших фінансових компаніях [24]. У

галузі фінансових послуг існує багато типів ризиків, таких як процентний ризик, кредитний ризик, ризик неплатоспроможності та інші ризики, які впливають на ці фінансові компанії. Для того, щоб оптимізувати прибуток для окремої особи або компанії з мінімальним ризиком, RMS застосовується для вимірювання та управління ризиками. Наприклад, в [10] запропонували підхід до управління ризиками для оцінки системної фінансової стабільності банківської системи. Загалом, RMS приймає процес оцінки, який включає ідентифікацію ризику, оцінку та подальше управління [7]. Наприклад, у додатку для мобільних платежів RMS ґрунтується на виявленні та стримуванні шахрайства. RMS спочатку збирає статус облікового запису користувача, кредитну картку, історію транзакцій та іншу інформацію, щоб у режимі реального часу перевірити, чи обліковий запис зламано чи ні. Після того, як RMS виявить ненормальну онлайн-транзакцію або скомпрометований мобільний пристрій, мобільний платіжний додаток застосує заходи стримування, щоб обмежити доступ до активів і зупинити транзакцію.

Завдяки функціям RMS його також можна застосовувати в середовищі HS2.2, щоб дозволити SP виявляти та полегшувати проблему спільного використання облікових даних сертифіката. Рисунок 3.10 ілюструє RMS для SP в мережах HS2.2. RMS складається з наступних блоків:

- Облікові дані сертифіката: облікові дані сертифіката стосуються сертифіката клієнта та повідомлення перевірки, підписаного закритим ключем пристрою.
- Параметр на основі токенів (автентифікація на основі токенів): Параметр на основі токенів відноситься до PPS MO ID або токенів, згенерованих пристроєм за допомогою підходів автентифікації на основі токенів.
- Перевірка: під перевіркою розуміється процес, під час якого сервер AAA SP перевіряє та перевіряє облікові дані, надані клієнтським пристроєм. У початковому процесі клієнтський пристрій має надати облікові

дані сертифіката серверу AAA SP під час автентифікації EAPTLS. Сервер AAA перевірить цілісність і дійсність сертифіката, а також підтвердить ідентичність клієнтського пристрою, пов'язаного з отриманим сертифікатом клієнта. У новому процесі RMS, окрім облікових даних сертифіката, сервер AAA має перевірити параметри на основі токенів.

- Виправлення підписки: цей блок відноситься до процесу виправлення підписки, який SP може вимагати від клієнтського пристрою виконати після етапу перевірки.
- Інформація про місцезнаходження та споживання (віддалений моніторинг): цей блок стосується інформації про місцезнаходження та споживання, зібраної за допомогою підходу віддаленого моніторингу
- Ідентифікація пристрою (пасивне зняття відбитків пальців): цей блок стосується ідентифікаційної інформації пристрою, зібраної за допомогою підходу пасивного зняття відбитків пальців.
- Оцінка ризиків і процес прийняття рішень: цей блок відноситься до оцінки ризиків і процесу прийняття рішень. Згідно з інформацією попередніх блоків, сервер прийняття рішень оцінить ризик і прийме рішення щодо доступу щодо запиту доступу підключеного пристрою.
- Ідентифікація пристрою (активний відбиток пальця) : цей блок стосується ідентифікаційної інформації пристрою, зібраної за допомогою підходу активного відбитка пальця.
- Облікові дані OSU: облікові дані OSU стосуються інформації, наприклад кредитної картки, номера банківського рахунку та іншої особистої інформації, яку користувач надає SP для створення облікового запису на етапі реєстрації.
- Correction: цей блок відноситься до заходів виправлення, які SP може вжити, щоб полегшити проблему спільного використання облікових даних сертифіката після її виявлення. Наприклад, SP може відкликати сертифікат або додати піратський пристрій до чорного списку, якщо він повторно підключиться до мережі SP.

