Towards Enhancing the Security of OAuth Implementations In Smart Phones

Mohamed Shehab and Fadi Mohsen

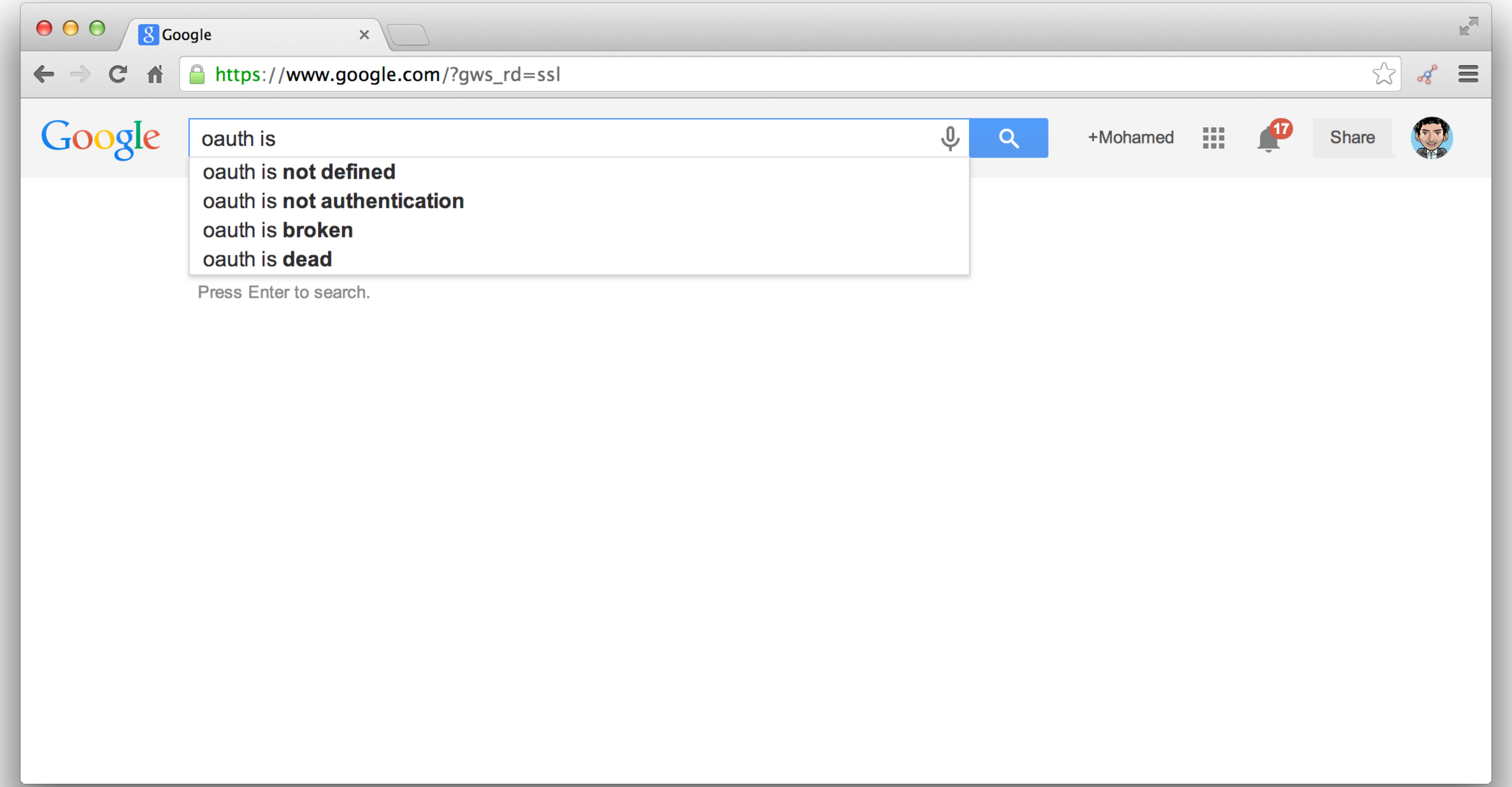
Department of Software and Information Systems
College of Computing and Informatics
University of North Carolina at Charlotte

UNC CHARLOTTE

IEEE MS 2014. Anchorage, AK

Introduction

- **What is OAuth?**



oauth is

oauth is **not defined**

oauth is **not authentication**

oauth is **broken**

oauth is **dead**

Press Enter to search.

Introduction

- The Open Authorization (OAuth) standard, enables the resource owner *(user)* to grant permissions to a third-party *(mobile app)* access to their resources that are hosted on a resource provider *(Facebook)*.

- With OAuth, the users are no longer required to share their credentials with third party apps in order to grant them authorizations.

- Who uses OAuth? All major service and resource providers such as Google, Facebook, Microsoft, Twitter, Dropbox, GitHub, Salesforce and many others.

Introduction (OAuth Flow)

Introduction (OAuth Flow)

- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**

Introduction (OAuth Flow)

- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**
- **These access tokens have limited *scope*. Which limits the permissions granted to the third party app.**

Introduction (OAuth Flow)

- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**

- **These access tokens have limited *scope*. Which limits the permissions granted to the third party app.**

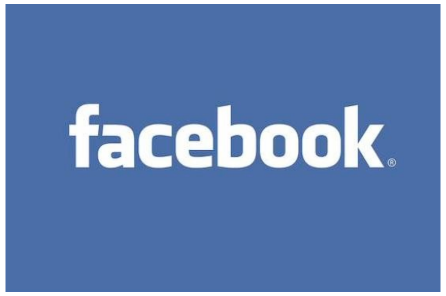
User
(Resource Owner)

Introduction (OAuth Flow)

- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**

- **These access tokens have limited *scope*. Which limits the permissions granted to the third party app.**

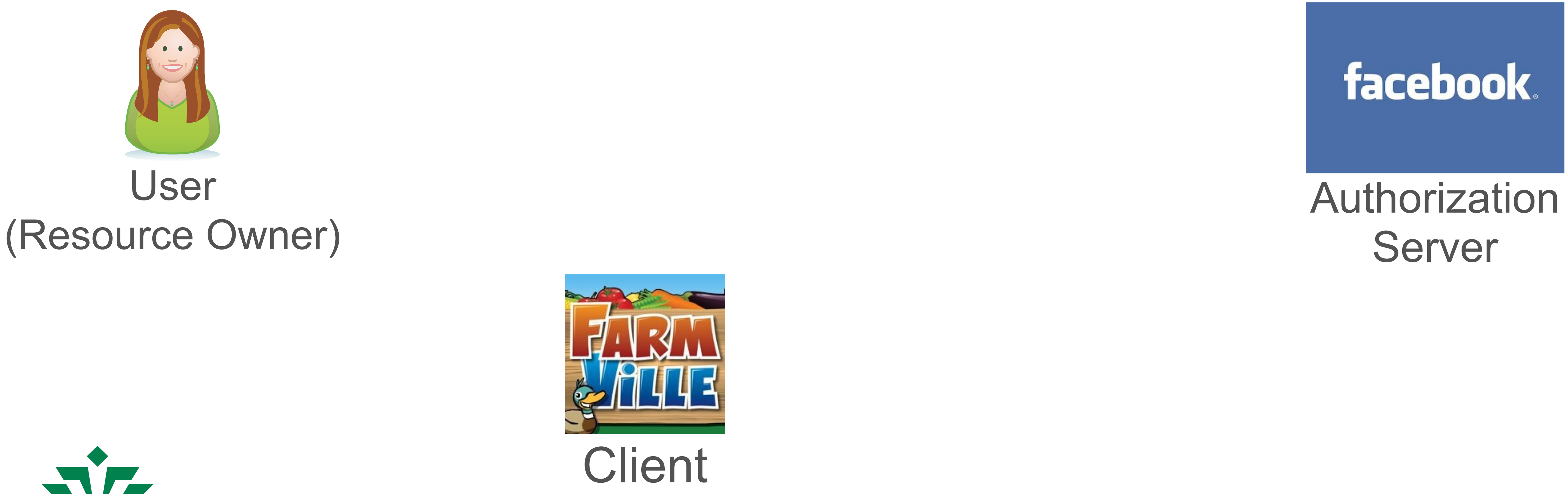
User
(Resource Owner)



Authorization
Server

Introduction (OAuth Flow)

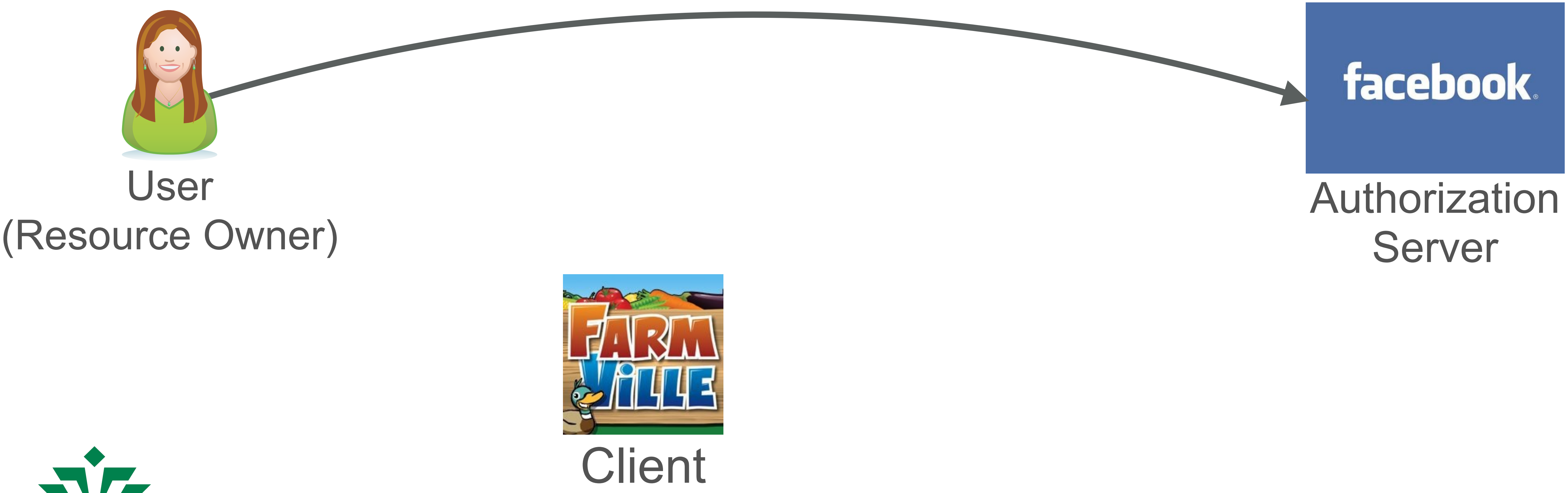
- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**
- **These access tokens have limited *scope*. Which limits the permissions granted to the third party app.**



Introduction (OAuth Flow)

- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**

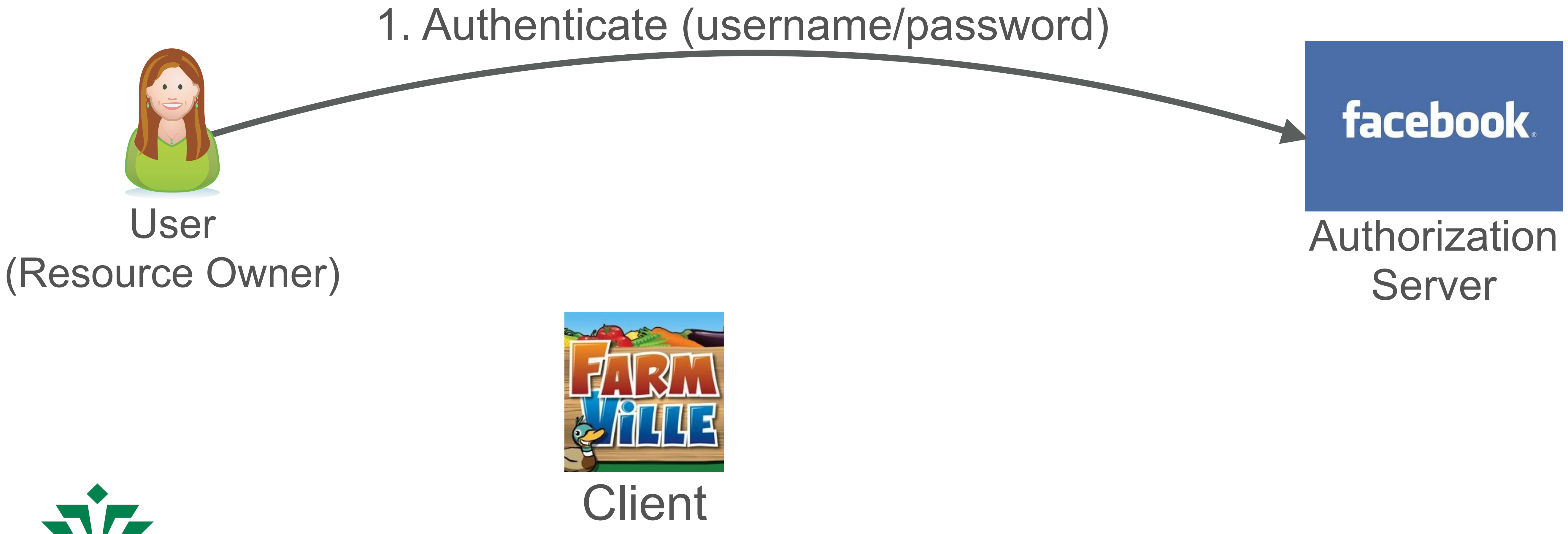
- **These access tokens have limited *scope*. Which limits the permissions granted to the third party app.**



Introduction (OAuth Flow)

- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**

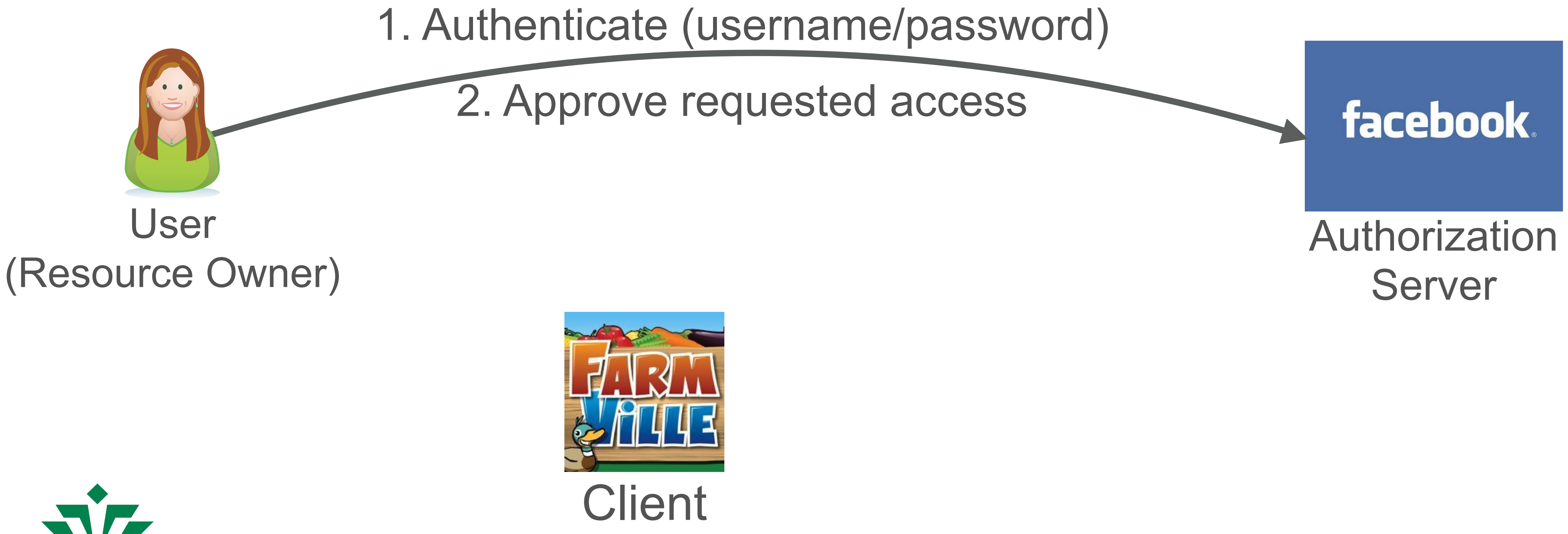
- **These access tokens have limited scope. Which limits the permissions granted to the third party app.**



Introduction (OAuth Flow)

- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**

- **These access tokens have limited *scope*. Which limits the permissions granted to the third party app.**



1. Authenticate (username/password)

2. Approve requested access

User
(Resource Owner)

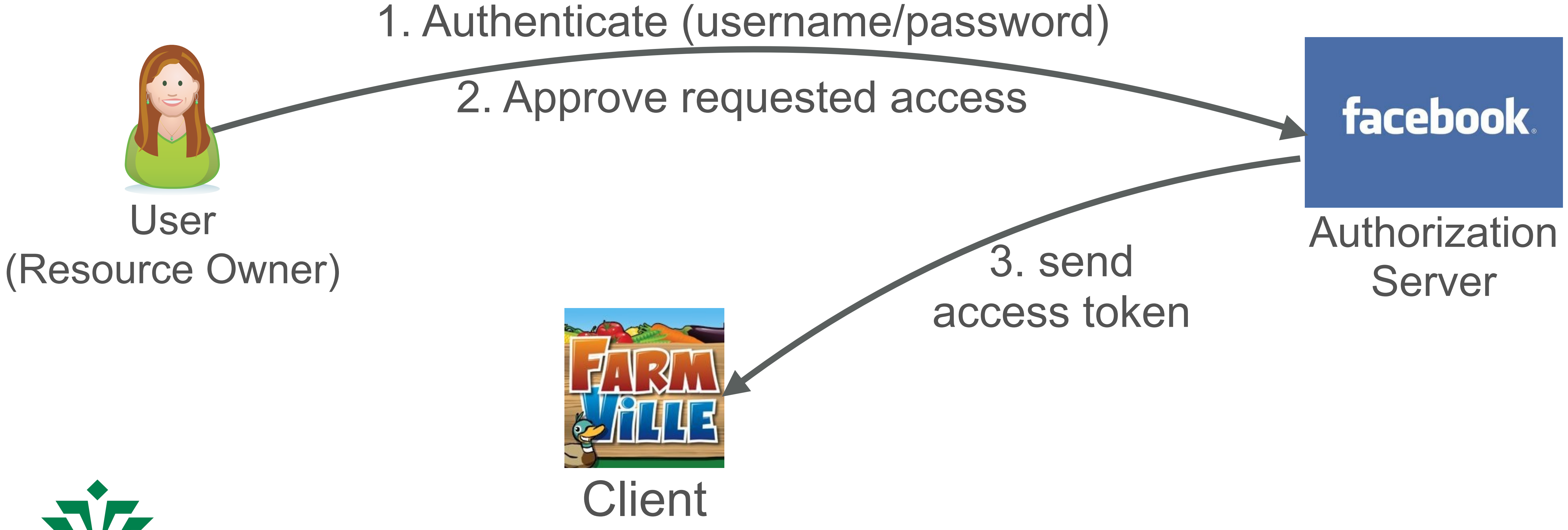
Authorization
Server

Client

Introduction (OAuth Flow)

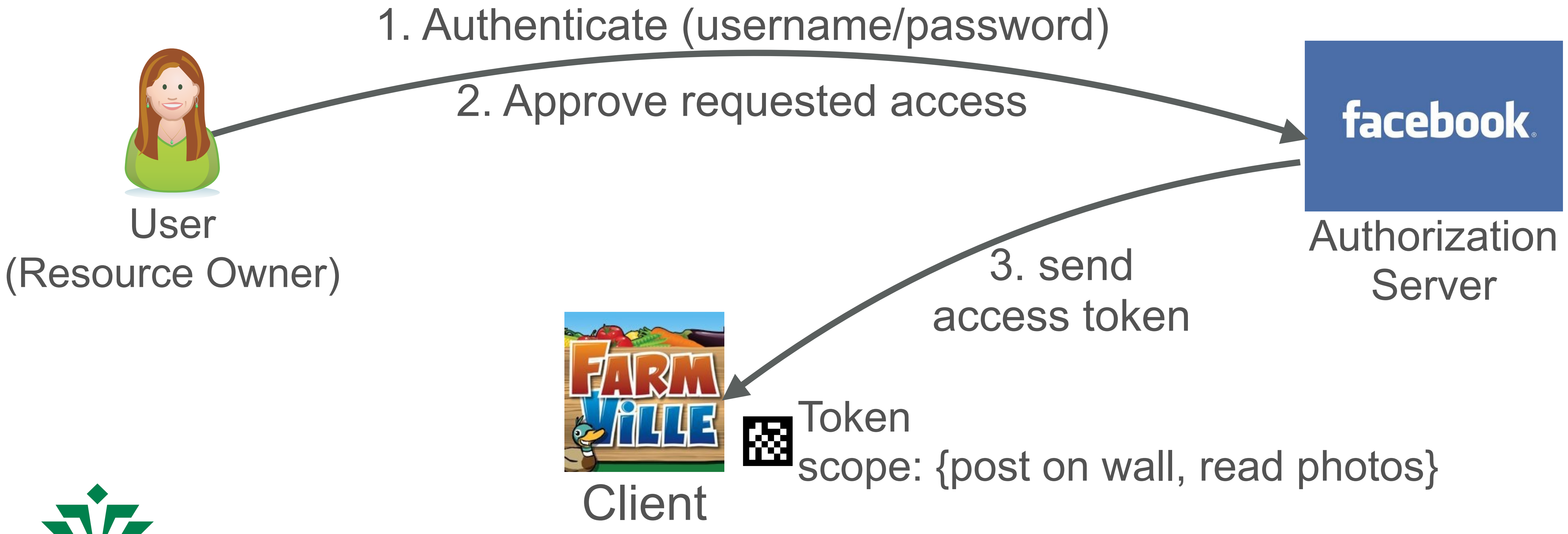
- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**

- **These access tokens have limited *scope*. Which limits the permissions granted to the third party app.**



