ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

КВАЛІФІКАЦІЙНА РОБОТА

НА ТЕМУ: Модель децентралізованої системи автентифікації

ВИКОНАВ:
Студент гр. КСМзм-21-1 Шокота Т. А.

КЕРІВНИК:
к.т.н. доц. Ільїна І. В.

ХАРКІВ
2022р.

Мета роботи

Метою даної роботи є розробка для мережі Інтернет єдиної та універсальної системи ідентифікації та автентифікації, яка відповідає критеріям, сформульованим нижче. Під системою ідентифікації розумітимемо:

– сховище особистостей, кожна з яких асоціюється з певною реальною людиною.
– спосіб, яким людина може довести володіння цією особою (автентифікація).

Під системою аутентифікації розумітимемо набір наступних структур:

– сховище даних, де містяться зашифровані персональні дані користувачів та їх публічні ключі.
– вузли мережі, що обслуговують сховище даних та відповідають за логіку процесу автентифікації.
– додаток-клієнт, здатний приймати та підтверджувати запити на автентифікацію, пов'язані з конкретною особистістю.
– протокол, яким взаємодіють перелічені вище частини системи.
– бібліотеки, за допомогою яких розробник легко зможе вбудувати аутентифікацію протоколу на свій сайт.

Завдання

Для досягнення поставленої мети вирішуються наступні задачі:

- аналіз існуючих протоколів аутентифікації, їх переваг та недоліків.
- підбір відповідного протоколу аутентифікації та наявної інфраструктури (якщо така у нього є): сховище особистостей, сховище даних, вузли мережі, додаток-клієнт.
- розробка бібліотек розробника та розповсюдження їх по каналах Open Source.
- розробка тестового сайту, який використовує систему автентифікації.
- аналіз рішення на зазначені вище критерії.

3

Зберігання паролей у незашифрованому вигляді на боці постачальника послуг

Переваги:

- Простота реалізації.
- Можливість відновлення старого втраченого пароля.

Недоліки:

- нестійкість до злому сховища.Система аутентифікації з таким методом зберігання паролів є незахищеною. При зломі постачальника послуг усі паролі стають доступними хакеру, і він може у будь-який час автентифікуватись під виглядом іншого користувача. Дані користувача знаходяться під великою загрозою.
- неуніверсальність. Користувач змушений щоразу заводити новий обліковий запис для кожного постачальника послуг з таким методом автентифікації. Оскільки заводити однакові пари логін-пароль вкрай небезпечно (а часто просто неможливо), це створює велику плутанину.

4

Зберігання паролів у зашифрованому вигляді на стороні постачальника послуг

Переваги:

– стійкість до злому сховища. Навіть у разі злому постачальника послуг хакер отримує не самі паролі, а результати застосування алгоритмів формування ключа до паролів. Такі алгоритми спеціально розроблені так, щоб ускладнити отримання пароля.

Недоліки:

– неуніверсальність. Користувачі все ще вимушені заводитиокремі облікові записи для кожного постачальника послуг.

5

OAuth, OpenID

Переваги:

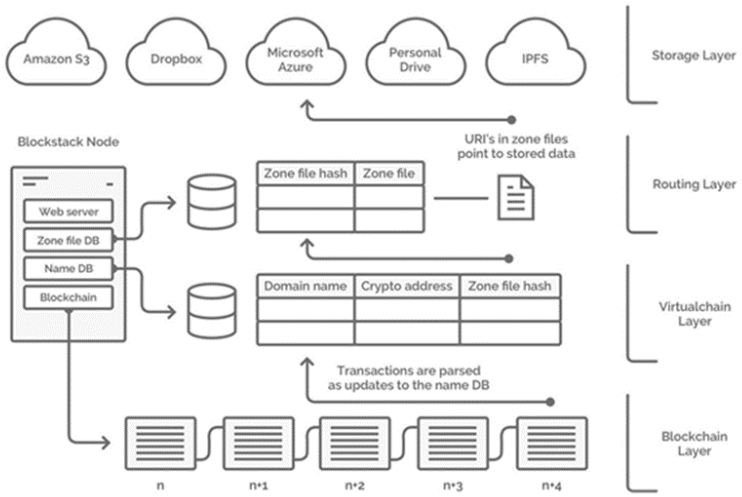
– універсальність. Маючи обліковий запис у identity provider, можна заходити під своїм ім'ям на будь-які сайти, що підтримують OpenID або OAuth. Такий обліковий запис служить універсальною мережевою особистістю і дозволяє використовувати одні й самі способи аутентифікації на безлічі майданчиків.
– Відкритість. OpenID та OAuth — повністю відкриті стандарти, не захищені патентами та копірайтом.

Недоліки:

– довіра до третього боку. Користуючись OpenID або OAuth, користувачі змушені довіряти аутентифікацію деякої третьої сторони. З цього випливає:
– нестійкість до зламів. Користувач не може контролювати зберігання своїх даних на серверах третьої сторони і теоретично вони можуть зберігатися там у відкритому вигляді.
– відсутність анонімності. Identity provider бачить і логує дії клієнта: коли він аутентифікувався, куди, з якої IP-адреси, як часто.
– єдина точка відмови. Якщо сервер автентифікації третьої сторони виявиться на якийсь час недоступним, або третя сторона зовсім припинить існування, користувач втратить можливість аутентифікуватися під своїм єдиним ім'ям.