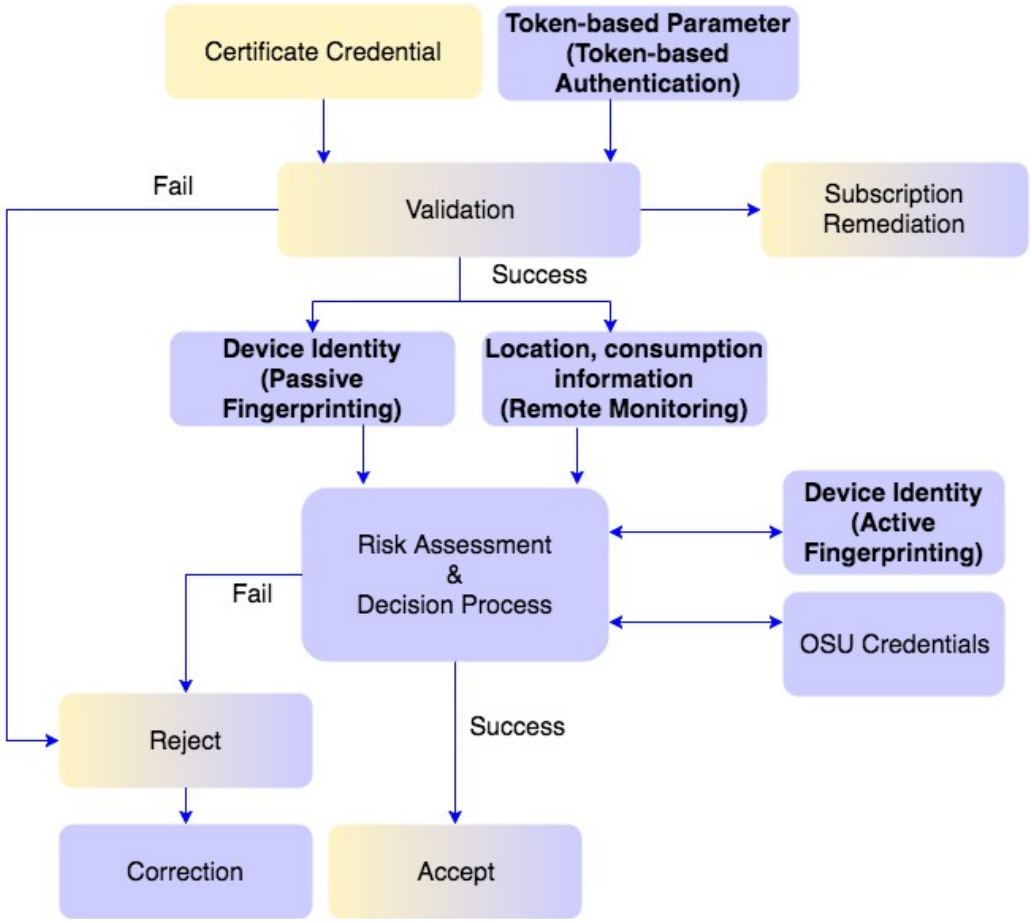

Рис. 3.10. Система управління ризиками для SP в мережах HS2.2

- Reject: цей блок відноситься до повідомлення про відхилення доступу, яке сервер AAA надсилає на клієнтський пристрій. Це вказує на те, що SP відхиляє запит на доступ від підключеного пристрою, і клієнтський пристрій не може отримати доступ до своєї служби підписки.
- Accept: цей блок відноситься до повідомлення про прийняття доступу, яке сервер AAA надсилає на клієнтський пристрій. Це вказує на те, що SP приймає запит на доступ від підключеного пристрою, і клієнтський пристрій може отримати доступ до його служби підписки.

Жовта частина на рисунку 3.10 показує оригінальний процес перевірки та автентифікації в стандарті HS2.2. Фіолетова частина малюнка показує нещодавно доданий процес перевірки та автентифікації на основі п’яти підходів.

У початковому процесі клієнтський пристрій має надати облікові дані свого сертифіката серверу AAA SP під час автентифікації EAP-TLS. Сервер AAA перевірить цілісність і дійсність сертифіката, а також підтвердить ідентичність клієнтського пристрою, пов'язаного з отриманим сертифікатом клієнта. Якщо перевірка не вдасться, SP відхилить запит доступу від пристрою. Якщо перевірка пройшла успішно, SP прийме запит на доступ і дозволить доступ до мережі. Крім того, SP запитає клієнтський пристрій виконати процес виправлення підписки, якщо термін дії сертифіката майже закінчиться. Однак, якщо піратський пристрій витягує достатньо інформації з законного пристрою, а потім імітує цей пристрій, SP не може виявити спільний доступ до облікових даних сертифіката, і перевірка завжди буде успішною. Рисунок 3.11 ілюструє робочий процес RMS.