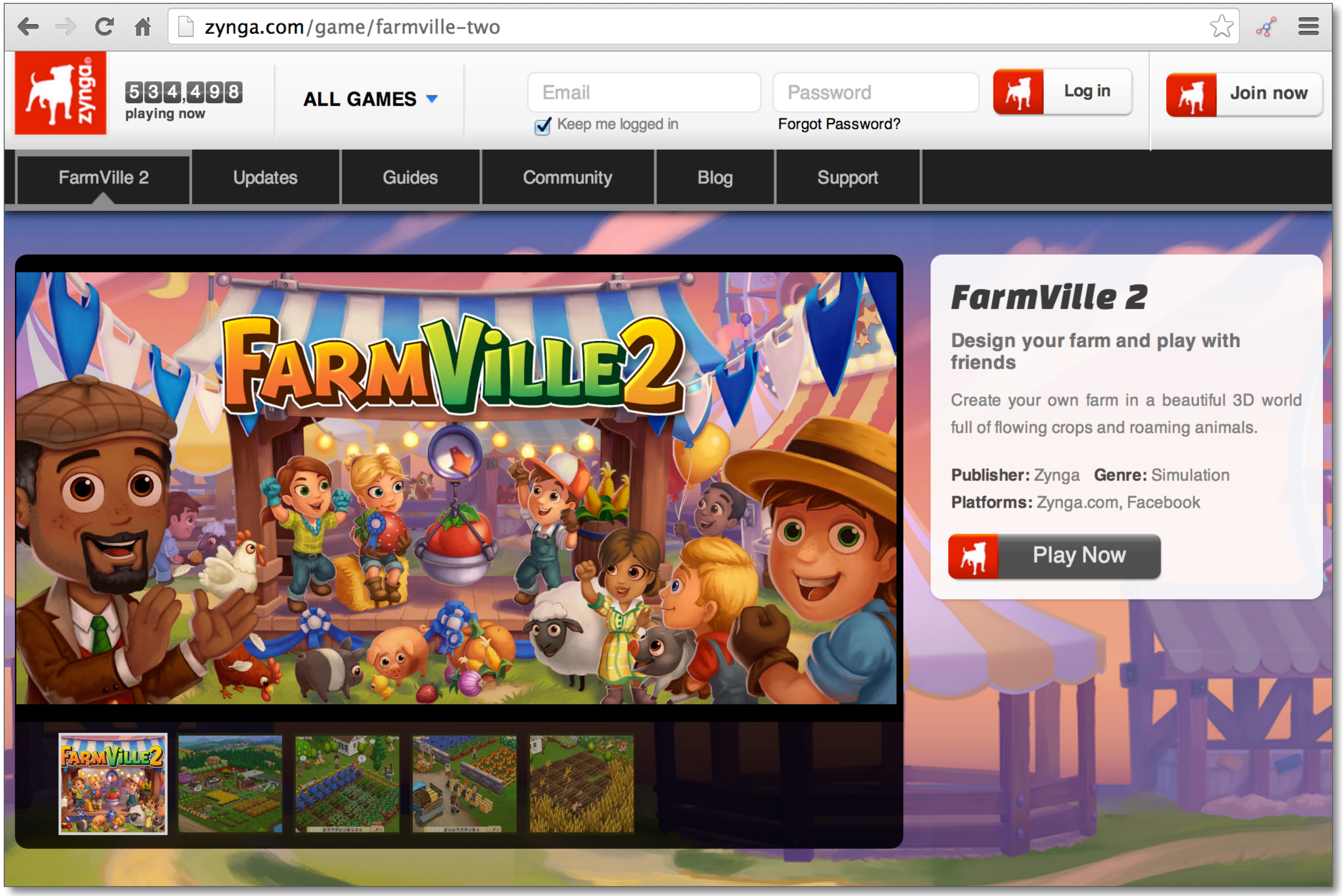
Introduction (OAuth Flow)

- **The OAuth framework enables the user to issue *access tokens* to the third party apps to make *api requests* on behalf of the user.**
- **These access tokens have limited *scope*. Which limits the permissions granted to the third party app.**


1. Authenticate (username/password)
2. Approve requested access
User
(Resource Owner)
facebook
Authorization
Server
3. send
access token
FARM
VILLE
Client
Token
scope: {post on wall, read photos}

Introduction (OAuth Flow)

- **User visits the client site.**



FarmVille 2

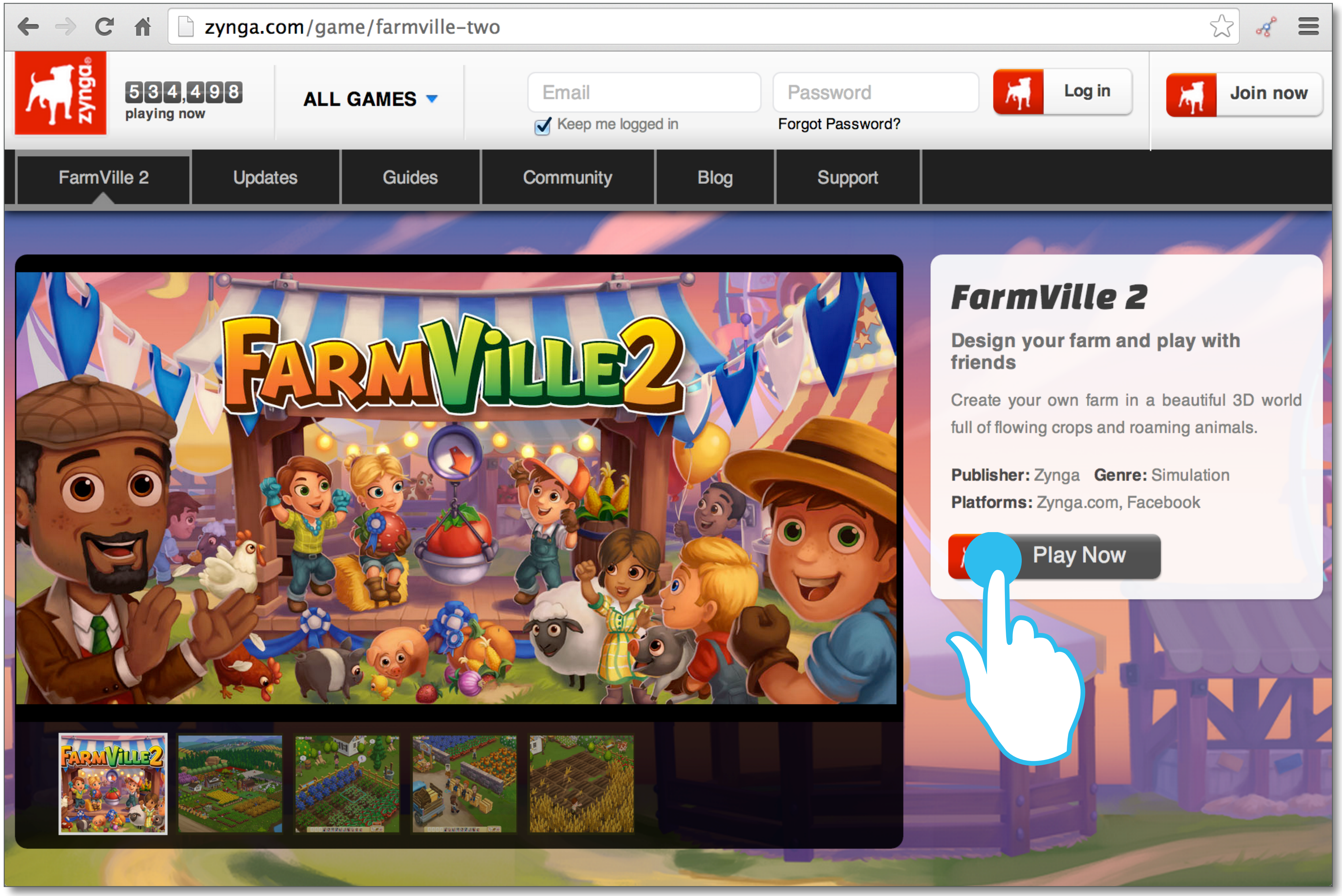
Design your farm and play with friends

Create your own farm in a beautiful 3D world full of flowing crops and roaming animals.

Publisher: Zynga **Genre:** Simulation
Platforms: Zynga.com, Facebook

Introduction (OAuth Flow)

- **User visits the client site.**



zynga.com/game/farmville-two

FarmVille 2

Design your farm and play with friends

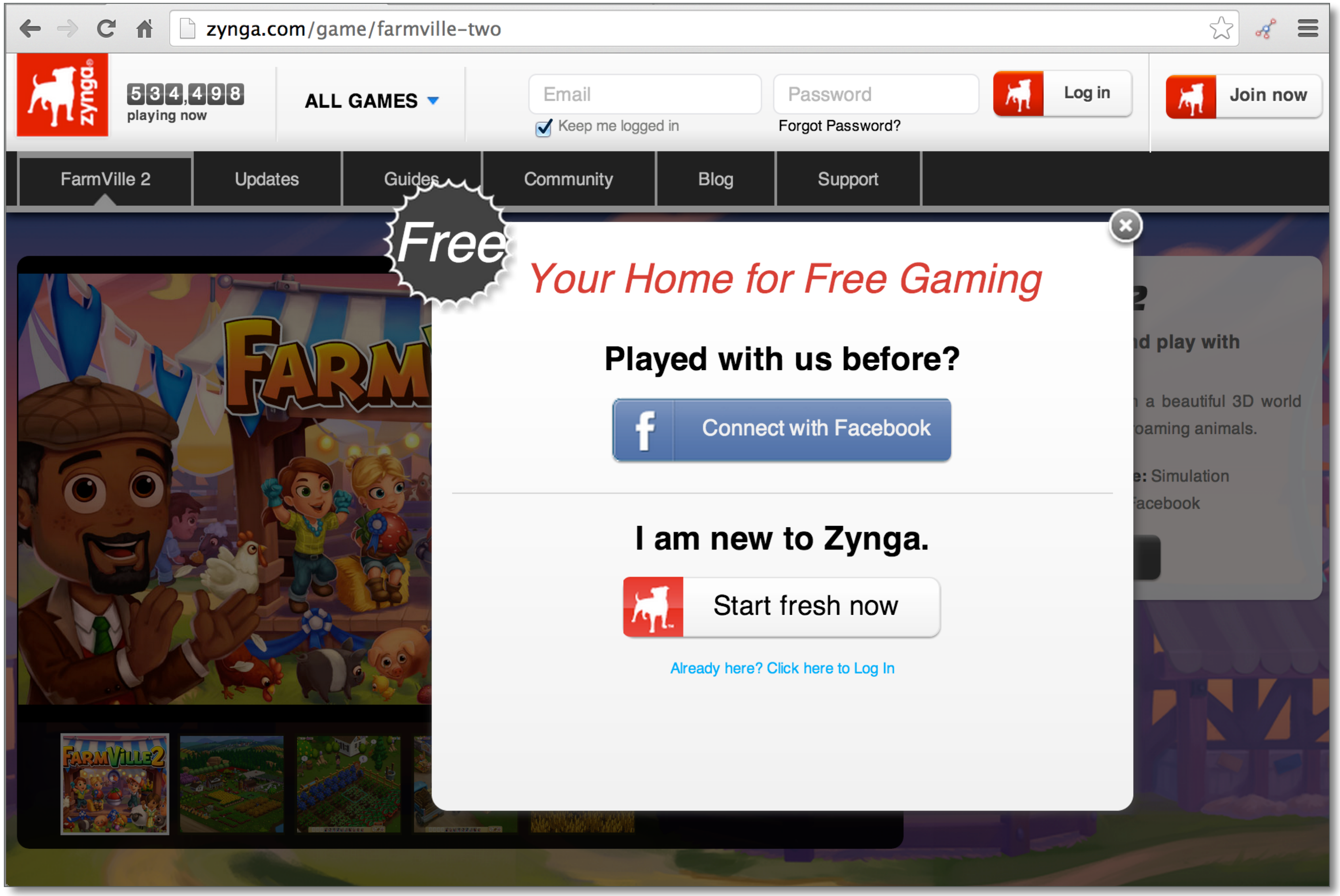
Create your own farm in a beautiful 3D world full of flowing crops and roaming animals.

Publisher: Zynga **Genre:** Simulation

Platforms: Zynga.com, Facebook

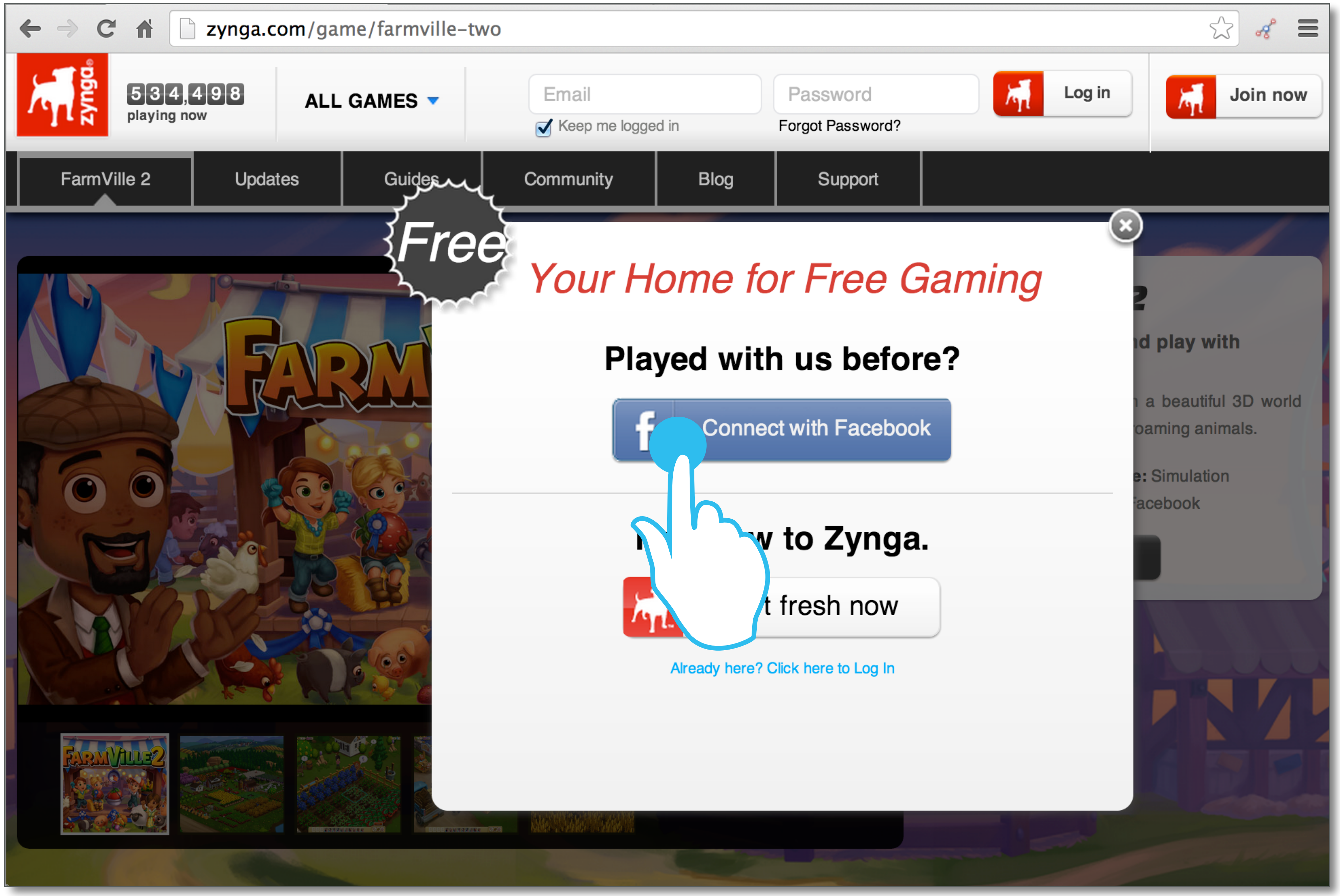
Introduction (OAuth Flow)

- **User asked to connect or login using Facebook**



Introduction (OAuth Flow)

- **User asked to connect or login using Facebook**



Your Home for Free Gaming

Played with us before?

Connect with Facebook

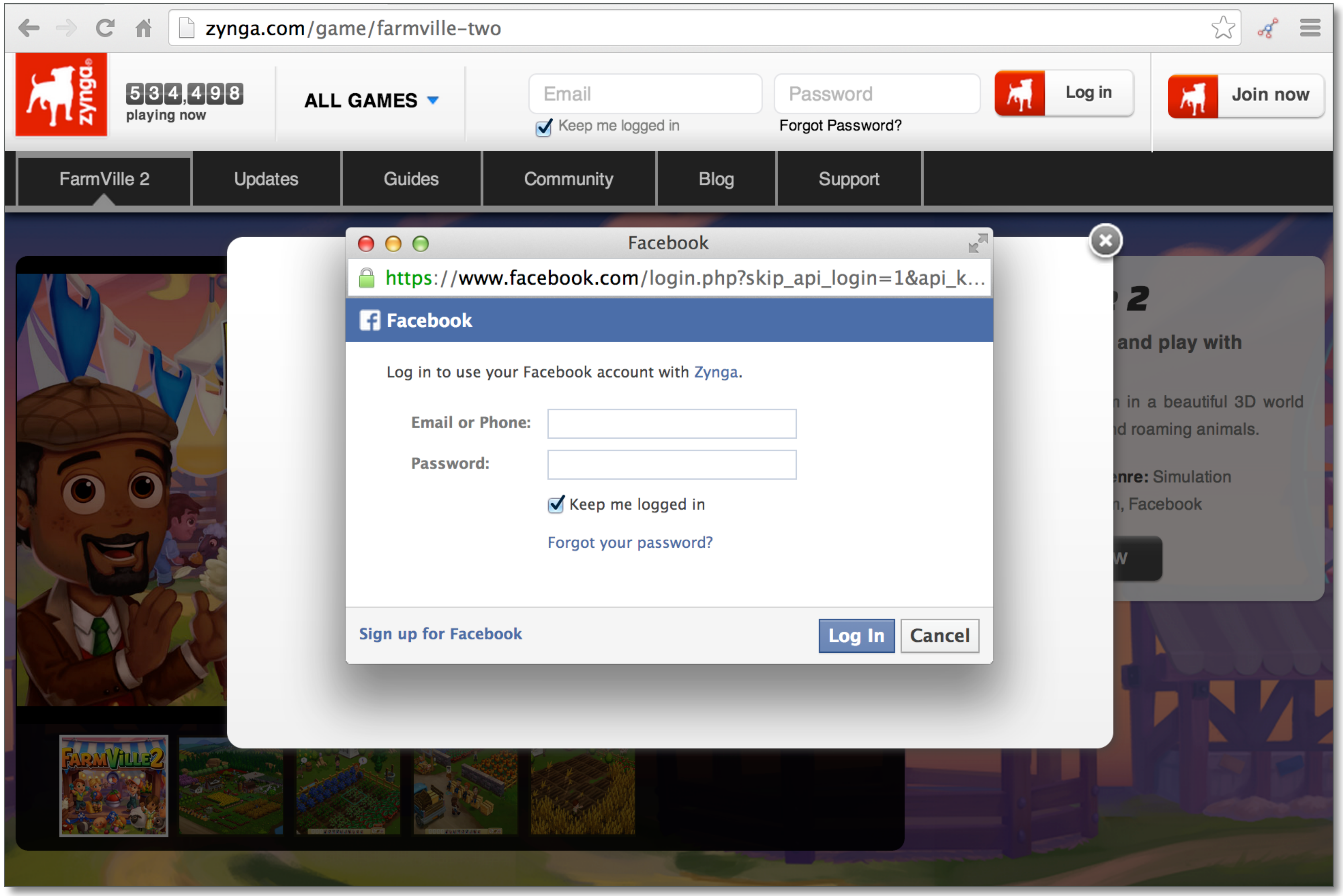
w to Zynga.

t fresh now

Already here? Click here to Log In

Introduction (OAuth Flow)

- **User asked to connect or login using Facebook**



zynga.com/game/farmville-two

534,498 playing now

ALL GAMES

Email

Password

☑ Keep me logged in

Forgot Password?

Log in

Join now

FarmVille 2 | Updates | Guides | Community | Blog | Support

Facebook

https://www.facebook.com/login.php?skip_api_login=1&api_k...

Facebook

Log in to use your Facebook account with Zynga.