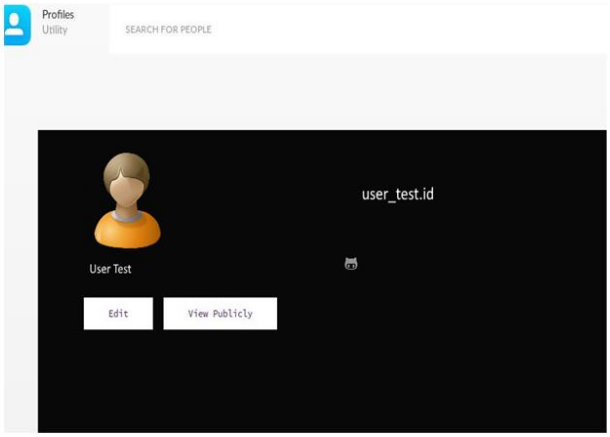
6

Модель системи автентифікації



7

Blockstack Portal



8

Реалізація поставлених завдань включає

1. аналіз відкритого протоколу автентифікації Blockstack, його інфраструктури, виявлення переваг та недоліків, використання його для реалізації бібліотек розробника blockchainauth та django-blockstack.
2. розробку на основі протоколу Blockstack Python-бібліотеки blockchainauth.
3. розробку Django-додатку django-blockstack, що використовує blockchainauth.
4. розробка тестового веб-сайту, що використовує django-blockstack для автентифікації користувачів.

9

Процес аутентифікації

Рисунок 1

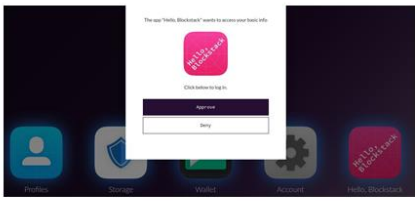


Рисунок 2

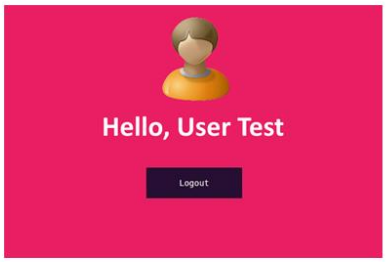


Рисунок 3

10

manifest.json

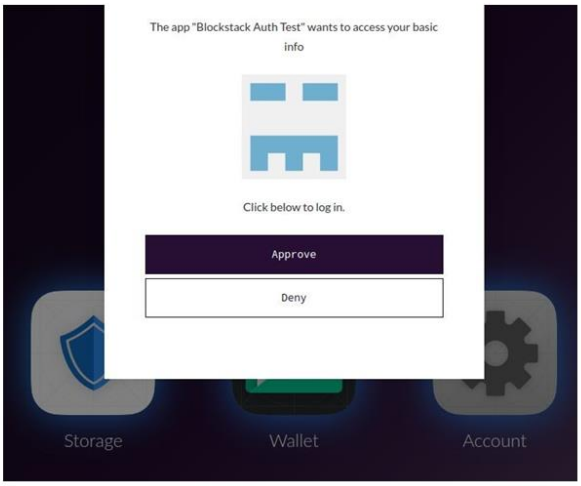
```
{
"name": "Hello, Blockstack", "start_url":
"localhost:5000",
"description": "A simple demo of Blockstack
Auth", "icons": [{
"src":
"https://helloblockstack.com/icon-
192x192.png", "sizes": "192x192",
"type": "image/png"
}]
}
```



```
"header": {"type": "JWT", "alg": "ES256K"}, "payload": {
"domain_name": "http://localhost:5000", "exp": 1493412486,
"iat": 1493408886,
"iss":
"did:btc-addr:1NZNxhoxobqwsNvTb16pdeiqvFvce3Yg8U", "jti":
"75719c8a-3679-45b7-9551-21b6dfc28444",
"manifest_uri": "http://localhost:5000/manifest.json",
"public_keys":
["027d28f9951ce46538951e3697c62588a87f1f1f295de4a14fdd4
c780fc52cfe69"],
"redirect_uri": "http://localhost:5000", "scopes": []
},
"signature":
"HBwhcgPj7hrKg_IOGyaMJ9L-U_kwE5EweK8H54E2fuNONeWElfJg-h
10LJvbwrvf_3TcgzQRbqdxGSmro8Ey6A"
}
```

Домашня сторінка тестового сайту. Користувач не автентифікований

Інтерфейс Blockstack Portal, запит на автентифікацію тестового сайту



14

Висновок

У цій роботі було проаналізовано підходи до аутентифікації, що застосовуються у світі. Були виявлені критичні недоліки, такі як централізованість та залежність процесу аутентифікації від третіх сторін. Аналіз великих витоків інформації та зловживань із боку постачальників імен показав серйозність цих недоліків.

Як відправна точка було обрано перспективний протокол автентифікації децентралізованої системи імен Blockstack. За рахунок зберігання імен у блокчейні він сам по собі вирішував наведені вище недоліки, проте через свою новизну був занадто незручний як для звичайних користувачів, так і для розробників веб-додатків, які могли б вбудувати автентифікацію через Blockstack у свою програму.

Вклад цієї роботи — спрощення встановлення системи автентифікації Blockstack для розробників веб-застосунків.