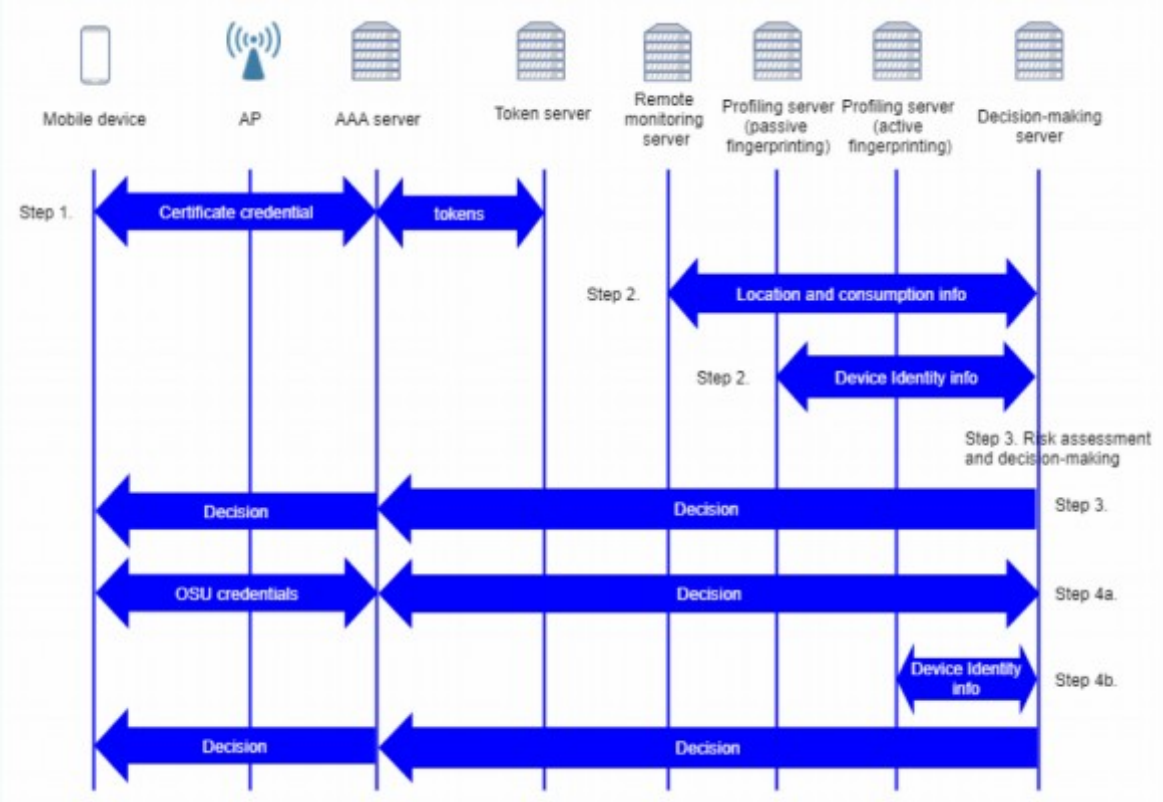

Рис. 3.11. Схема системи управління ризиками

Крок 1. Окрім облікових даних сертифіката, клієнтський пристрій також має надавати параметри на основі токенів, токени, згенеровані

серверами SP, і PPS MO ID, з метою виявлення. Подібно до перевірки сертифіката, сервер AAA також має перевірити дійсність токена та перевірити статус цього токена в посиланні бази даних токенів на клієнтський пристрій. Перевірка дійсності призначена для перевірки того, чи закінчився термін дії токена. Перевірка статусу токена призначена для перевірки, чи пов'язаний токен із цим конкретним пристроєм і чи використовувався він чи ні. Якщо токен не пов'язано з пристроєм, він не зберігатиметься в посиланні бази даних токенів на цей пристрій. Іншими словами, ідентичний токен не може бути знайдений у базі даних токенів. Якщо токен використовувався для автентифікації, SP встановить його статус як недійсний у базі даних токенів. Оскільки SP не може передбачити, коли клієнтський пристрій знову підключиться до мережі, час закінчення буде встановлено відповідно до політики SP. Отже, можливо, термін дії токена не закінчився, але він уже був використаний.