Email or Phone:

Password:

☑ Keep me logged in

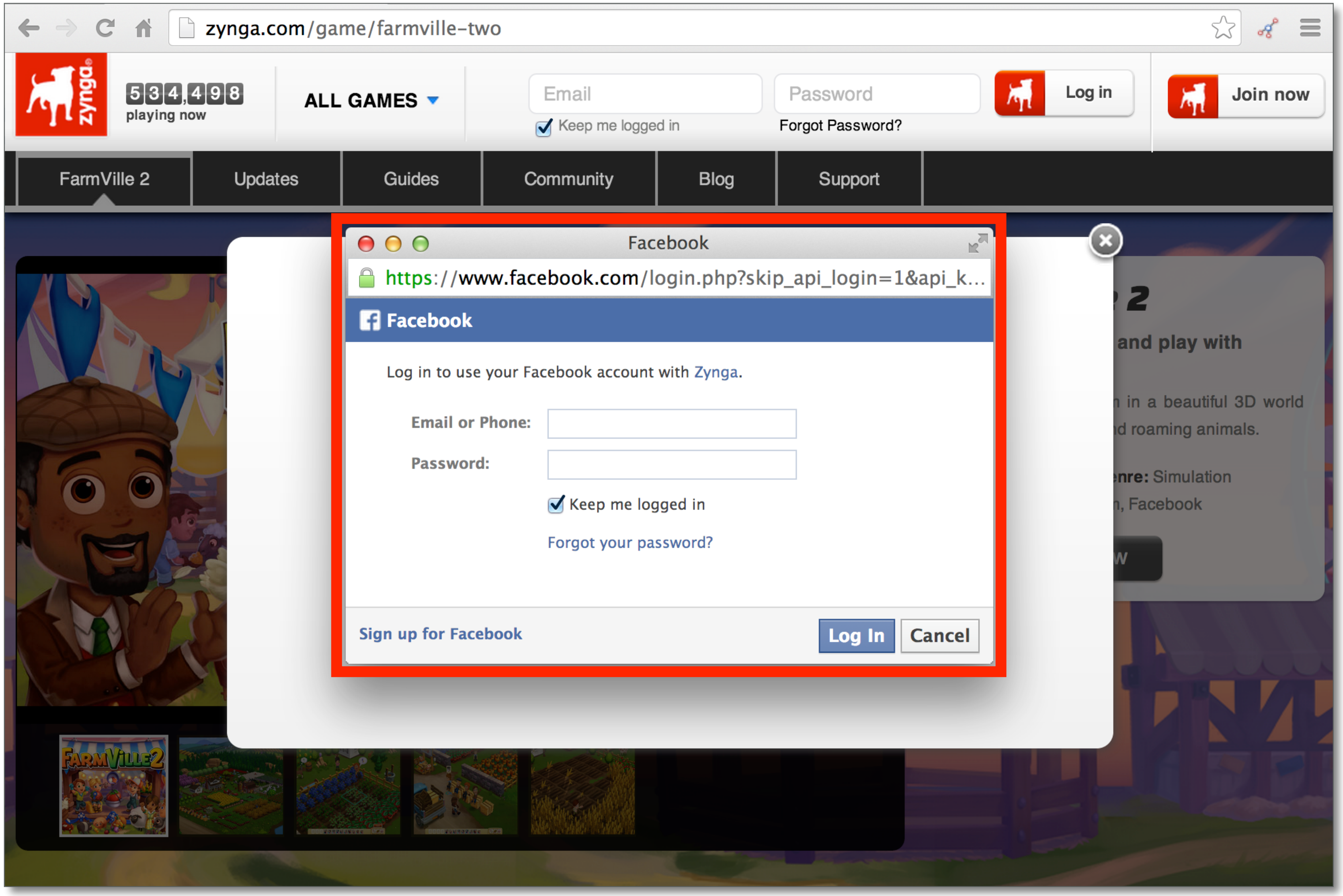
Forgot your password?

Sign up for Facebook

Log In | **Cancel**

Introduction (OAuth Flow)

- **User asked to connect or login using Facebook**



Log in to use your Facebook account with Zynga.

Email or Phone:

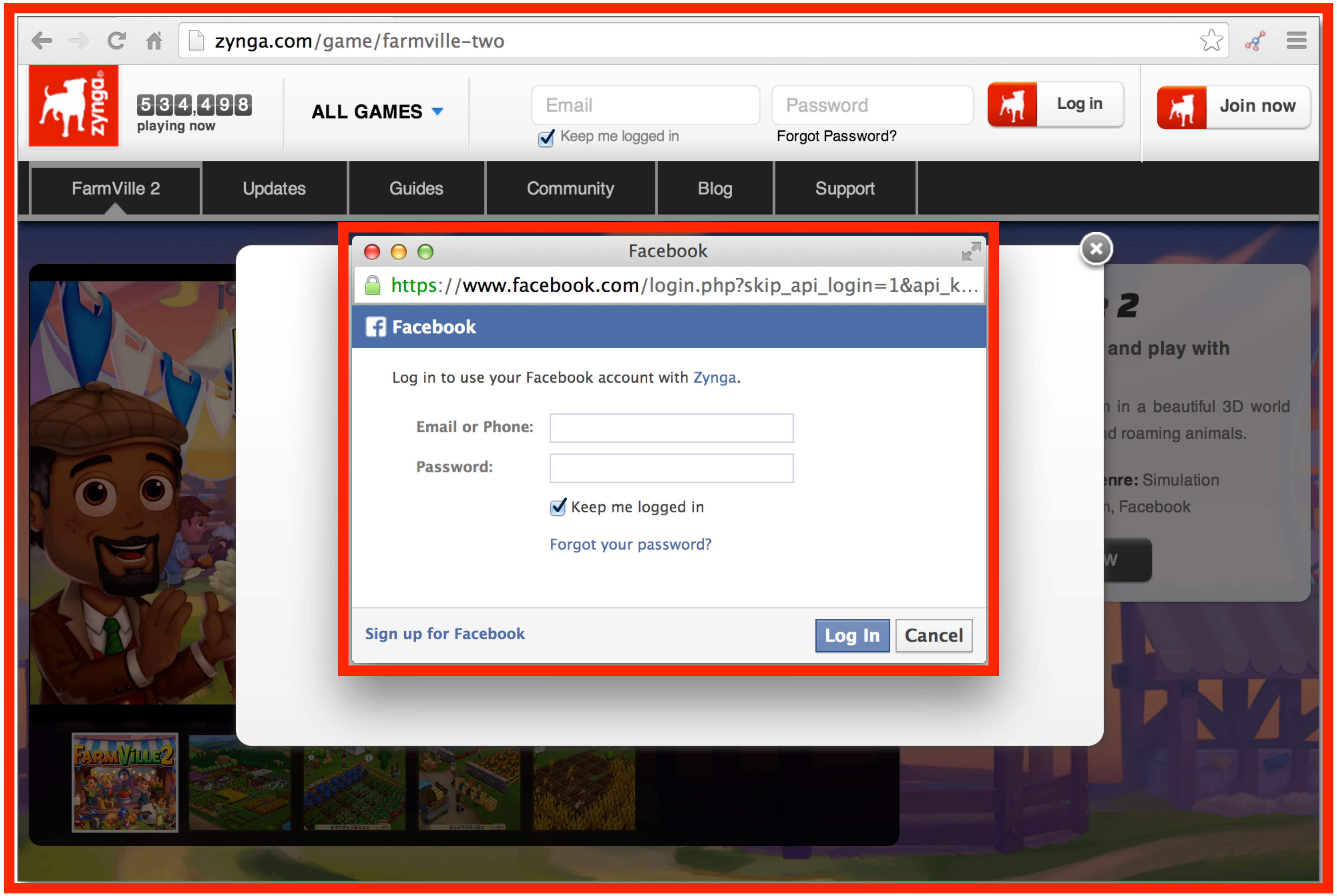
Password:

☑ Keep me logged in

Forgot your password?

Introduction (OAuth Flow)

- **User asked to connect or login using Facebook**



zynga.com/game/farmville-two

https://www.facebook.com/login.php?skip_api_login=1&api_k…

Log in to use your Facebook account with Zynga.

Email or Phone:

Password:

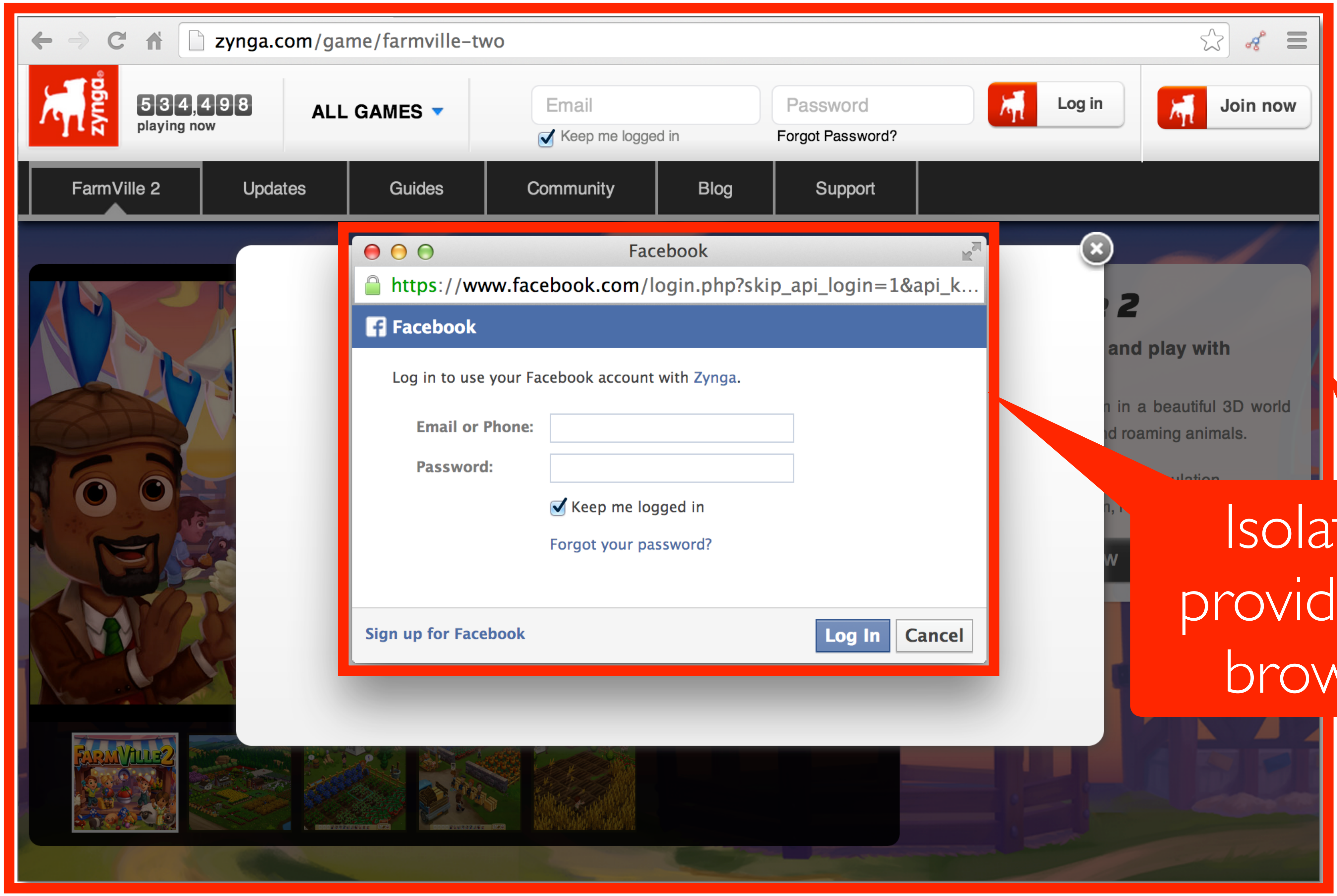
☑ Keep me logged in

Forgot your password?

Sign up for Facebook

Introduction (OAuth Flow)

- **User asked to connect or login using Facebook**



zynga.com/game/farmville-two

https://www.facebook.com/login.php?skip_api_login=1&api_k…

Log in to use your Facebook account with Zynga.

Email or Phone:

Password:

☑ Keep me logged in

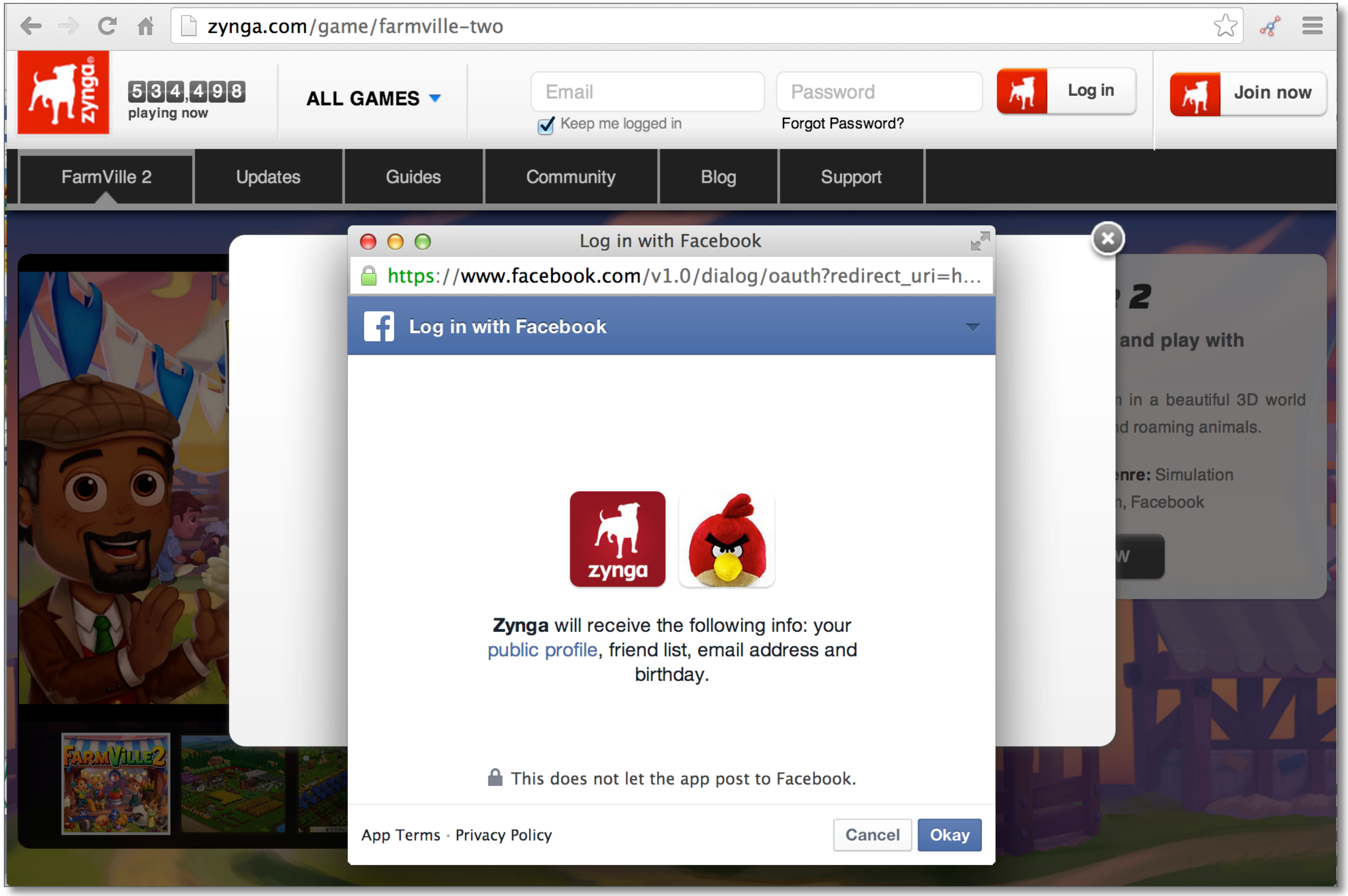
Forgot your password?

Sign up for Facebook

Isolation provided by browser

Introduction (OAuth Flow)

- **User asked to grant authorization (scope).**

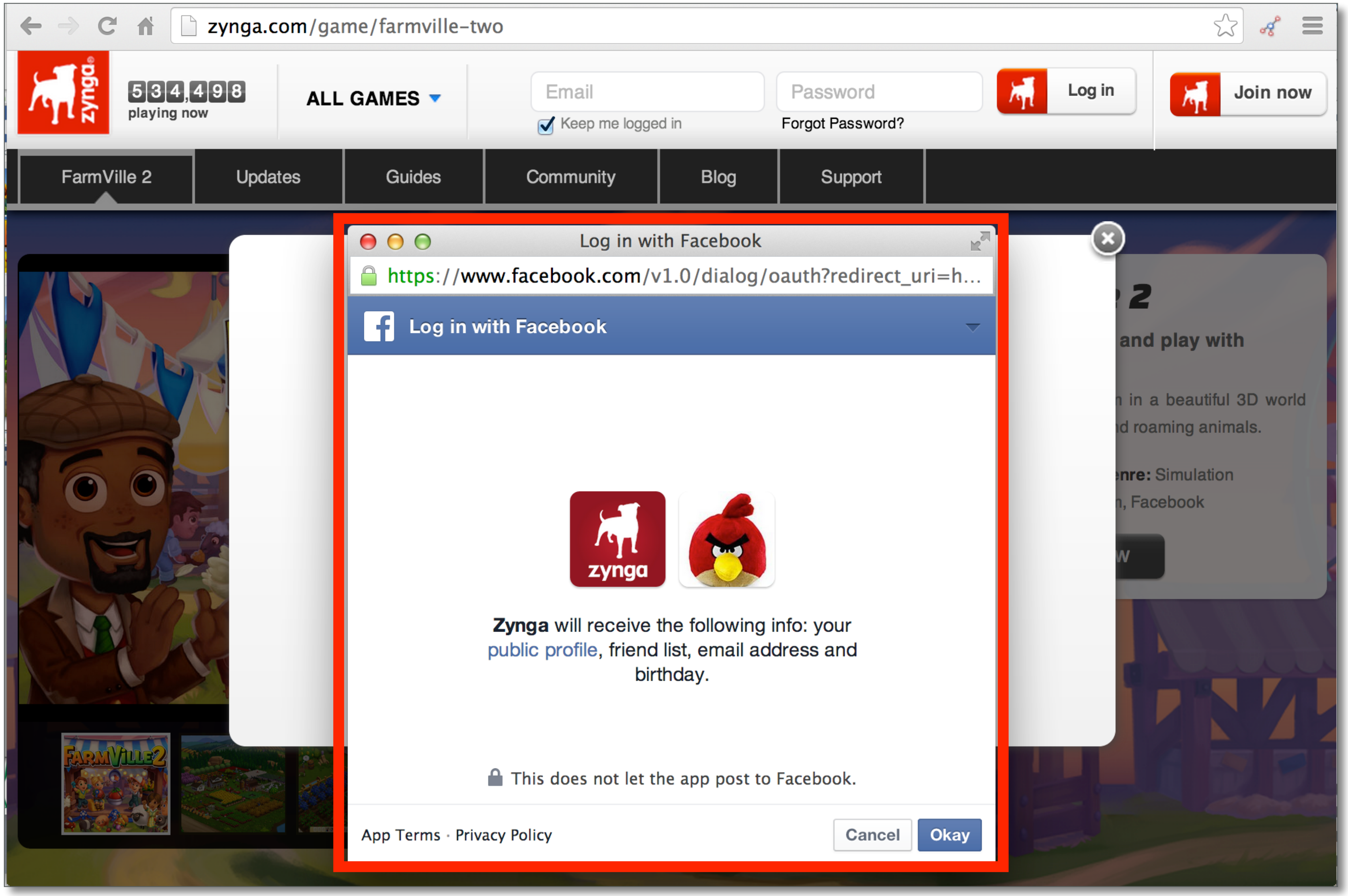


Zynga will receive the following info: your public profile, friend list, email address and birthday.

This does not let the app post to Facebook.

Introduction (OAuth Flow)

- **User asked to grant authorization (scope).**

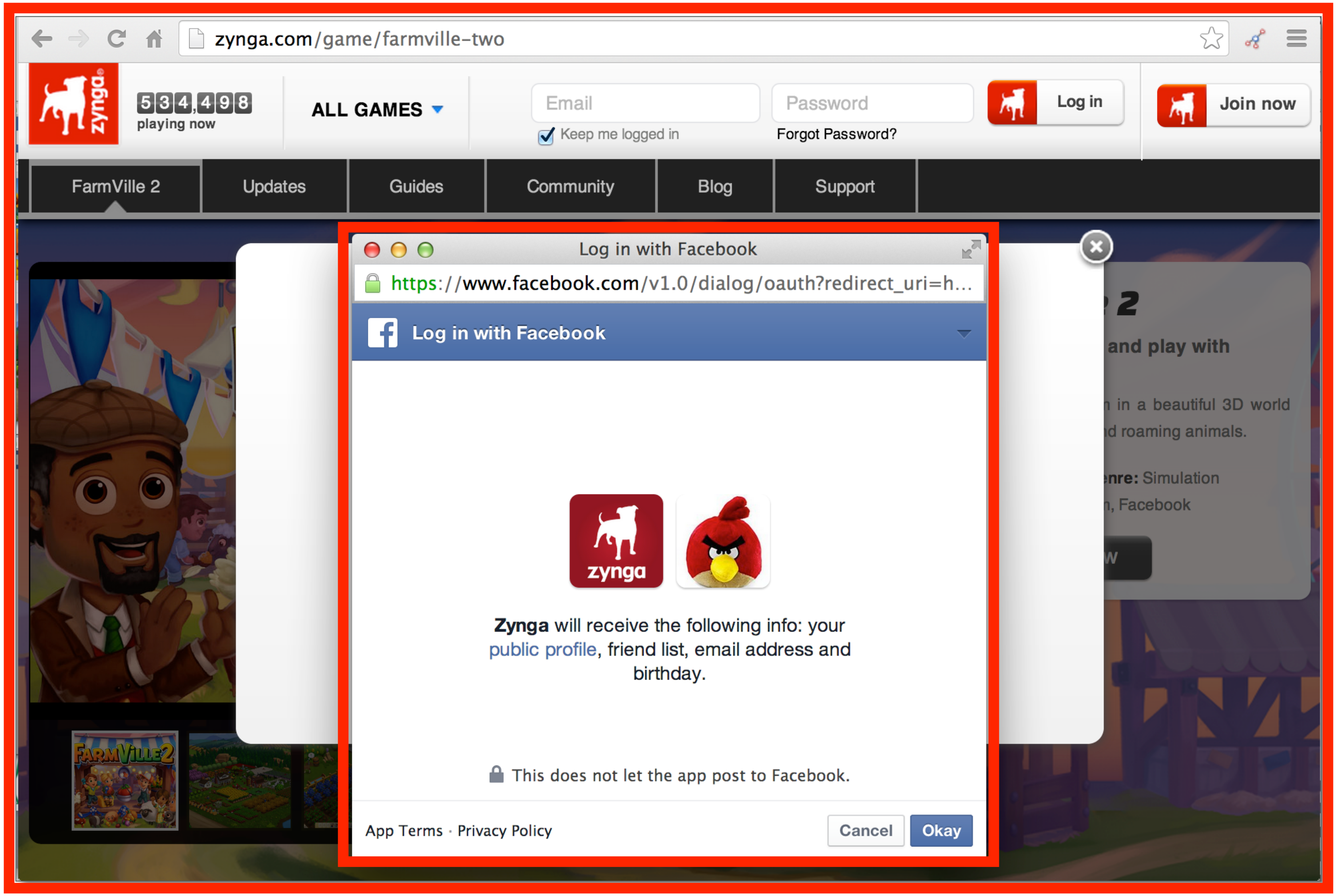


Zynga will receive the following info: your public profile, friend list, email address and birthday.