У результаті на етапі перевірки сервер AAA надішле клієнтському пристрою відхилення доступу за таких трьох ситуацій:

1) Посвідчення пристрою, що міститься в сертифікаті клієнта, відрізняється від посвідчення, яке зберігається в серверній базі даних;

2) Повідомлення, підписане закритим ключем, не можна перевірити відкритим ключем, який пристрій надіслав на сервер OSU під час процесу реєстрації сертифіката клієнта;

3) Токен не знайдено в посиланні бази даних токенів на клієнтський пристрій;

4) Статус токена в базі даних токенів недійсний. Процес перевірки токена такий самий, як показано на рисунку 3.9.

Сервер AAA запитає клієнтський пристрій на процес виправлення підписки, коли:

1) Термін дії сертифіката клієнта закінчився;

2) Термін дії токена закінчився.

Крок 2. Далі, якщо перевірка як облікових даних сертифіката, так і параметрів на основі токена пройшла успішно, сервер прийняття рішень повинен зібрати інформацію про пристрій, наприклад, географічне розташування пристрою, використання послуги підписки та ідентифікацію пристрою, за допомогою дистанційного моніторингу та пасивного зняття відбитків пальців. Ця інформація про пристрій використовуватиметься як вхідні дані для оцінки ризиків і процесу прийняття рішень.

Як пояснюється в розділі 3.1, використання параметрів на основі токенів може допомогти SP виявити наявність спільного використання облікових даних сертифіката, коли неавторизований пристрій B не синхронізувався з авторизованим пристроєм A. Після успішного завершення синхронізації використання параметрів на основі токенів стає неефективним при виявленні спільного використання облікових даних. Оскільки синхронізація програмного забезпечення не може вплинути на інформацію про місцезнаходження та використання служби клієнтського пристрою, дистанційний моніторинг усе ще можна використовувати для виявлення спільного використання облікових даних сертифіката. Окрім цієї інформації, сервер AAA також має збирати інформацію для ідентифікації пристрою, щоб відрізнити легальні пристрої від піратських. Використовуючи підхід пасивного зняття відбитків пальців, SP може профілювати кожен клієнтський пристрій під час зв’язку з пристроєм з метою виявлення ідентифікації та виявлення спільного використання.

Крок 3. На цьому етапі SP оцінить ризик і прийме рішення відповідно до заданої функції f.

Після обчислення є три ситуації:

1) Якщо значення f знаходиться між 0 і threshold low , це означає, що SP візьме на себе ризик і прийме запит на доступ від підключеного пристрою;

2) Якщо значення f знаходиться між високим порогом і 100, це означає, що SP не ризикне і відхилить запит доступу від підключеного пристрою;

3) Якщо значення f знаходиться між низьким порогом і високим порогом , це означає, що SP використовуватиме інші методи для переоцінки ризику.

Крок 4a. У першій ситуації на кроці 3, якщо пристрій отримує дозвіл на доступ, він отримає доступ до мережі та використовуватиме послугу підписки. У другому випадку, якщо пристрій отримає відхилення доступу, він від'єднається від точки доступу або мережі гарячої точки. Для третьої ситуації SP може вимагати, щоб користувач вручну надав облікові дані OSU, наприклад кредитну картку, номер банківського рахунку та іншу особисту інформацію, щоб прийняти рішення.

1) Якщо користувач не може вручну надати ті самі облікові дані OSU, які він/вона надав на етапі реєстрації, то SP відхилить запит доступу від підключеного пристрою;

2) Якщо користувач може вручну надати ті самі облікові дані OSU, тоді SP отримає доступ до запиту доступу з підключеного пристрою;

Крок 4b. В якості альтернативи SP може використовувати активний підхід відбитків пальців, коли SP надсилає певні коди як виклик на пристрій і аналізує відповідь на виклик для виявлення ідентифікатора пристрою.

1) Якщо ідентифікатор підключеного пристрою збігається з інформацією, що зберігається в базі даних на стороні сервера, тоді SP отримає доступ до запиту доступу від підключеного пристрою;

2) Якщо ідентифікаційні дані підключеного пристрою суттєво відрізняються від інформації, що зберігається в базі даних на стороні сервера, тоді SP відхилить запит на доступ від підключеного пристрою. Оскільки користувач може змінити відбиток пальця пристрою, наприклад, оновити ОС, цей RMS дозволяє вносити невеликі зміни з часом.

RMS використовує комбінований підхід для полегшення проблеми спільного використання облікових даних сертифіката, що складається з автентифікації на основі токенів, віддаленого моніторингу та відбитків пальців пристрою. На основі цих підходів SP може використовувати цю

систему для визначення ідентифікації пристрою та наявності спільного використання облікових даних сертифіката, а потім вжити заходів для пом’якшення проблеми за допомогою функції роз’єднання або функції деавтентифікації.

Висновки до розділу

Для зменшення ризиків, пов'язаних зі спільним використанням облікових даних сертифікатів в мережах HS2.2, пропонується комплекс заходів, спрямованих на попередження або виявлення таких порушень. У даному розділі представлено систематизований огляд підходів до вирішення цієї проблеми, які умовно поділено на категорії: автентифікація на основі токенів, віддалений моніторинг та методи безпечного зберігання. Наприкінці розділу проведено порівняльний аналіз запропонованих підходів та сформульовано висновки щодо їх ефективності та застосовності в різних сценаріях.

ВИСНОВКИ

У магістерській роботі досліджено моделі та методи спільного використання облікових даних на основі сертифікатів у середовищі HS2.2. Основні внески цієї роботи включають:

- Визначення проблеми спільного використання сертифікатів у мережах HS2.2;

- Пропозиція кількох підходів для запобігання спільному використанню сертифікатів або його виявлення, а також оцінка кожного підходу з точки зору їх впливу на архітектуру та стандарт HS2.2;

- Розробка алгоритму роботи системи управління ризиками, яка інтегрує більшість запропонованих підходів.

Перше завдання цієї роботи полягало у визначенні наявності проблеми для HS2.2 та ідентифікації інформації, необхідної для передачі між легітимним та піратським пристроями для спільного використання сертифікатів. Аналіз двох сценаріїв — випадкового обміну та викрадення облікових даних — дозволив визначити ключові елементи, необхідні для завершення обміну: PPS MO, клієнтський сертифікат, закритий ключ пристрою, DevInfo MO та DevDetail MO. Було також досліджено технічну можливість отримання цієї інформації з легітимного пристрою.

Друге завдання полягало в дослідженні методів виявлення або запобігання обміну сертифікатами. Для виявлення використано два підходи на основі токенів (генерація нових токенів та збільшення частоти виправлення), які дозволяють виявляти спільне використання сертифікатів, коли піратський пристрій підключається до мережі HS2.2 без синхронізації з легітимним пристроєм. Також застосовано підхід віддаленого моніторингу для ідентифікації одночасної присутності двох пристроїв з однаковими ідентифікаторами в різних місцях. Крім того, для ідентифікації пристрою та виявлення спільного використання було використано два підходи на основі відбитків пальців пристрою — активний і пасивний.

Для запобігання обміну запропоновано два підходи на основі токенів, які перешкоджають одночасному доступу до підписки, оскільки токени призначені для одноразового використання. Крім того, застосовано підхід безпечного зберігання, який запобігає передачі необхідної інформації від легітимного пристрою до піратського через надійне зберігання цих даних. У роботі також проведено аналіз наслідків кожного підходу для архітектури та стандарту HS2.2, а також обговорено їхні обмеження при вирішенні проблеми спільного використання сертифікатів.

Останнім завданням стало розроблення алгоритму управління ризиками для постачальників послуг, яка дозволяє виявляти випадки спільного використання сертифікатів та ефективно вирішувати цю проблему.

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Martin Abadi, Krishna Bharat, and Johannes Marais. System and method for generating unique passwords, October 31 2000. US Patent 6,141,760. 2
2. Open Mobile Alliance. Oma device management standardized objects. Approved Version, 1, 2008. 8
3. Alysson Bessani, Miguel Correia, Bruno Quaresma, Fernando Andr e, and Paulo Sousa. Depsky: dependable and secure storage in a cloud-of-clouds. ACM Transactions on Storage (TOS), 9(4):12, 2013. 42
4. Christine Blakemore, Jo~ao Redol, and Miguel Correia. Fingerprinting for web applications: from devices to related groups. In Trustcom/BigDataSE/I SPA, 2016 IEEE, pages 144{151.IEEE, 2016. 37
5. John Jules Alexander Boyer and Eric Fernand Le Saint. Secure digital credential sharing arrangement, September 21 2010. US Patent 7,802,293. 2
6. Sergey Bratus, Cory Cornelius, David Kotz, and Daniel Peebles. Active behavioral ngerprinting of wireless devices. In Proceedings of the rst ACM conference on Wireless network security, pages 56{61. ACM, 2008. 38
7. Ken Brown and Peter Moles. Credit risk management. Edinburgh Business School, Heriot-Watt University, UK, 2012. 44
8. Emanuele Cesena, Hans L ohr, Gianluca Ramunno, Ahmad-Reza Sadeghi, and Davide Vernizzi. Anonymous authentication with tls and daa. In International Conference on Trust and Trustworthy Computing, pages 47{62. Springer, 2010. 2
9. Dave Cooper. Internet x. 509 public key infrastructure certicate and certicate revocation list (crl) prole. 2008. 16
10. Helmut Elsinger, Alfred Lehar, and Martin Summer. Risk assessment for banking systems. Management science, 52(9):1301{1314, 2006. 44
11. Ana Ferreira, Jean-Louis Huynen, Vincent Koenig, and Gabriele Lenzini. Socio-technical security analysis of wireless hotspots. In International