This does not let the app post to Facebook.

Introduction (OAuth Flow)

- **User asked to grant authorization (scope).**

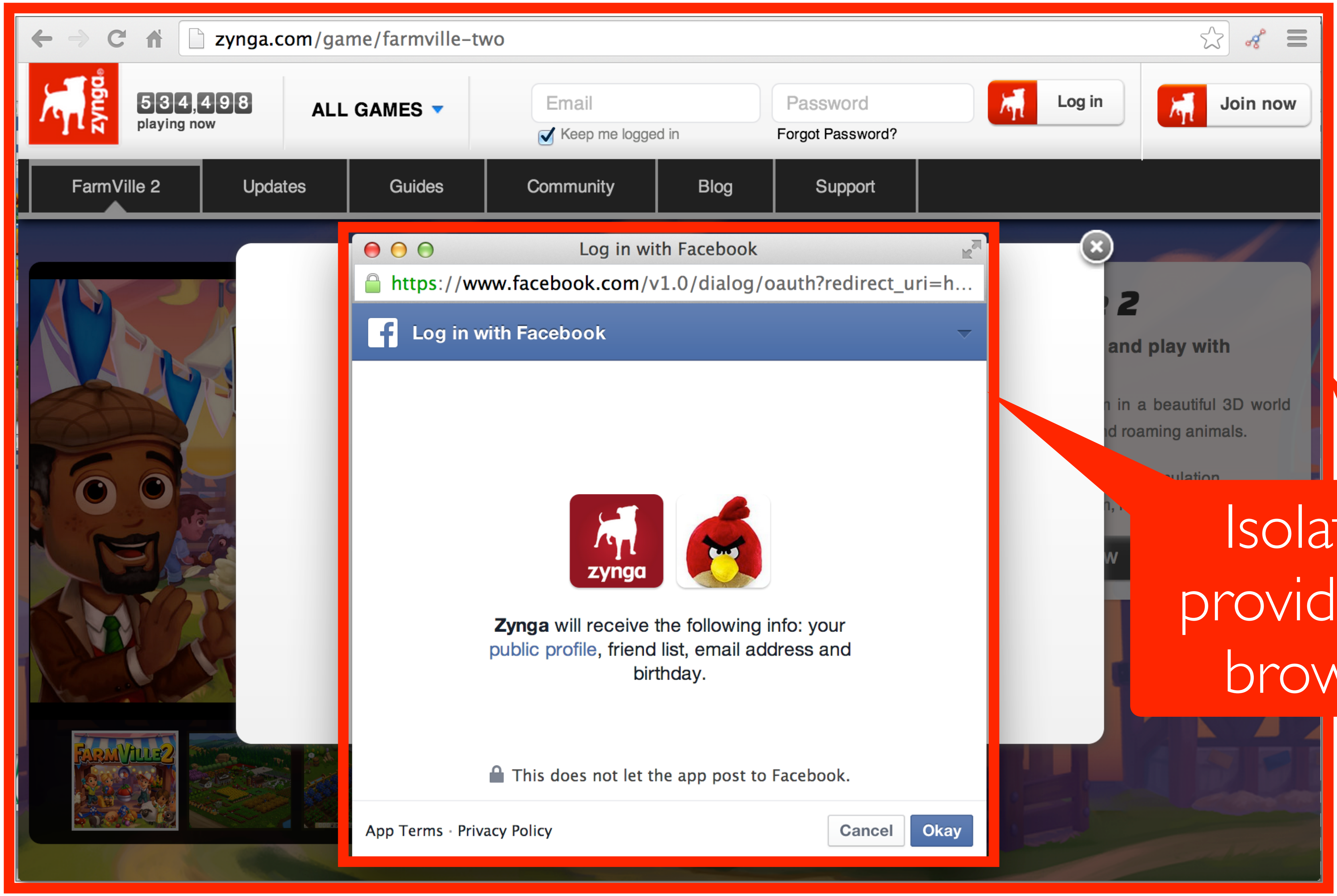


Zynga will receive the following info: your public profile, friend list, email address and birthday.

This does not let the app post to Facebook.

Introduction (OAuth Flow)

- **User asked to grant authorization (scope).**



zynga.com/game/farmville-two

Log in with Facebook

https://www.facebook.com/v1.0/dialog/oauth?redirect_uri=h...

Log in with Facebook

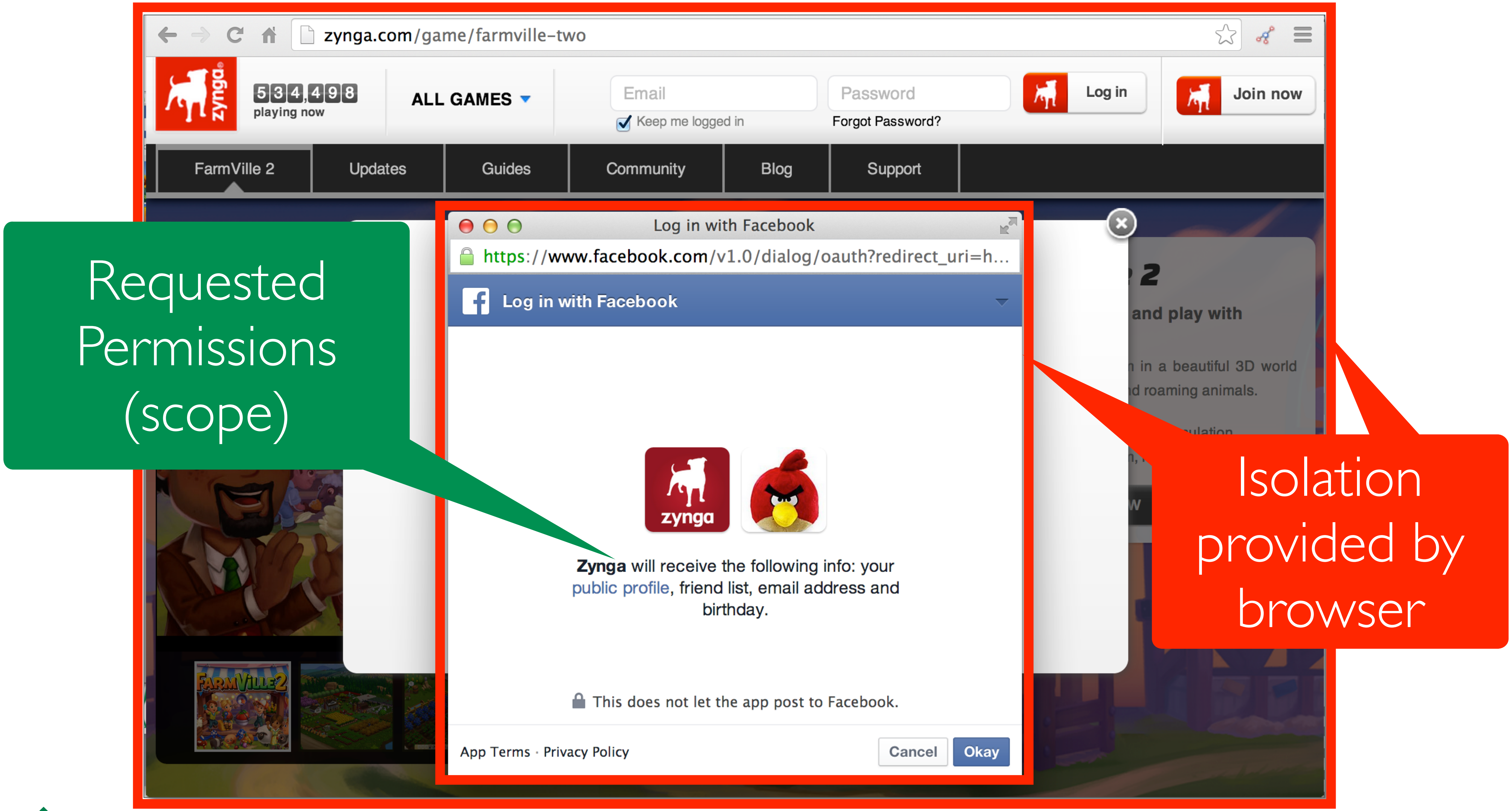
Zynga will receive the following info: your public profile, friend list, email address and birthday.

This does not let the app post to Facebook.

Isolation provided by browser

Introduction (OAuth Flow)

- **User asked to grant authorization (scope).**



zynga.com/game/farmville-two

https://www.facebook.com/v1.0/dialog/oauth?redirect_uri=h…

Zynga will receive the following info: your public profile, friend list, email address and birthday.

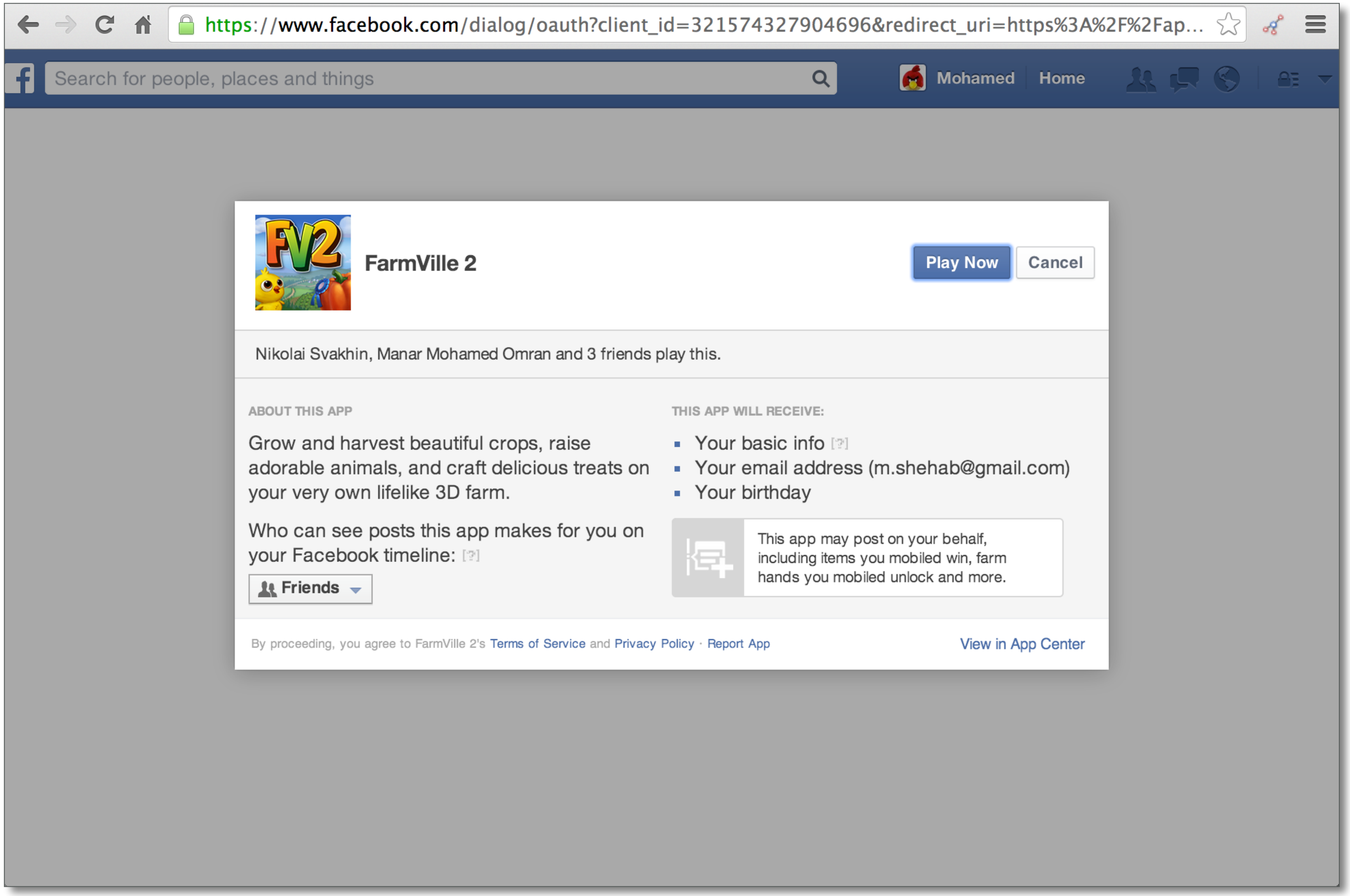
This does not let the app post to Facebook.

Requested Permissions (scope)

Isolation provided by browser

Introduction (OAuth Flow)

- **User asked to grant authorization (scope).**



https://www.facebook.com/dialog/oauth?client_id=321574327904696&redirect_uri=https%3A%2F%2Fap…

FarmVille 2

Nikolai Svakhin, Manar Mohamed Omran and 3 friends play this.

ABOUT THIS APP

Grow and harvest beautiful crops, raise adorable animals, and craft delicious treats on your very own lifelike 3D farm.

Who can see posts this app makes for you on your Facebook timeline: [?]

THIS APP WILL RECEIVE:

- Your basic info [?]
- Your email address (m.shehab@gmail.com)
- Your birthday

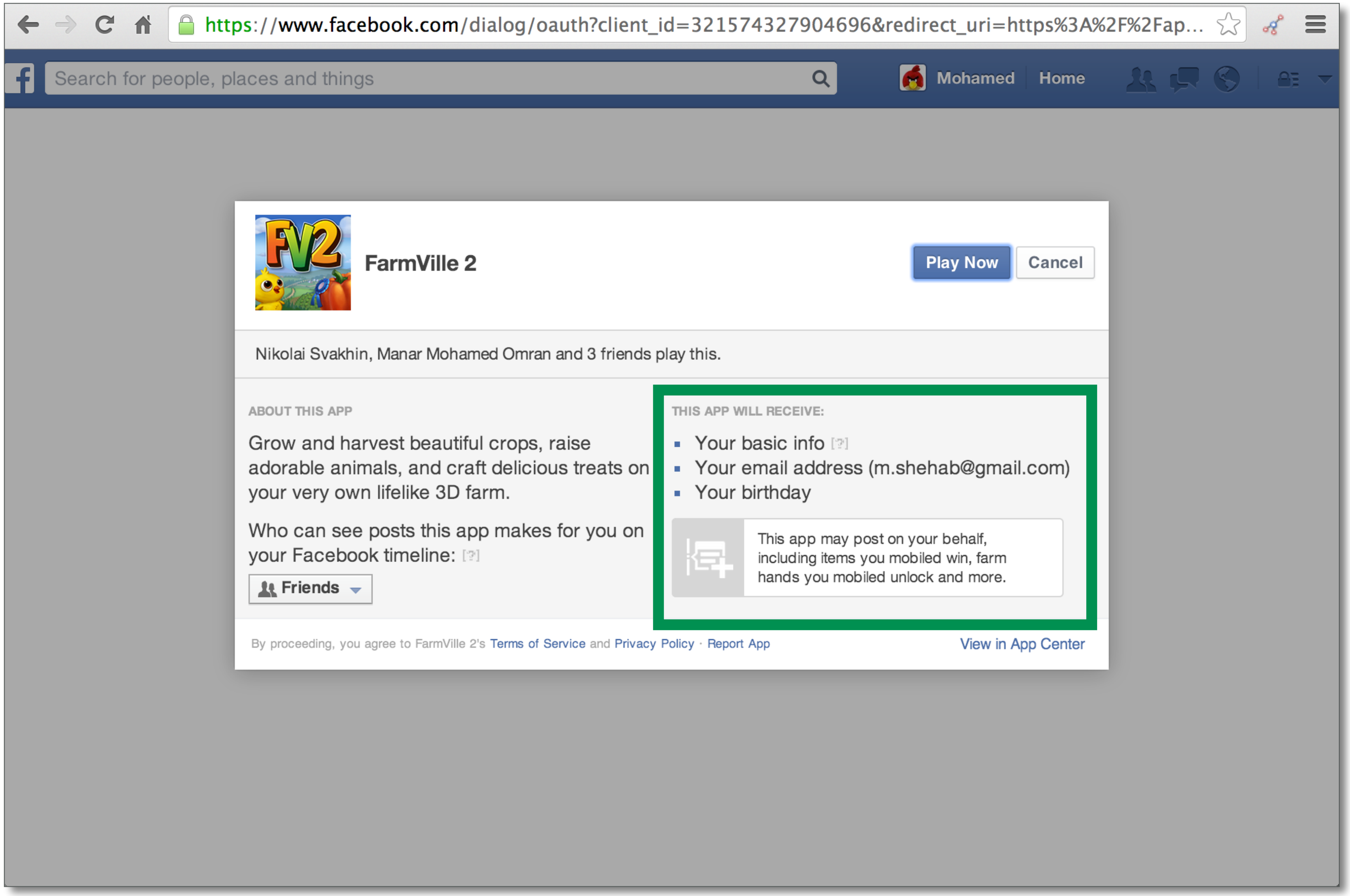
This app may post on your behalf, including items you mobiled win, farm hands you mobiled unlock and more.

By proceeding, you agree to FarmVille 2's Terms of Service and Privacy Policy · Report App

View in App Center

Introduction (OAuth Flow)

- **User asked to grant authorization (scope).**



https://www.facebook.com/dialog/oauth?client_id=321574327904696&redirect_uri=https%3A%2F%2Fap...

FarmVille 2

Nikolai Svakhin, Manar Mohamed Omran and 3 friends play this.

ABOUT THIS APP

Grow and harvest beautiful crops, raise adorable animals, and craft delicious treats on your very own lifelike 3D farm.

Who can see posts this app makes for you on your Facebook timeline: [?]

THIS APP WILL RECEIVE:

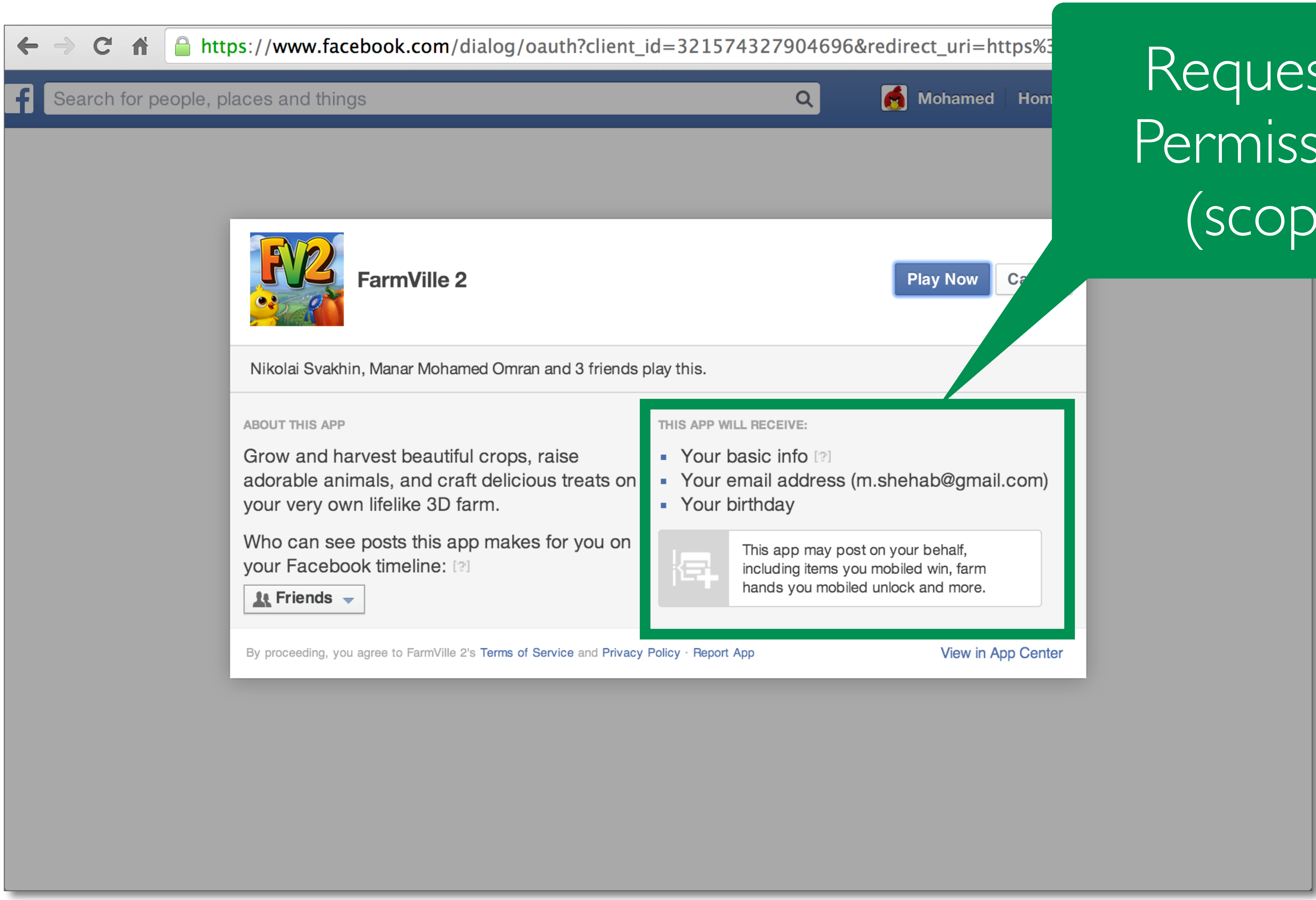
- Your basic info [?]
- Your email address (m.shehab@gmail.com)
- Your birthday

This app may post on your behalf, including items you mobiled win, farm hands you mobiled unlock and more.

By proceeding, you agree to FarmVille 2's Terms of Service and Privacy Policy · Report App

Introduction (OAuth Flow)

- **User asked to grant authorization (scope).**



https://www.facebook.com/dialog/oauth?client_id=321574327904696&redirect_uri=https%3

FarmVille 2

Nikolai Svakhin, Manar Mohamed Omran and 3 friends play this.