Conference on Human Aspects of Information Security, Privacy, and Trust, pages 306{317. Springer, 2014. 1

12. Benjamin D Goldstein. Method and apparatus for secure storage of data, October 5 1999. US Patent 5,963,642. 42 54 Credential Sharing based on Hotspot 2.0 Release 2 Specication
13. Suman Jana and Sneha K Kasera. On fast and accurate detection of unauthorized wireless access points using clock skews. IEEE Transactions on Mobile Computing, 9(3):449{462, 2010. 40
14. Robert N Johnson, Ronald D Smith, Charlotte K Smith, Edward C Kight, and George H Harrop. Smart remote monitoring system and method, April 22 2003. US Patent 6,553,336. 35
15. Tadayoshi Kohno, Andre Broido, and Kimberly C Cla y. Remote physical devicengerprinting. IEEE Transactions on Dependable and Secure Computing, 2(2):93{108, 2005. 40
16. Dongjiang Li, Cheng Cheng, and Bo Zhang. Vehicle remote monitoring system based on android. In Software Engineering and Service Science (ICSESS), 2016 7th IEEE International Conference on, pages 722{725. IEEE, 2016. 35
17. Takashi Matsunaka, Akira Yamada, and Ayumu Kubota. Passive osngerprinting by dns tracanalysis. In Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference On, pages 243{250. IEEE, 2013. 40
18. K. Metal and F. Hydraulic. Wi-Fi Alliance Hotspot 2.0 (Release 2) Technical Specication, December 2016. 1, 3, 5, 8, 9, 12
19. Phyllis Michaelides. Generic token-based authentication system, December 9 2003. US Patent App. 10/731,629. 2
20. Abdullah Na, William Isaac, Shashank Varshney, and Ekram Khan. An iot based system for remote monitoring of soil characteristics. In Information Technology (InCITe)-The Next Generation IT Summit on the Theme-

Internet of Things: Connect your Worlds, International Conference on, pages 316{320. IEEE, 2016. 35

21. Nalini K. Ratha, Jonathan H. Connell, and Ruud M. Bolle. Enhancing security and privacy in biometrics-based authentication systems. IBM systems Journal, 40(3):614{634, 2001. 2
22. Barry Ribbeck. Public key infrastructure. 2016. 16
23. Kaimin Ruan, Chao Hu, Weixing Lin, Xianli Wang, and Changzhu Song. Remote monitoring system for family health examination. In Information and Automation (ICIA), 2016 IEEE International Conference on, pages 148{153. IEEE, 2016. 35
24. Anthony Saunders and Marcia Millon Cornett. Financial institutions management: A risk management approach. Irwin/McGraw-Hill, 2003. 44
25. Aashish Sharma, Zbigniew Kalbarczyk, R Iyer, and James Barlow. Analysis of credential stealing attacks in an open networked environment. In Network and System Security (NSS), 2010 4th International Conference on, pages 144{151. IEEE, 2010. 2
26. Chao Shen, Ruiyuan Lu, Saeid Samizade, and Liang He. Passive ngerprinting for wireless devices: A multi-level decision approach. In Identity, Security and Behavior Analysis (ISBA), 2017 IEEE International Conference on, pages 1{6. IEEE, 2017. 40 Credential Sharing based on Hotspot 2.0 Release 2 Specication
27. Dan Simon, Bernard Aboba, and Ryan Hurst. The eap-tls authentication protocol. Technical report, 2008. 12
28. John A Soltesz. System for the secure storage and transmission of data, June 25 1991. US Patent 5,027,401. 42
29. Parekh Tanvi, Gawshinde Sonal, and Sharma Mayank Kumar. Token based authentication using mobile phone. In Communication Systems and Network Technologies (CSNT), 2011 International Conference on, pages 85{88. IEEE, 2011. 2