ABOUT THIS APP

Grow and harvest beautiful crops, raise adorable animals, and craft delicious treats on your very own lifelike 3D farm.

Who can see posts this app makes for you on your Facebook timeline: [?]

THIS APP WILL RECEIVE:

- Your basic info [?]
- Your email address (m.shehab@gmail.com)
- Your birthday

This app may post on your behalf, including items you mobiled win, farm hands you mobiled unlock and more.

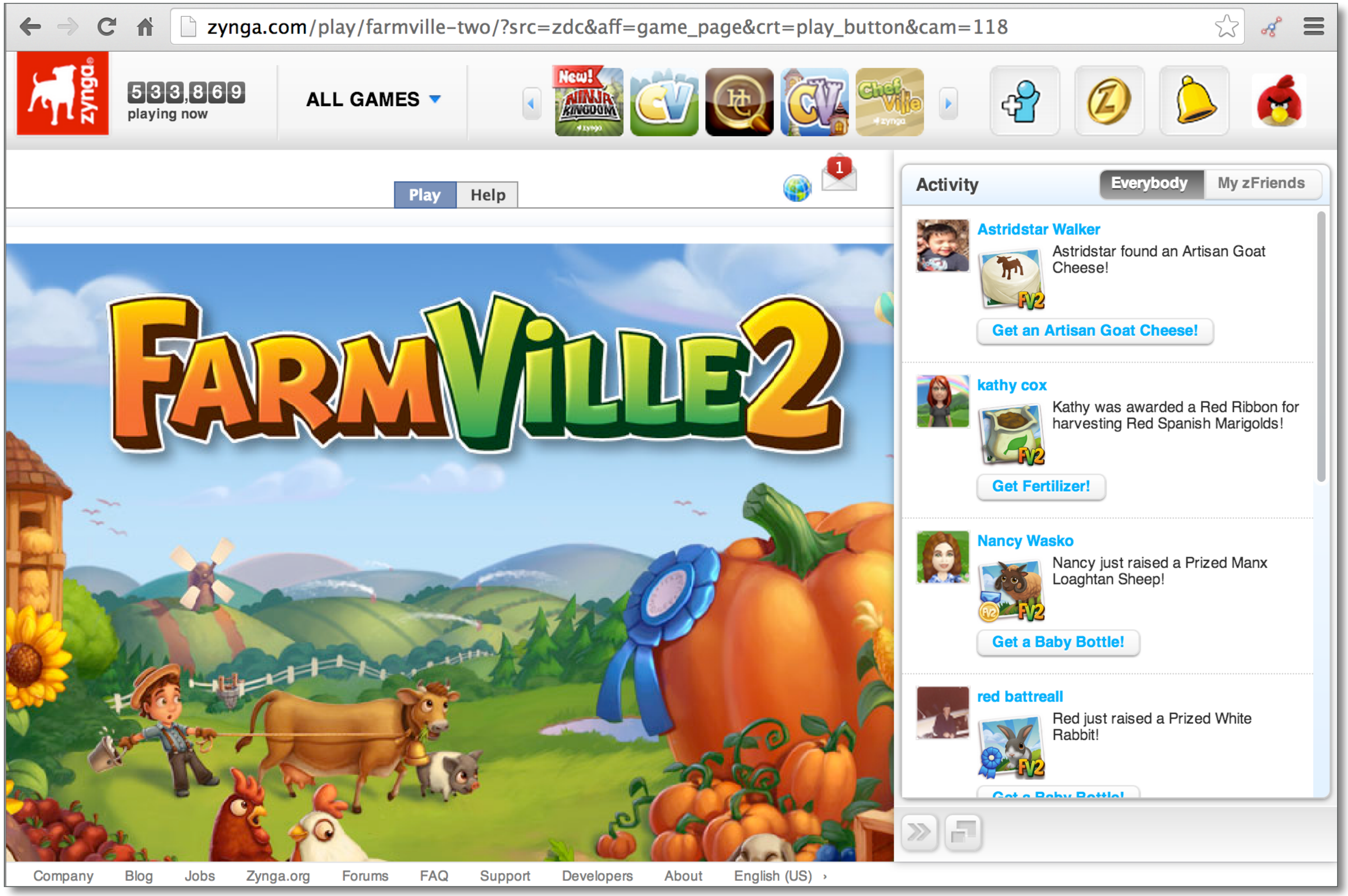
By proceeding, you agree to FarmVille 2's Terms of Service and Privacy Policy · Report App

View in App Center

Requested Permissions (scope)

Introduction (OAuth Flow)

- **App gains access to user resources.**



zynga.com/play/farmville-two/?src=zdc&aff=game_page&crt=play_button&cam=118

Activity

Astridstar Walker

Astridstar found an Artisan Goat Cheese!

Get an Artisan Goat Cheese!

kathy cox

Kathy was awarded a Red Ribbon for harvesting Red Spanish Marigolds!

Get Fertilizer!

Nancy Wasko

Nancy just raised a Prized Manx Loaghtan Sheep!

Get a Baby Bottle!

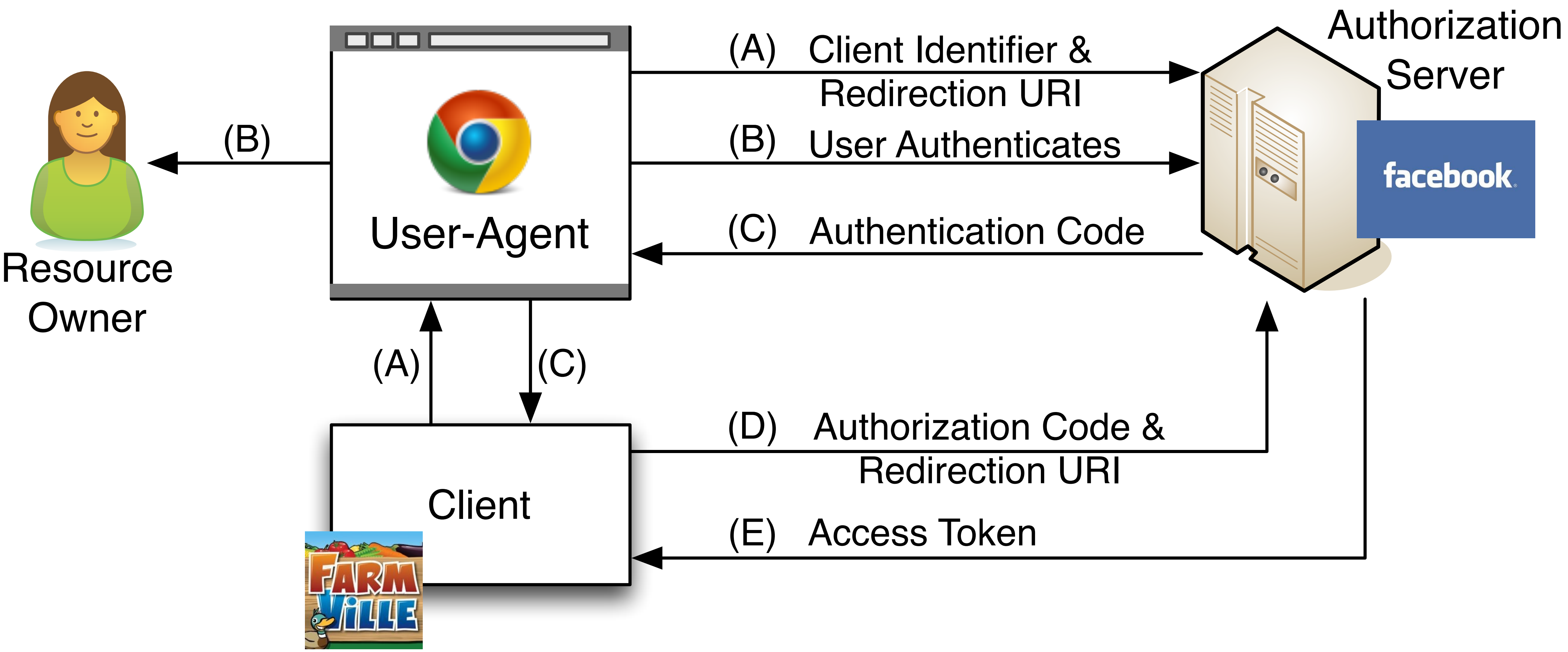
red battreall

Red just raised a Prized White Rabbit!

UNC CHARLOTTE

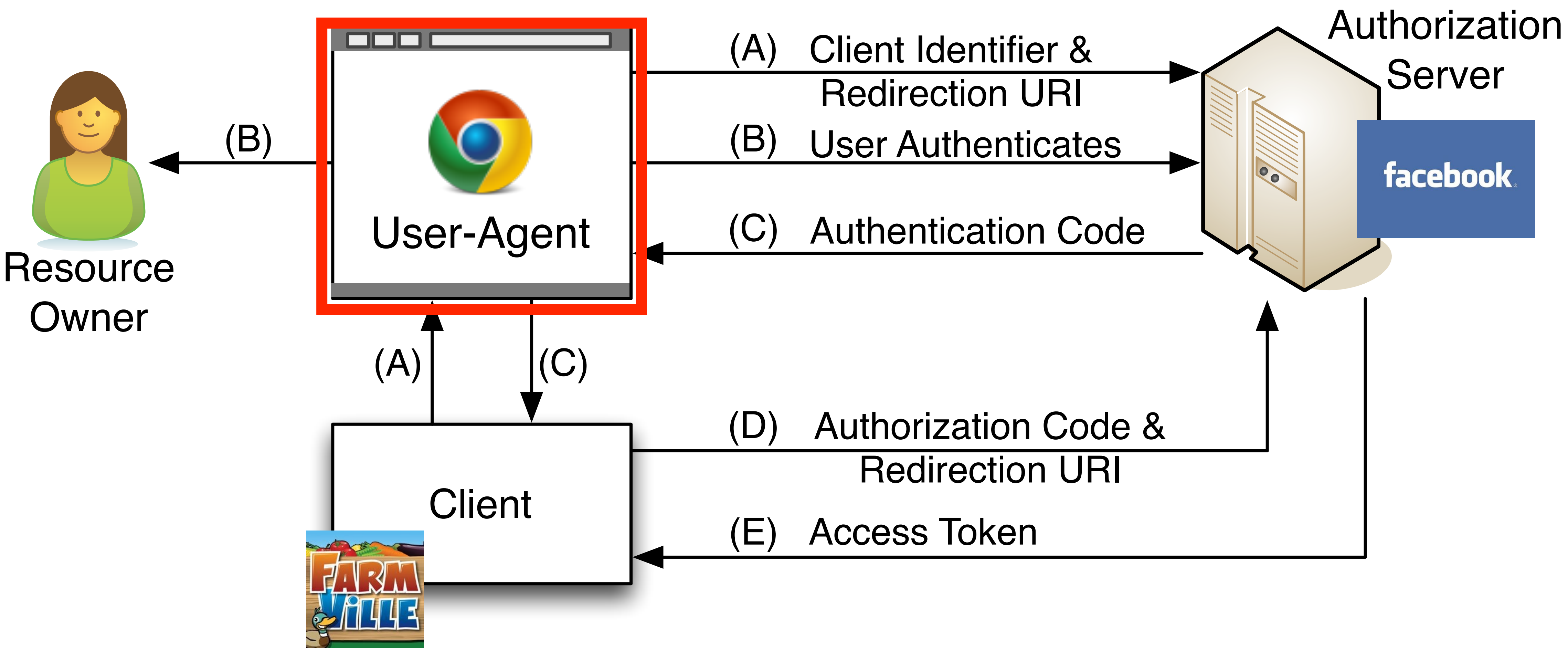
Web App OAuth Flow

- **The user agent is usually the browser**
  - The user interacts through the browser
  - The ***browser*** provides the required ***isolation*** between the ***client*** and the ***authorization server.***



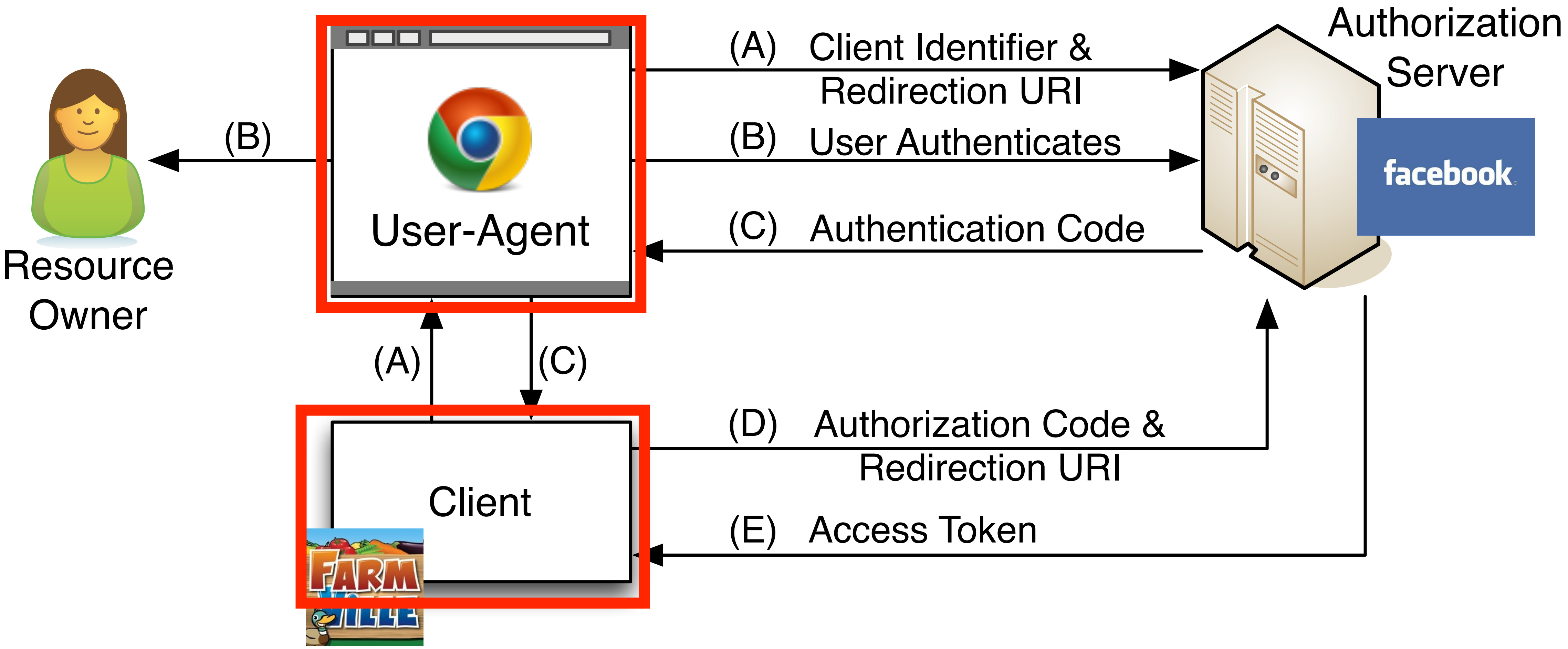
Web App OAuth Flow

- **The user agent is usually the browser**
  - The user interacts through the browser
  - The ***browser*** provides the required ***isolation*** between the ***client*** and the ***authorization server.***



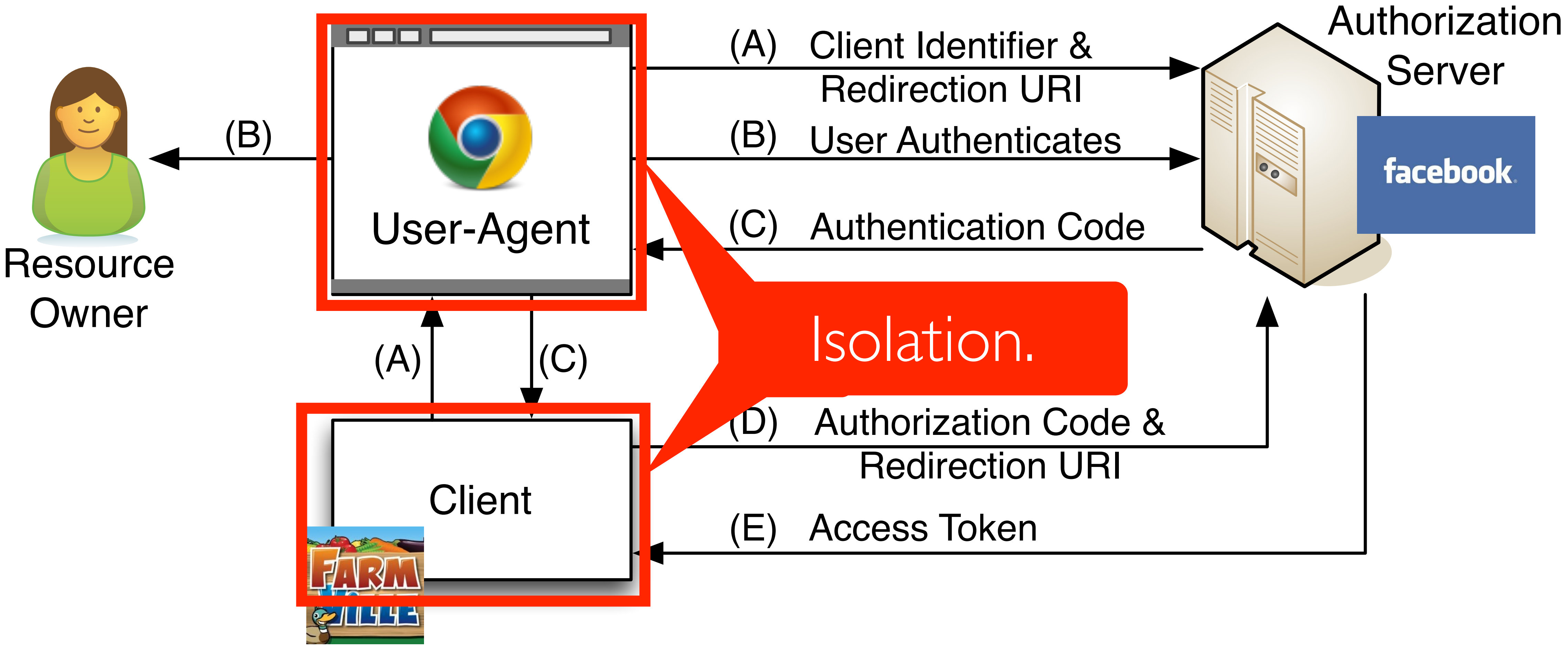
Web App OAuth Flow

- **The user agent is usually the browser**
  - The user interacts through the browser
  - The ***browser*** provides the required ***isolation*** between the ***client*** and the ***authorization server.***



Web App OAuth Flow

- **The user agent is usually the browser**
  - The user interacts through the browser
  - The ***browser*** provides the required ***isolation*** between the ***client*** and the ***authorization server.***



OAuth in Smart Phones

OAuth in Smart Phones

- **Several smart phone apps request access to user resources that are hosted in resource providers.**
  - For example, photo sharing application requesting access to user's Facebook photo albums.

OAuth in Smart Phones

- **Several smart phone apps request access to user resources that are hosted in resource providers.**
  - For example, photo sharing application requesting access to user's Facebook photo albums.

- **The main challenges in OAuth implementation in smart phone apps are:**
  - How to ***implement*** the ***user-agent***?
  - How to ***communicate*** the ***token*** from the ***user-agent*** to the ***app*** (client).

Our Contributions

Our Contributions

- **We identified the different OAuth implementations in smart phone frameworks, and summarized the vulnerabilities present in each of the implementations.**

Our Contributions

- **We identified the different OAuth implementations in smart phone frameworks, and summarized the vulnerabilities present in each of the implementations.**
- **We conducted an empirical study on the OAuth implementations in the SDKs offered by the popular resource providers, and by the app developers.**

Our Contributions

- **We identified the different OAuth implementations in smart phone frameworks, and summarized the vulnerabilities present in each of the implementations.**

- **We conducted an empirical study on the OAuth implementations in the SDKs offered by the popular resource providers, and by the app developers.**

- **We proposed a framework (*OAuth Manager*) that can provide protections against current OAuth vulnerabilities in smart phones.**

Our Contributions

- We identified the different OAuth implementations in smart phone frameworks, and summarized the vulnerabilities present in each of the implementations.

- We conducted an empirical study on the OAuth implementations in the SDKs offered by the popular resource providers, and by the app developers.

- We proposed a framework (*OAuth Manager*) that can provide protections against current OAuth vulnerabilities in smart phones.

- We compared our framework with other OAuth implementations in terms of performance and security.

OAuth in Smart Phones

- **There are three main approaches for implementing OAuth in smart phone apps**
  - Type 1: Through an ***Embedded Web Browser Component***.
  - Type 2: Using the ***Native Browser***.
  - Type 3: Using the ***Provider's Native App***.

Type 1: Embedded Web Browser Component

Type 1: Embedded Web Browser Component

- **The embedded web browser component is a UI component that can be embedded in a mobile app to render online content within the hosting app.**
  - WebView in Android.
  - UIWebView in iOS.
  - WebBrowser in Windows.

Type 1: Embedded Web Browser Component

- **The embedded web browser component is a UI component that can be embedded in a mobile app to render online content within the hosting app.**
  - WebView in Android.
  - UIWebView in iOS.
  - WebBrowser in Windows.

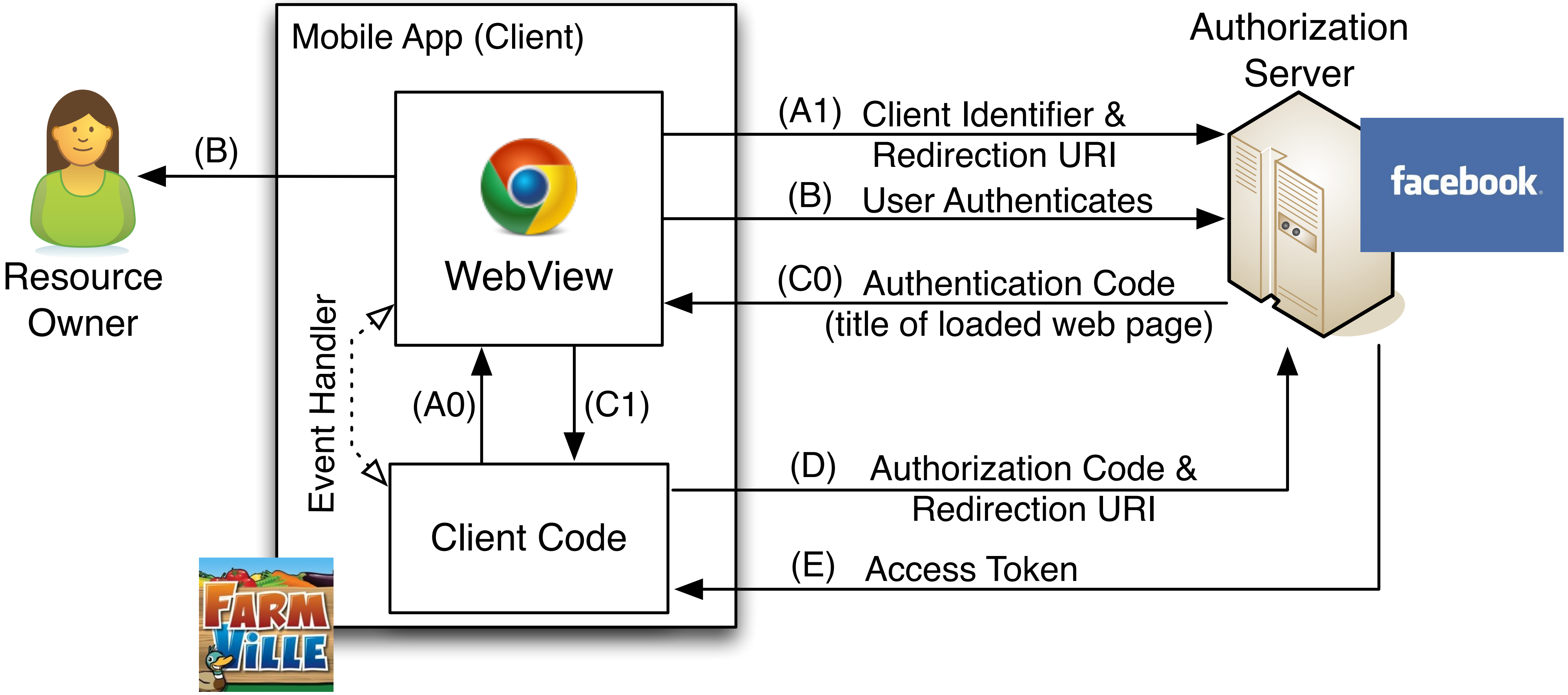
- **The embedded web browser component executes in the context of the hosting app and can be controlled, monitored and manipulated by the hosting app.**

Type 1: Embedded Web Browser Component

- **The embedded web browser component is used to:**
  - present the user with the authentication page (username/password).
  - present the user with the authorization page listing the required permissions.

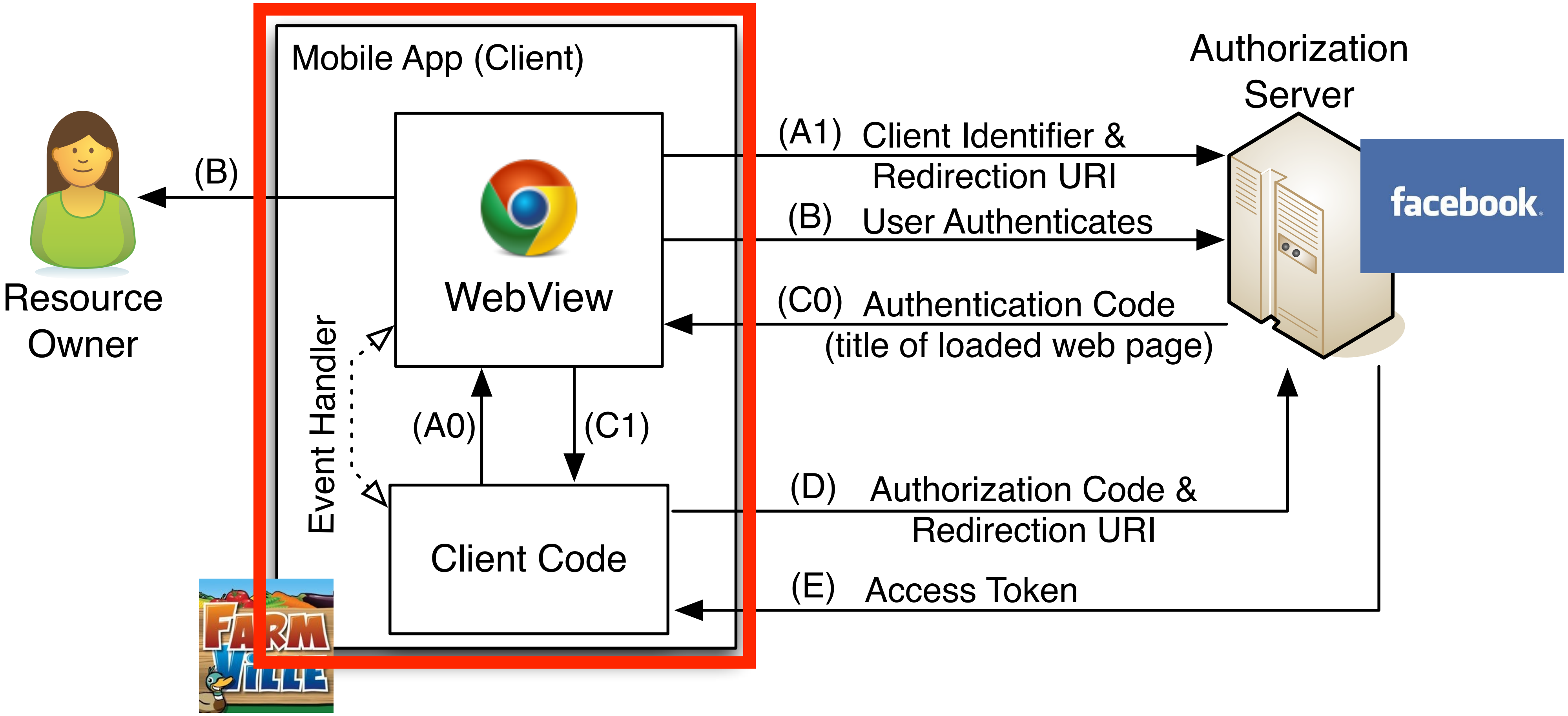
Type 1: Embedded Web Browser Component

- **The embedded web browser component is used to:**
  - present the user with the authentication page (username/password).
  - present the user with the authorization page listing the required permissions.



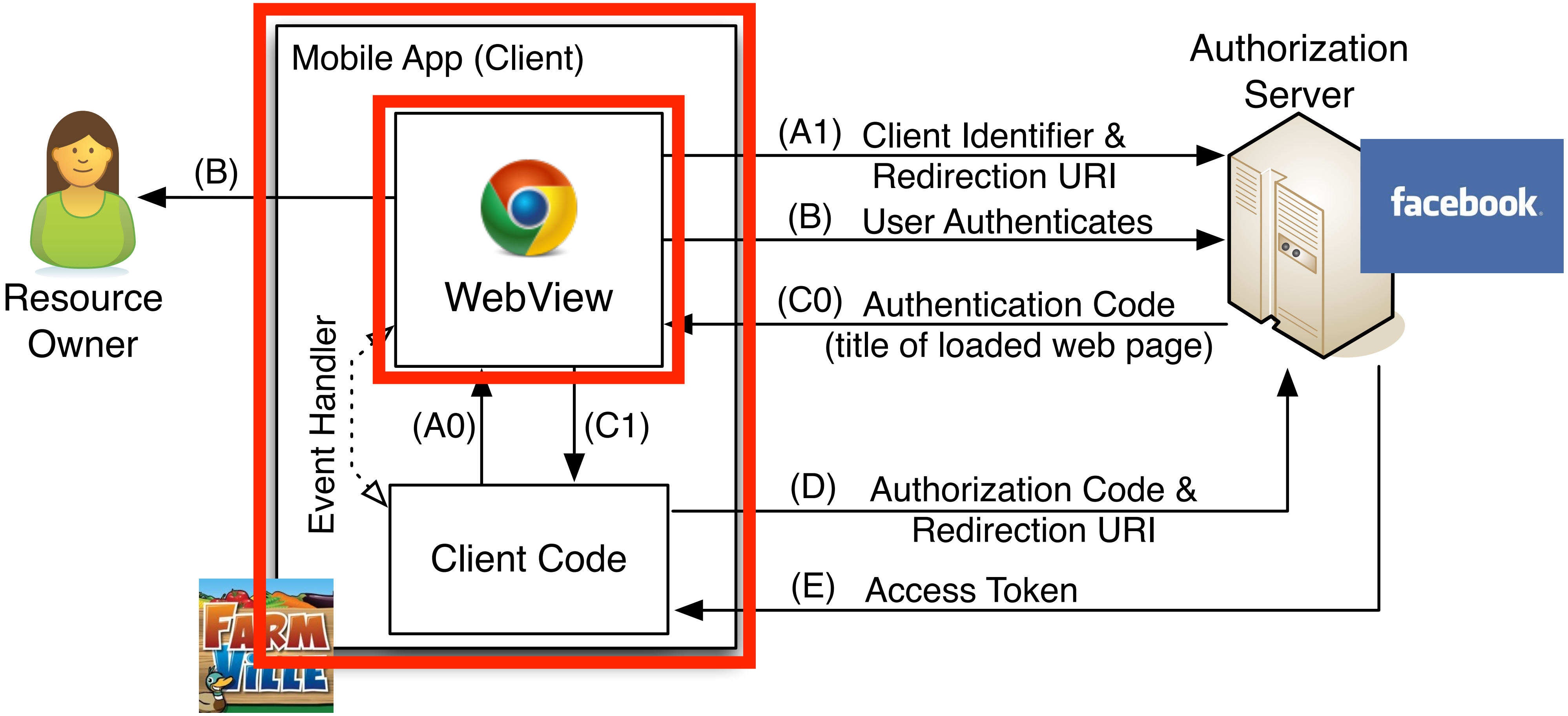
Type 1: Embedded Web Browser Component

- **The embedded web browser component is used to:**
  - present the user with the authentication page (username/password).
  - present the user with the authorization page listing the required permissions.



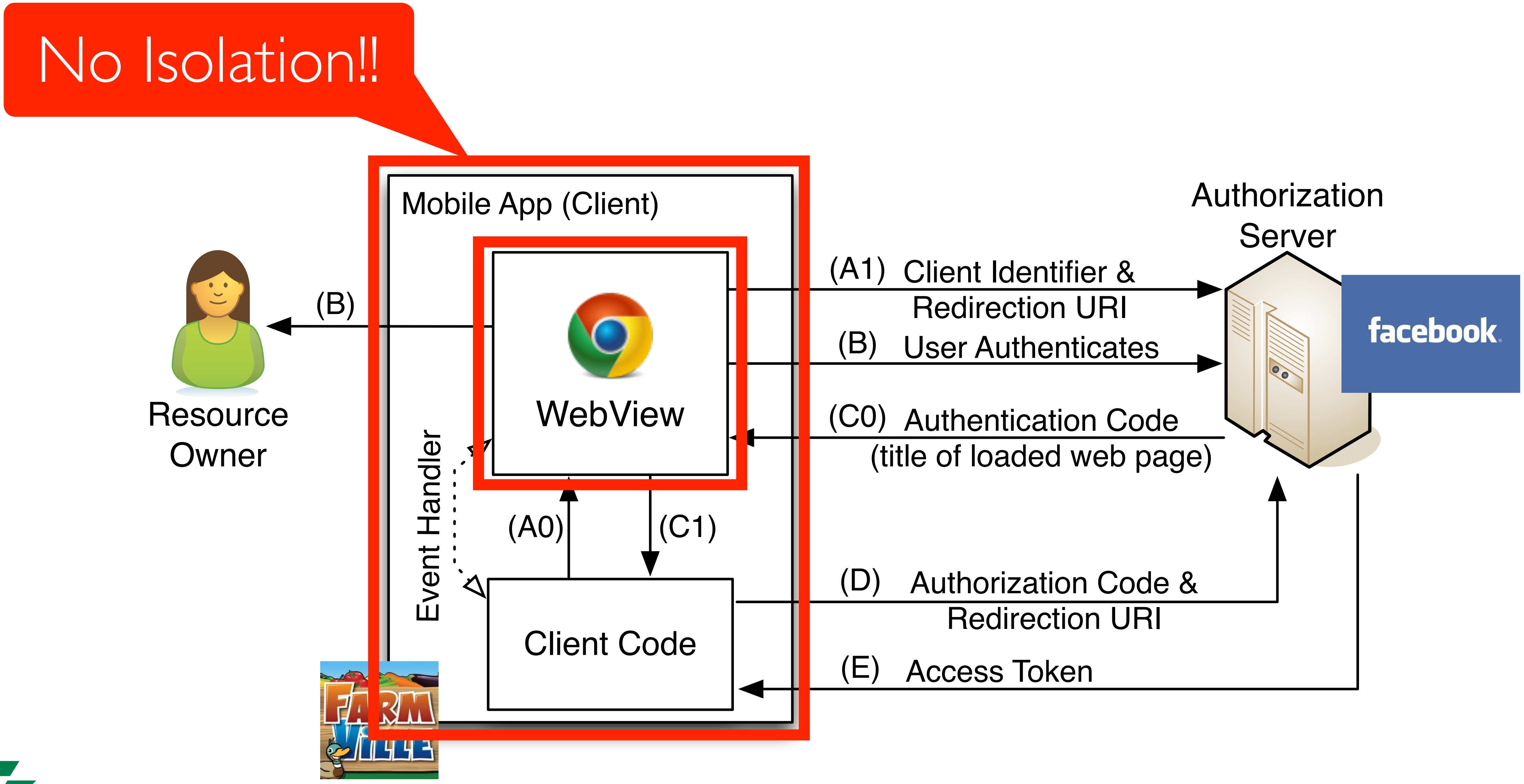
Type 1: Embedded Web Browser Component

- **The embedded web browser component is used to:**
  - present the user with the authentication page (username/password).
  - present the user with the authorization page listing the required permissions.



Type 1: Embedded Web Browser Component

- **The embedded web browser component is used to:**
  - present the user with the authentication page (username/password).
  - present the user with the authorization page listing the required permissions.



Type 1: Embedded Web Browser Component

Type 1: Embedded Web Browser Component

- **No isolation between the user-agent and the client (app). The client app is both the user-agent and the client app.**

Type 1: Embedded Web Browser Component

- **No isolation between the user-agent and the client (app). The client app is both the user-agent and the client app.**
- **A malicious hosting app can take control of the hosted web browser component and launch attacks on both the user authentication and application authorization pages:**

Type 1: Embedded Web Browser Component

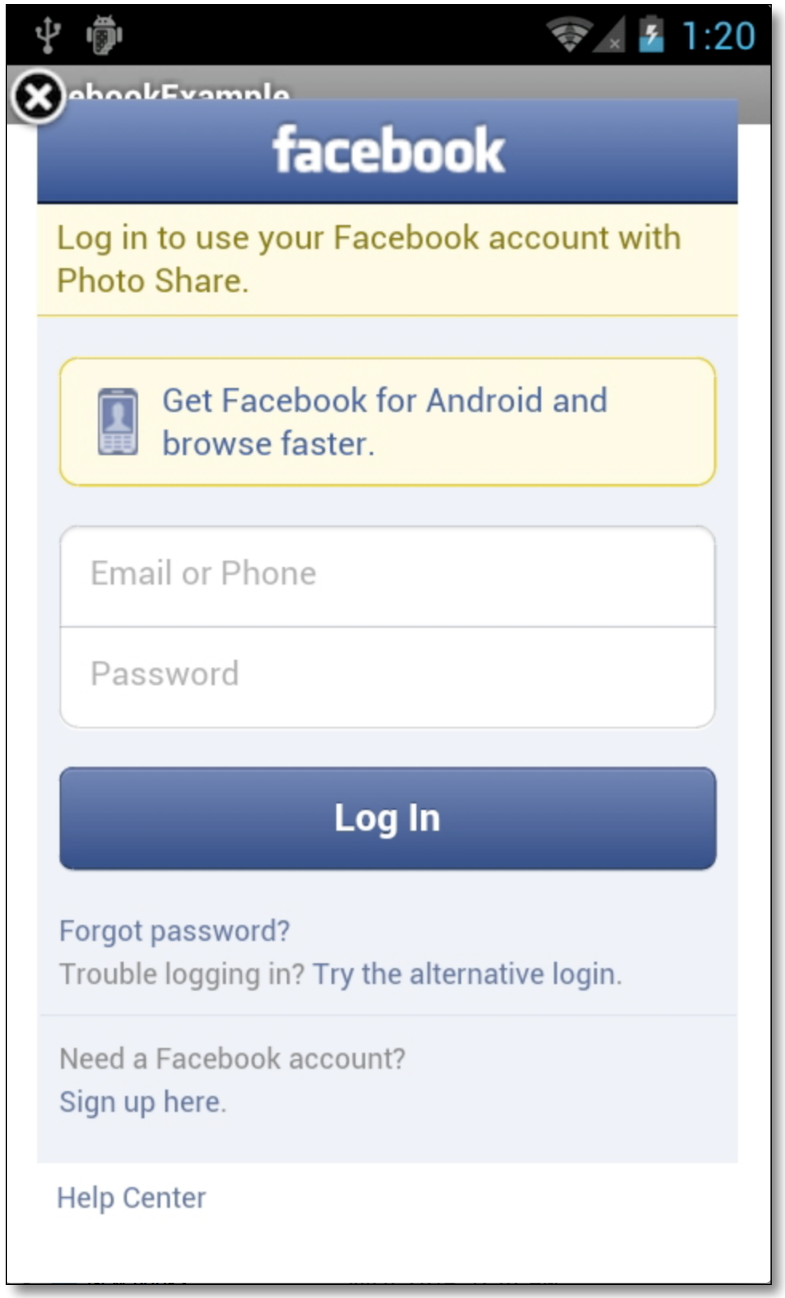
- **No isolation between the user-agent and the client (app). The client app is both the user-agent and the client app.**
- **A malicious hosting app can take control of the hosted web browser component and launch attacks on both the user authentication and application authorization pages:**
  - Can steal the user password

Type 1: Embedded Web Browser Component

- **No isolation between the user-agent and the client (app). The client app is both the user-agent and the client app.**
- **A malicious hosting app can take control of the hosted web browser component and launch attacks on both the user authentication and application authorization pages:**
  - Can steal the user password
  - Can modify the authorization page to spoof the user into authorizing permissions to the hosted app.

Type 1: Embedded Web Browser Component

- **Can register event handlers in the loaded page to send the username/password to the hosting app.**



Log in to use your Facebook account with Photo Share.

Get Facebook for Android and browse faster.

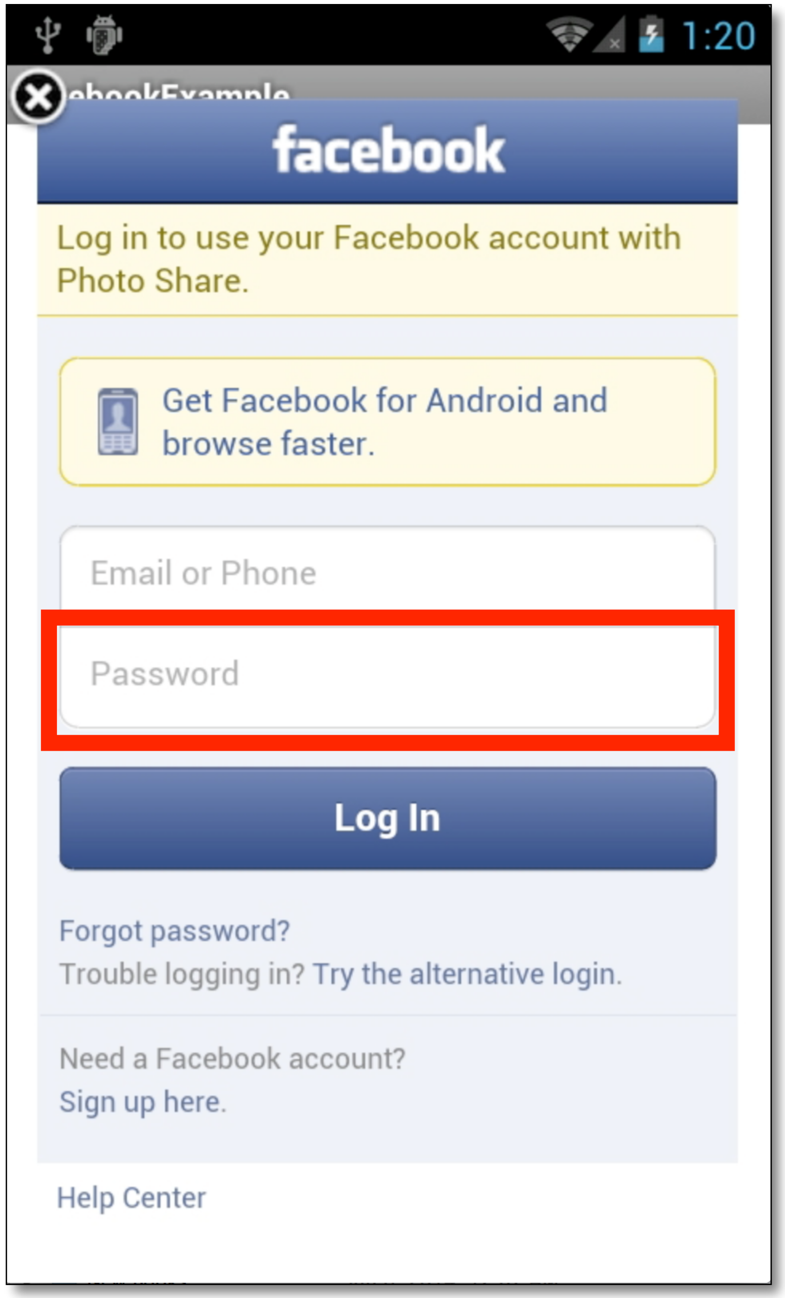
Forgot password?
Trouble logging in? Try the alternative login.