30. Bhavani Thuraisingham, XiaoFeng Wang, and Vinod Yegneswaran. Security and Privacy in Communication Networks: 11th International Conference, SecureComm 2015, Dallas, TX, USA, October 26-29, 2015, Revised Selected Papers, volume 164. Springer, 2016.
31. WiFiAlliance. Wi-Fi CERTIFIED Passpoint (Release 2) Deployment Guidelines. December 2016. 4, 7
32. Harkins, D. (2020). Secure Hotspot 2.0 Wireless Networking Using WPA3 and Enhanced Privacy. IEEE Communications Standards Magazine, 4(1), 72-78.
33. Seyedzadegan, M., & Othman, M. (2015). Hotspot 2.0 Technology: Standardization and Security. IEEE Communications Surveys & Tutorials, 17(4), 2164-2189.
34. Bertino, E., Paci, F., & Ferrini, R. (2019). Privacy-Aware Role-Based Access Control for Smart Cities and IoT Systems. IEEE Access, 7, 5461-5473.
35. Liao, C. H., & Hsiao, Y. D. (2016). A Comprehensive Survey on Token-based Authentication Mechanisms in Wireless Networks. Journal of Network and Computer Applications, 71, 42-55.
36. Khan, M. A., Salah, K., & Kalutarage, H. (2018). IoT Security: Review, Blockchain Solutions, and Open Challenges. Future Generation Computer Systems, 82, 395-411.
37. Saxena, N., Roy, A., & Kim, N. (2019). Enhanced Authentication for 5G Enabled IoT Systems Using Blockchain. IEEE Access, 7, 67700-67712.
38. Kumar, S., & Zhang, J. (2017). Device Fingerprinting to Enhance Security in Wireless Networks: A Survey. IEEE Communications Surveys & Tutorials, 19(1), 850-876.
39. Farris, I., Borgia, E., & Paolino, M. (2017). Improving Authentication in Wi-Fi Networks: The Impact of Passpoint and EAP Extensions on User Experience and Security. IEEE Transactions on Wireless Communications, 16(11), 7490-7505.

40. Krawczyk, H., & Bellare, M. (2016). HMAC: Keyed-Hashing for Message Authentication. IETF RFC 2104.
41. Jiang, W., Chen, X., & Cheng, X. (2017). Towards Efficient and Secure Certificate-based Authentication in Wireless Networks. IEEE Transactions on Wireless Communications, 16(12), 8070-8083.
42. Grassi, P. A., Garcia, M. E., & Fenton, J. L. (2017). Digital Identity Guidelines: Authentication and Lifecycle Management. National Institute of Standards and Technology (NIST), Special Publication 800-63B.
43. Liu, Y., He, Y., & Xie, W. (2019). Token-Based Authentication in Decentralized Access Control Systems: Survey and Challenges. IEEE Access, 7, 123456-123471.
44. Choudhury, S., Chattopadhyay, S., & Debnath, P. (2018). A Secure Authentication Protocol for Wireless Networks Based on Certificate-less Public Key Cryptography. Journal of Information Security and Applications, 40, 17-25.
45. Arfaoui, G., Ahmim, A., & Fournier-Viger, P. (2020). Security and Privacy in Wi-Fi Networks: Current Status and Future Directions. IEEE Communications Surveys & Tutorials, 22(4), 2342-2362.
46. Stallings, W. (2017). Cryptography and Network Security: Principles and Practice. Pearson.
47. Aboba, B., & Simon, D. (2016). Extensible Authentication Protocol (EAP) Key Management Framework. IETF RFC 5247.
48. Feng, X., & Zhu, H. (2018). Secure Certificate-Based Authentication Protocol for Wireless Mesh Networks. International Journal of Distributed Sensor Networks, 14(3), 1-8.
49. Zhang, Y., Zhang, J., & Fang, Y. (2019). Security and Privacy in Mobile Hotspots: A Survey of Wireless Technologies and Challenges. IEEE Communications Surveys & Tutorials, 21(2), 1322-1350.