Need a Facebook account?
Sign up here.

Help Center

Type 1: Embedded Web Browser Component

- **Can register event handlers in the loaded page to send the username/password to the hosting app.**



1:20

ebookExample

facebook

Log in to use your Facebook account with Photo Share.

Get Facebook for Android and browse faster.

Email or Phone

Password

Log In

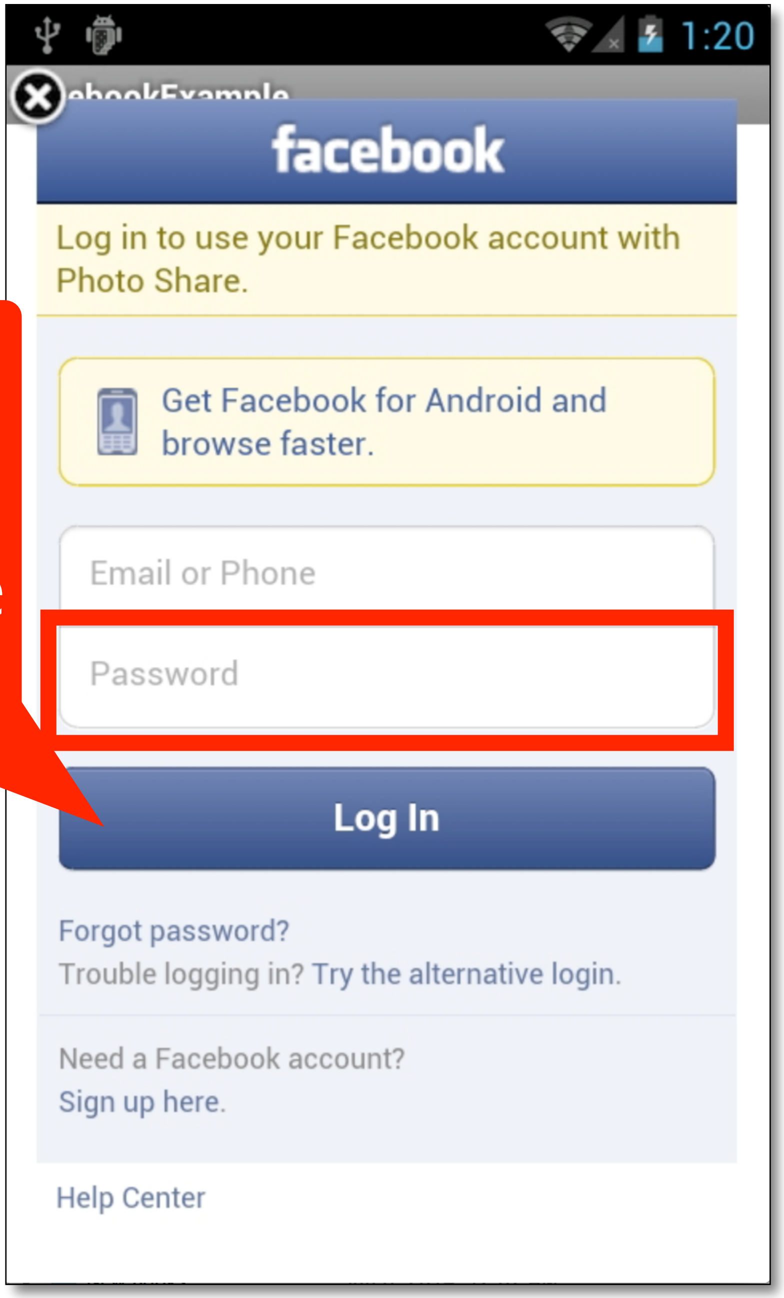
Forgot password?
Trouble logging in? Try the alternative login.

Need a Facebook account?
Sign up here.

Help Center

Type 1: Embedded Web Browser Component

- **Can register event handlers in the loaded page to send the username/password to the hosting app.**



1:20

ebookExample

facebook

Log in to use your Facebook account with Photo Share.

Get Facebook for Android and browse faster.

Email or Phone

Password

Log In

Forgot password?

Trouble logging in? Try the alternative login.

Need a Facebook account?

Sign up here.

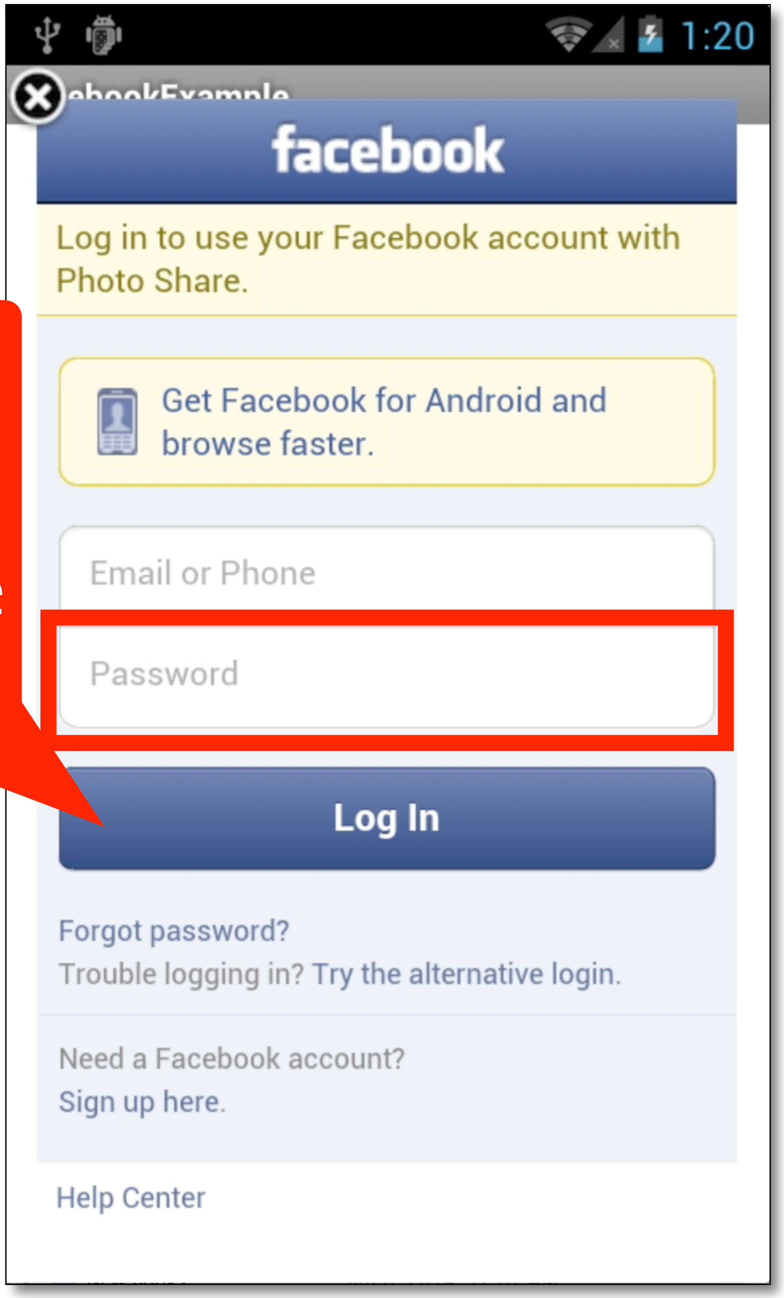
Help Center

Register an event handler to retrieve the password

Type 1: Embedded Web Browser Component

- **Can register event handlers in the loaded page to send the username/password to the hosting app.**

Register an event handler to retrieve the password



1:20

ebookExample

facebook

Log in to use your Facebook account with Photo Share.

Get Facebook for Android and browse faster.

Email or Phone

Password

Log In

Forgot password?
Trouble logging in? Try the alternative login.

Need a Facebook account?
Sign up here.

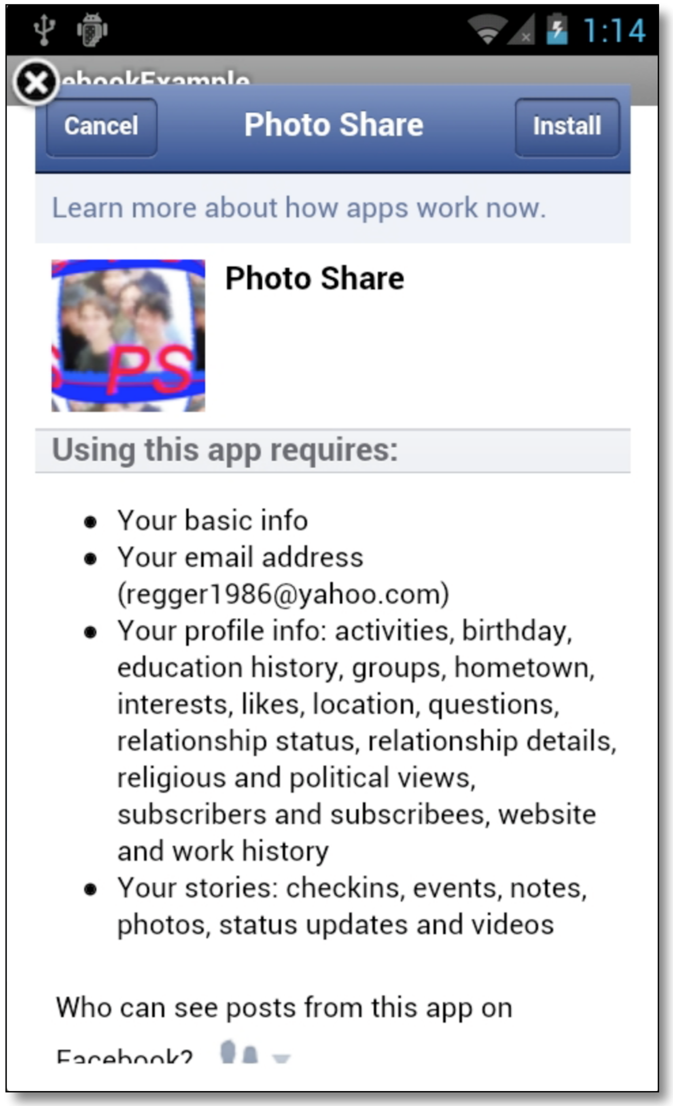
Help Center

```
//Java (Native Client App)
myWebView.getSettings().setJavaScriptEnabled(true);
myWebView.addJavaScriptInterface(this , "JSInterface");
myWebView.loadUrl("javascript:" + contents of attack.js);


//JavaScript (attack.js)
var submitBtn = document.getElementById('btn_id');
submitBtn.onclick = function(){
    var email = document.getElementById('email_id').value;
    var password = document.getElementById('pwd_id').value;
    JSInterface.jsCall(email, password);
    return true;
}
```

Type 1: Embedded Web Browser Component

- **Manipulate the authorization page to show a different set of permissions than what the user is actually authorizing.**



Learn more about how apps work now.

Photo Share

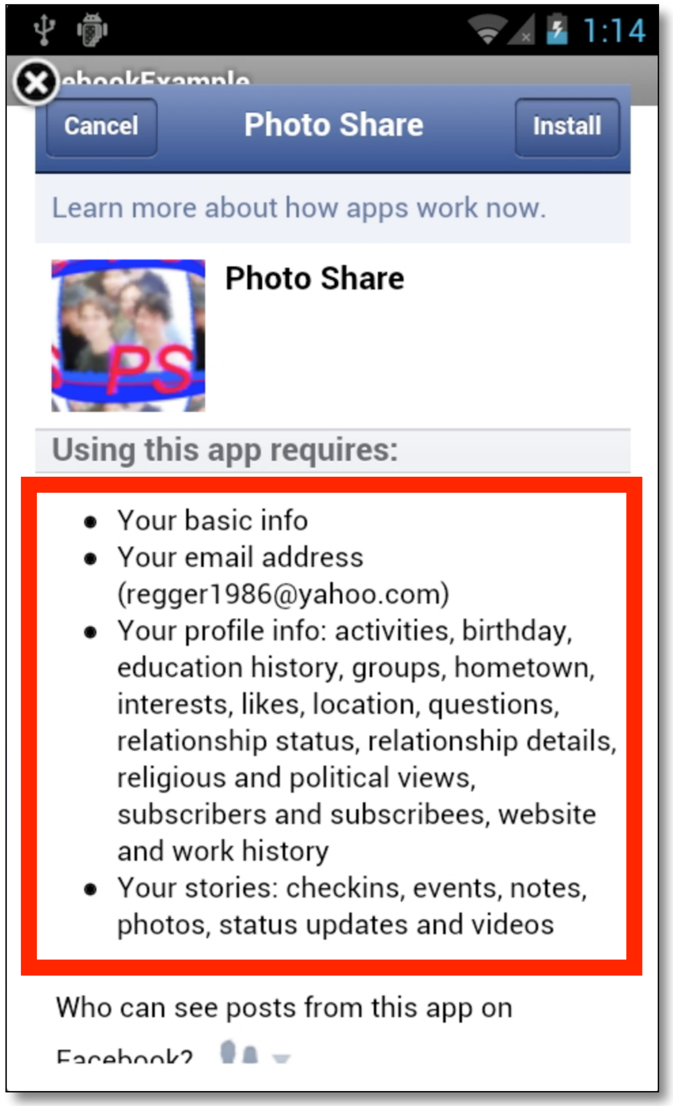
Using this app requires:

- Your basic info
- Your email address (regger1986@yahoo.com)
- Your profile info: activities, birthday, education history, groups, hometown, interests, likes, location, questions, relationship status, relationship details, religious and political views, subscribers and subscribees, website and work history
- Your stories: checkins, events, notes, photos, status updates and videos

Who can see posts from this app on Facebook?

Type 1: Embedded Web Browser Component

- **Manipulate the authorization page to show a different set of permissions than what the user is actually authorizing.**



Learn more about how apps work now.

Photo Share

Using this app requires:

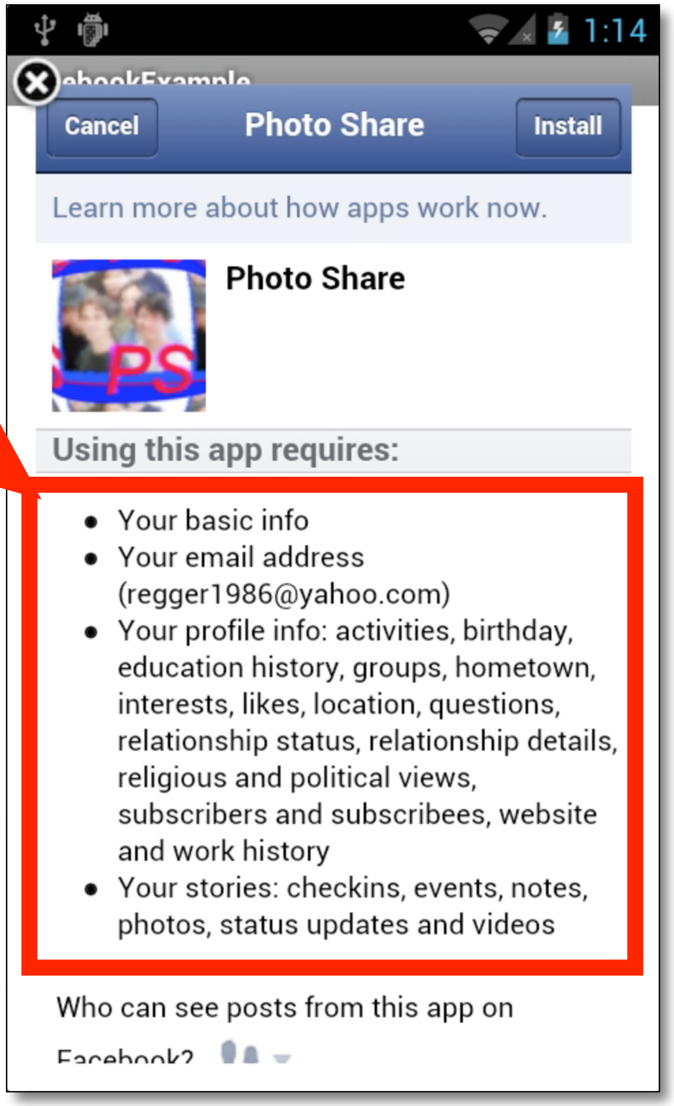
- Your basic info
- Your email address (regger1986@yahoo.com)
- Your profile info: activities, birthday, education history, groups, hometown, interests, likes, location, questions, relationship status, relationship details, religious and political views, subscribers and subscribees, website and work history
- Your stories: checkins, events, notes, photos, status updates and videos

Who can see posts from this app on Facebook?

Type 1: Embedded Web Browser Component

- **Manipulate the authorization page to show a different set of permissions than what the user is actually authorizing.**

Original requested permissions



Learn more about how apps work now.

Photo Share

Using this app requires:

- Your basic info
- Your email address (regger1986@yahoo.com)
- Your profile info: activities, birthday, education history, groups, hometown, interests, likes, location, questions, relationship status, relationship details, religious and political views, subscribers and subscribees, website and work history
- Your stories: checkins, events, notes, photos, status updates and videos

Who can see posts from this app on Facebook?

Type 1: Embedded Web Browser Component

- **Manipulate the authorization page to show a different set of permissions than what the user is actually authorizing.**

Original requested permissions

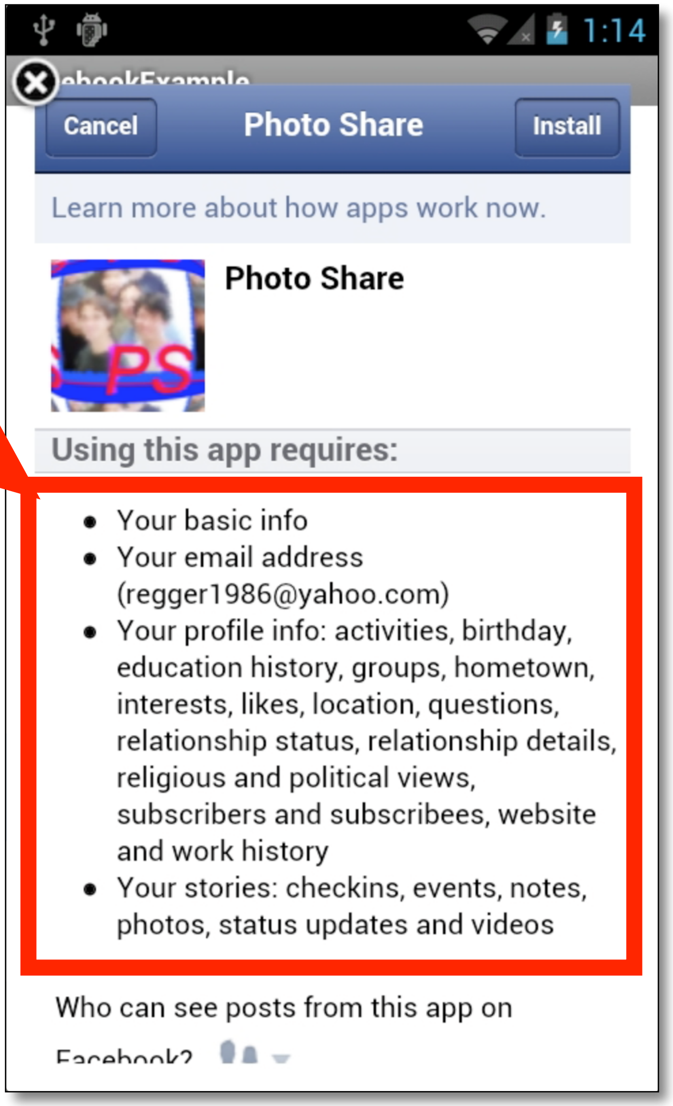


Photo Share

Learn more about how apps work now.

Photo Share

Using this app requires:

- Your basic info
- Your email address (regger1986@yahoo.com)
- Your profile info: activities, birthday, education history, groups, hometown, interests, likes, location, questions, relationship status, relationship details, religious and political views, subscribers and subscribees, website and work history
- Your stories: checkins, events, notes, photos, status updates and videos

Who can see posts from this app on Facebook?

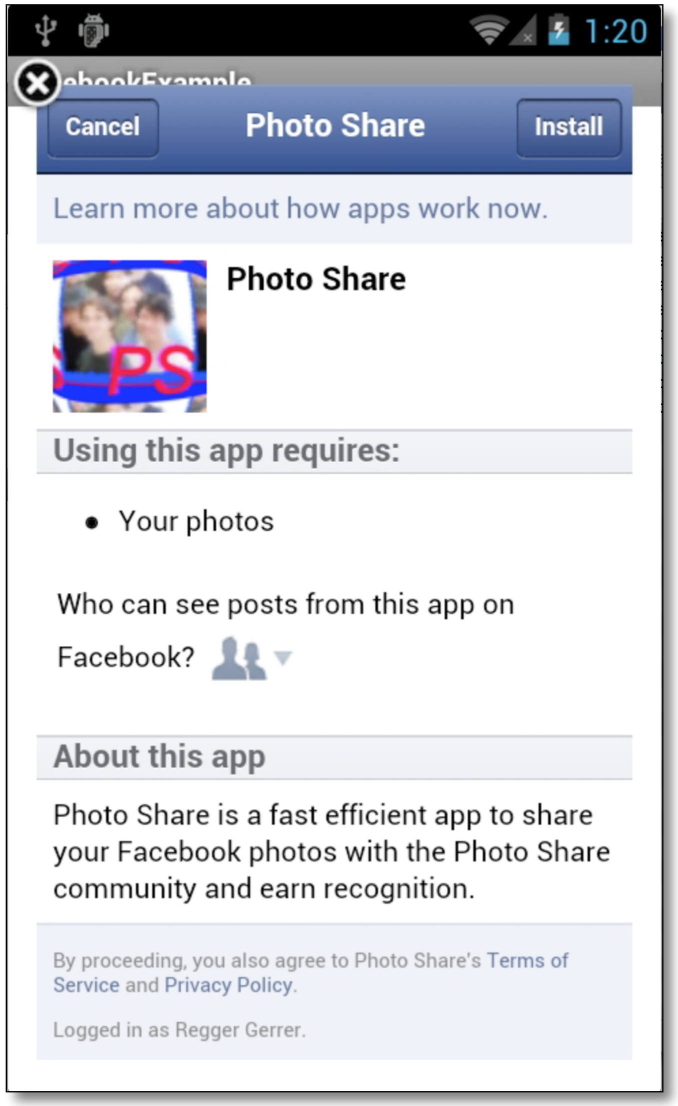


Photo Share

Learn more about how apps work now.

Photo Share

Using this app requires:

- Your photos

Who can see posts from this app on Facebook?

About this app

Photo Share is a fast efficient app to share your Facebook photos with the Photo Share community and earn recognition.

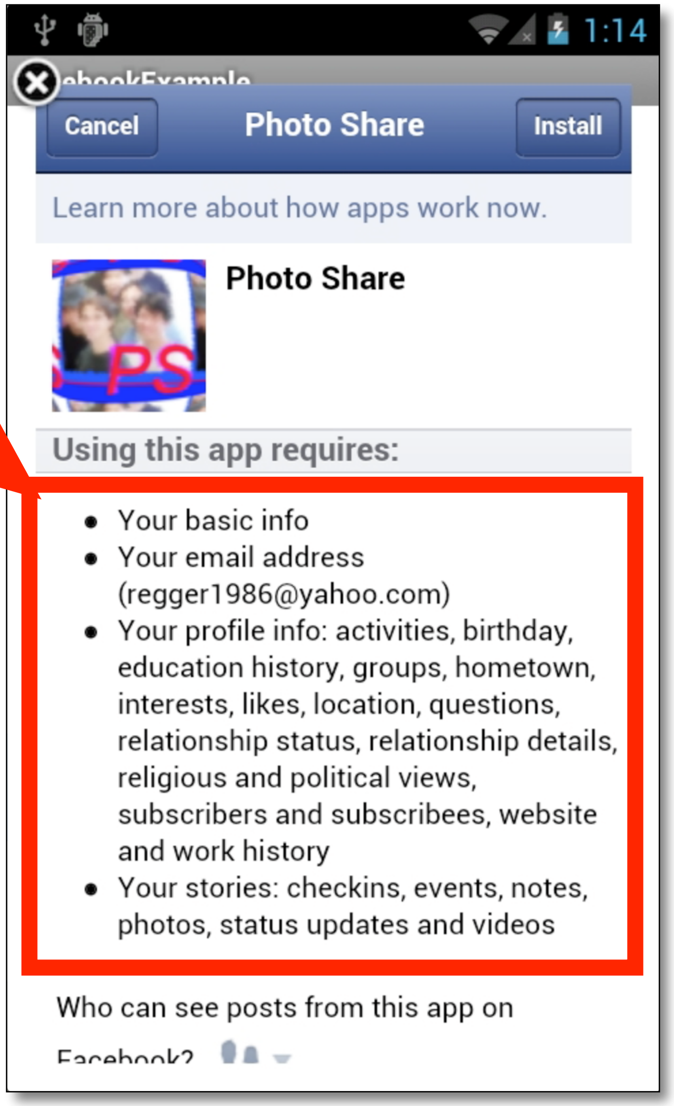
By proceeding, you also agree to Photo Share's Terms of Service and Privacy Policy.

Logged in as Regger Gerrer.

Type 1: Embedded Web Browser Component

- **Manipulate the authorization page to show a different set of permissions than what the user is actually authorizing.**

Original requested permissions



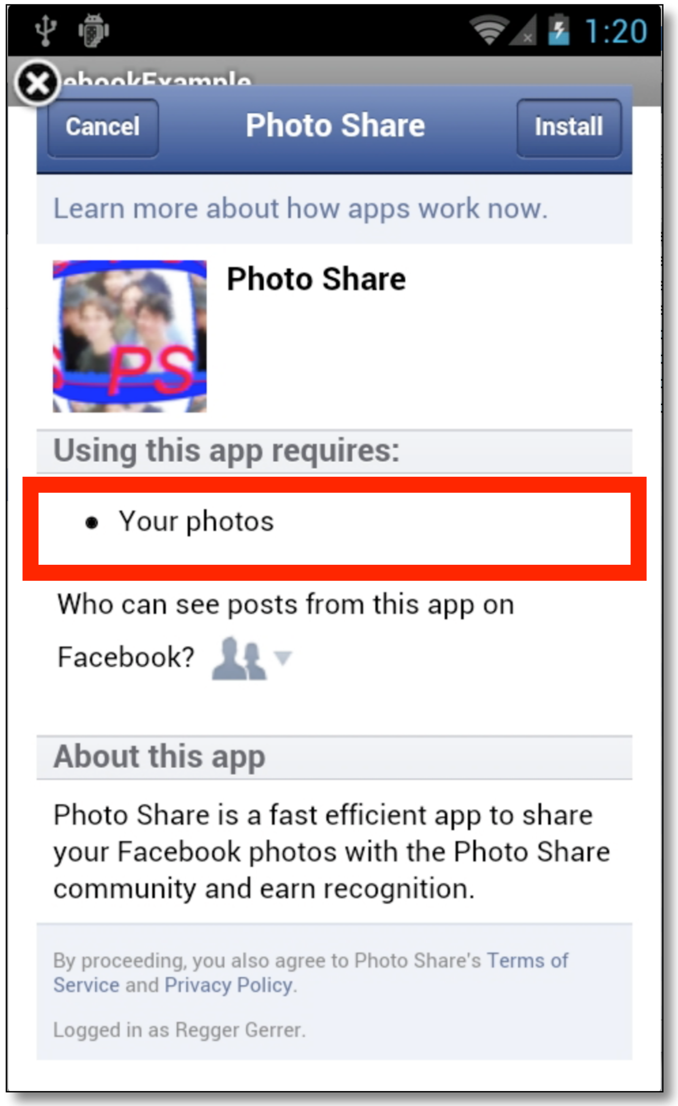
Learn more about how apps work now.

Photo Share

Using this app requires:

- Your basic info
- Your email address (regger1986@yahoo.com)
- Your profile info: activities, birthday, education history, groups, hometown, interests, likes, location, questions, relationship status, relationship details, religious and political views, subscribers and subscribees, website and work history
- Your stories: checkins, events, notes, photos, status updates and videos

Who can see posts from this app on Facebook?



Learn more about how apps work now.

Photo Share

Using this app requires:

- Your photos

Who can see posts from this app on Facebook?

About this app

Photo Share is a fast efficient app to share your Facebook photos with the Photo Share community and earn recognition.

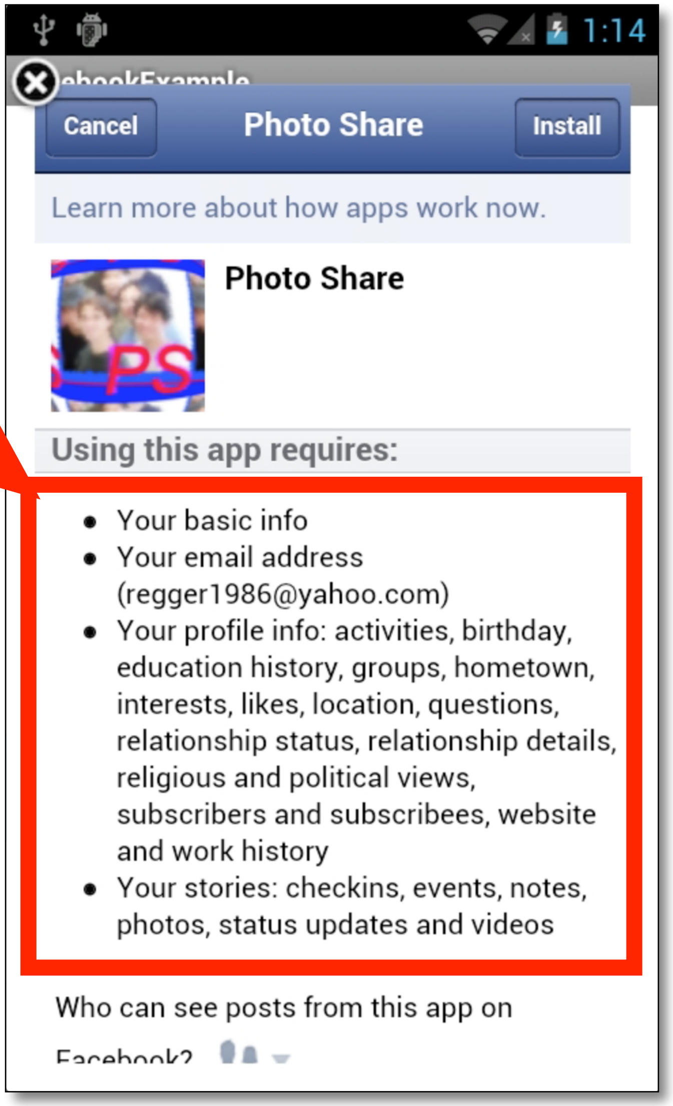
By proceeding, you also agree to Photo Share's Terms of Service and Privacy Policy.

Logged in as Regger Gerrer.

Type 1: Embedded Web Browser Component

- **Manipulate the authorization page to show a different set of permissions than what the user is actually authorizing.**

Original requested permissions



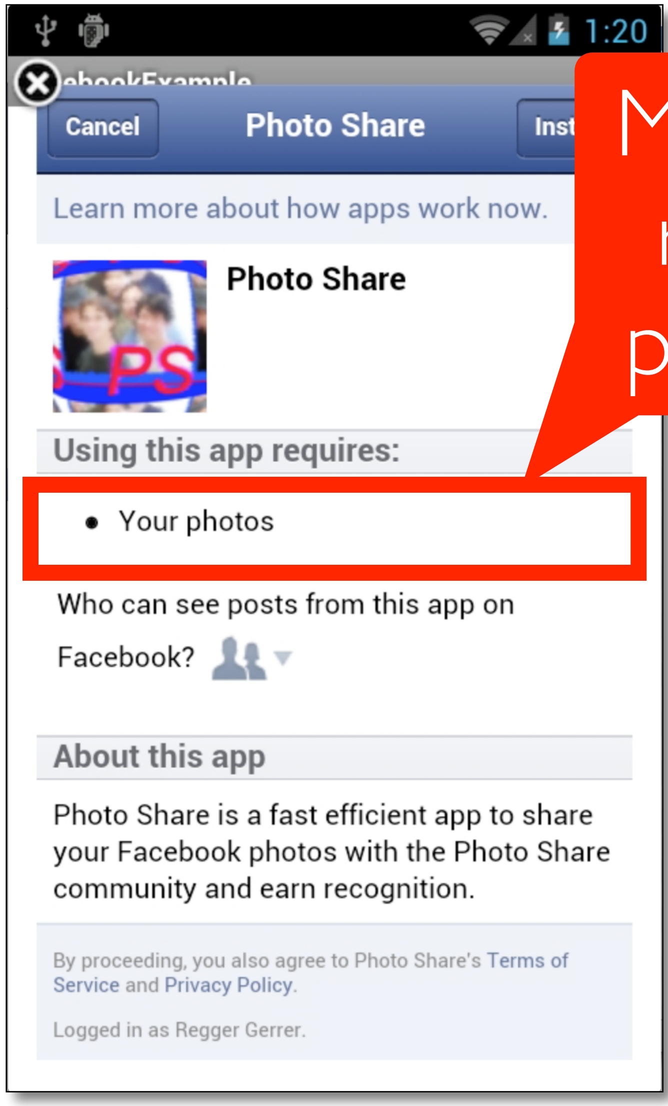
Learn more about how apps work now.

Photo Share

Using this app requires:

- Your basic info
- Your email address (regger1986@yahoo.com)
- Your profile info: activities, birthday, education history, groups, hometown, interests, likes, location, questions, relationship status, relationship details, religious and political views, subscribers and subscribees, website and work history
- Your stories: checkins, events, notes, photos, status updates and videos

Who can see posts from this app on Facebook?



Learn more about how apps work now.

Photo Share

Using this app requires:

- Your photos

Who can see posts from this app on Facebook?

About this app

Photo Share is a fast efficient app to share your Facebook photos with the Photo Share community and earn recognition.

By proceeding, you also agree to Photo Share's Terms of Service and Privacy Policy.

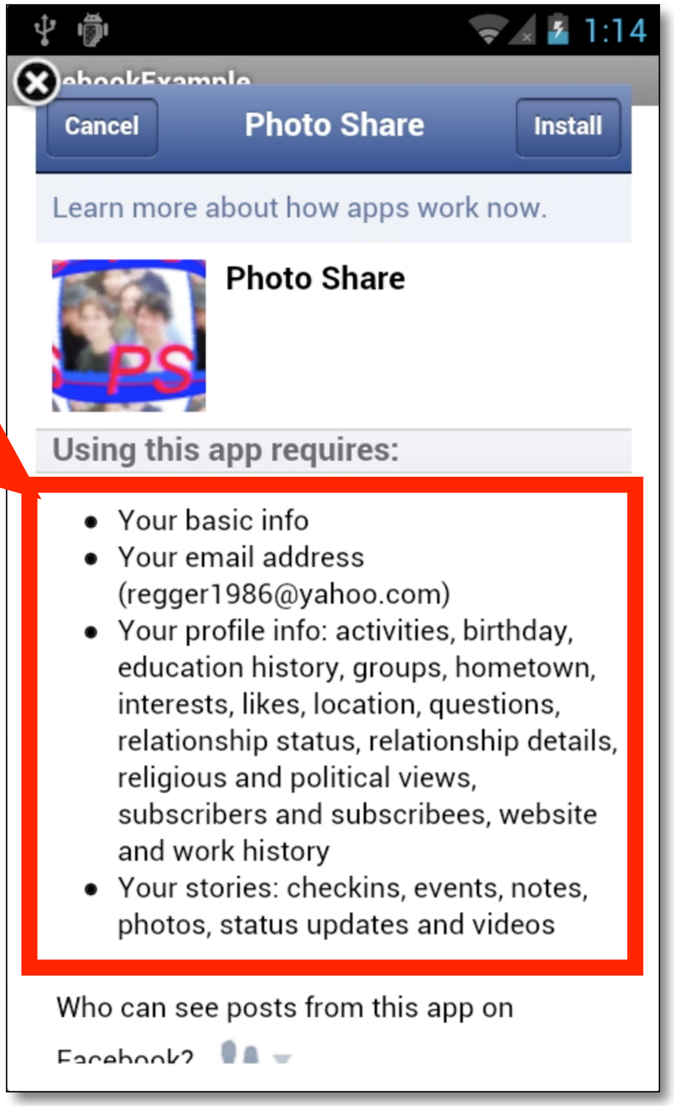
Logged in as Regger Gerrer.

Manipulated requested permissions

Type 1: Embedded Web Browser Component

- **Manipulate the authorization page to show a different set of permissions than what the user is actually authorizing.**

Original requested permissions



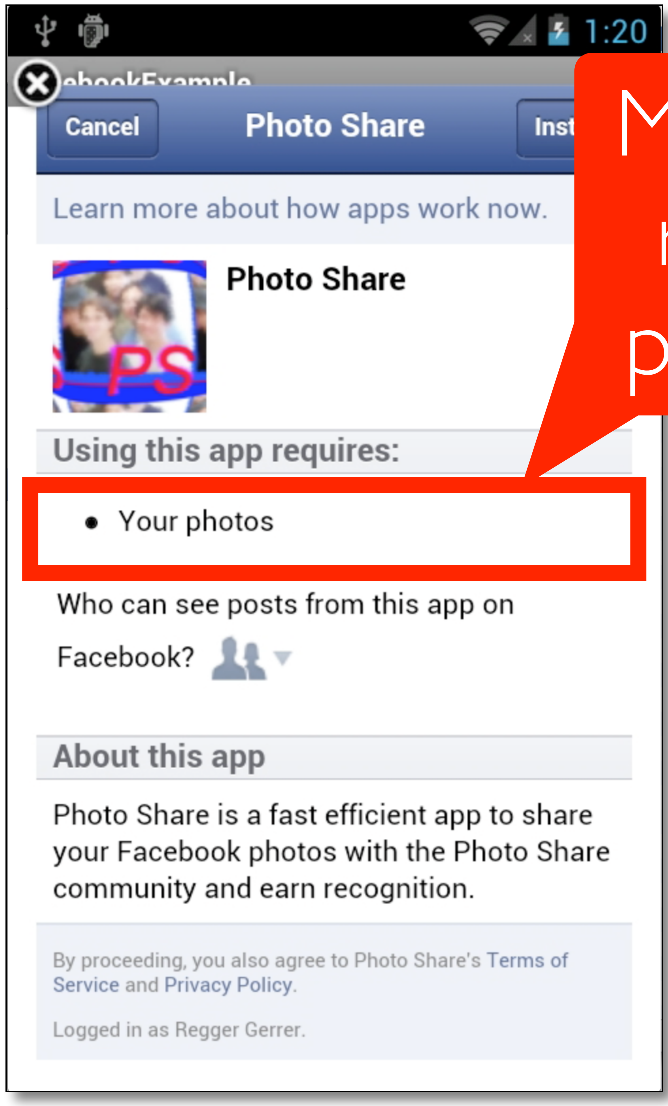
Learn more about how apps work now.

Photo Share

Using this app requires:

- Your basic info
- Your email address (regger1986@yahoo.com)
- Your profile info: activities, birthday, education history, groups, hometown, interests, likes, location, questions, relationship status, relationship details, religious and political views, subscribers and subscribees, website and work history
- Your stories: checkins, events, notes, photos, status updates and videos

Who can see posts from this app on Facebook?



Learn more about how apps work now.

Photo Share

Using this app requires:

- Your photos

Who can see posts from this app on Facebook?

About this app

Photo Share is a fast efficient app to share your Facebook photos with the Photo Share community and earn recognition.

By proceeding, you also agree to Photo Share's Terms of Service and Privacy Policy.

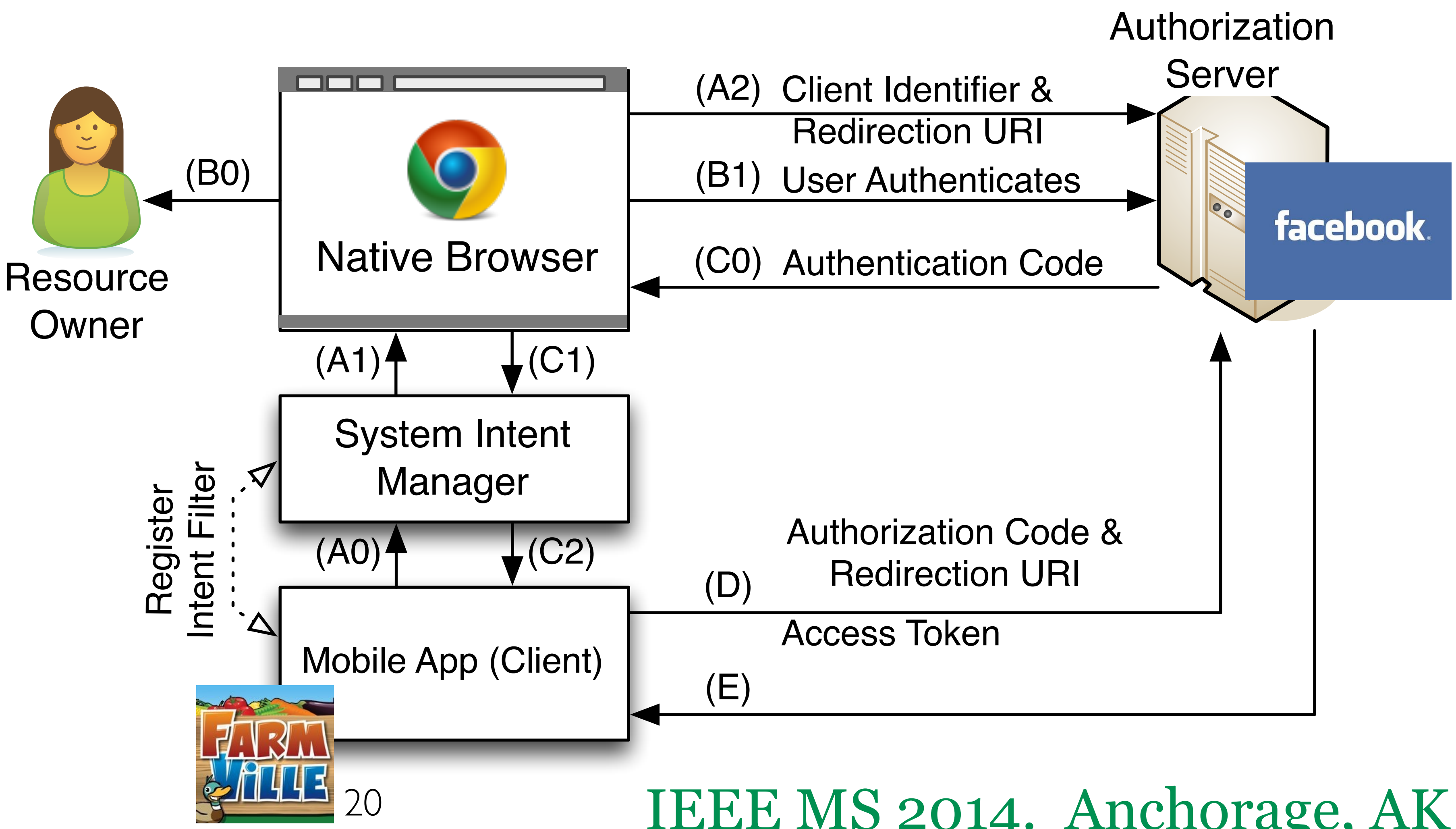
Logged in as Regger Gerrer.

Manipulated requested permissions

```
var permsUL = document.getElementById('perm_ul');
var permsUL.innerHTML = '<li><div>Your photos</div></li>';
```

Type 2: Using the Native Browser

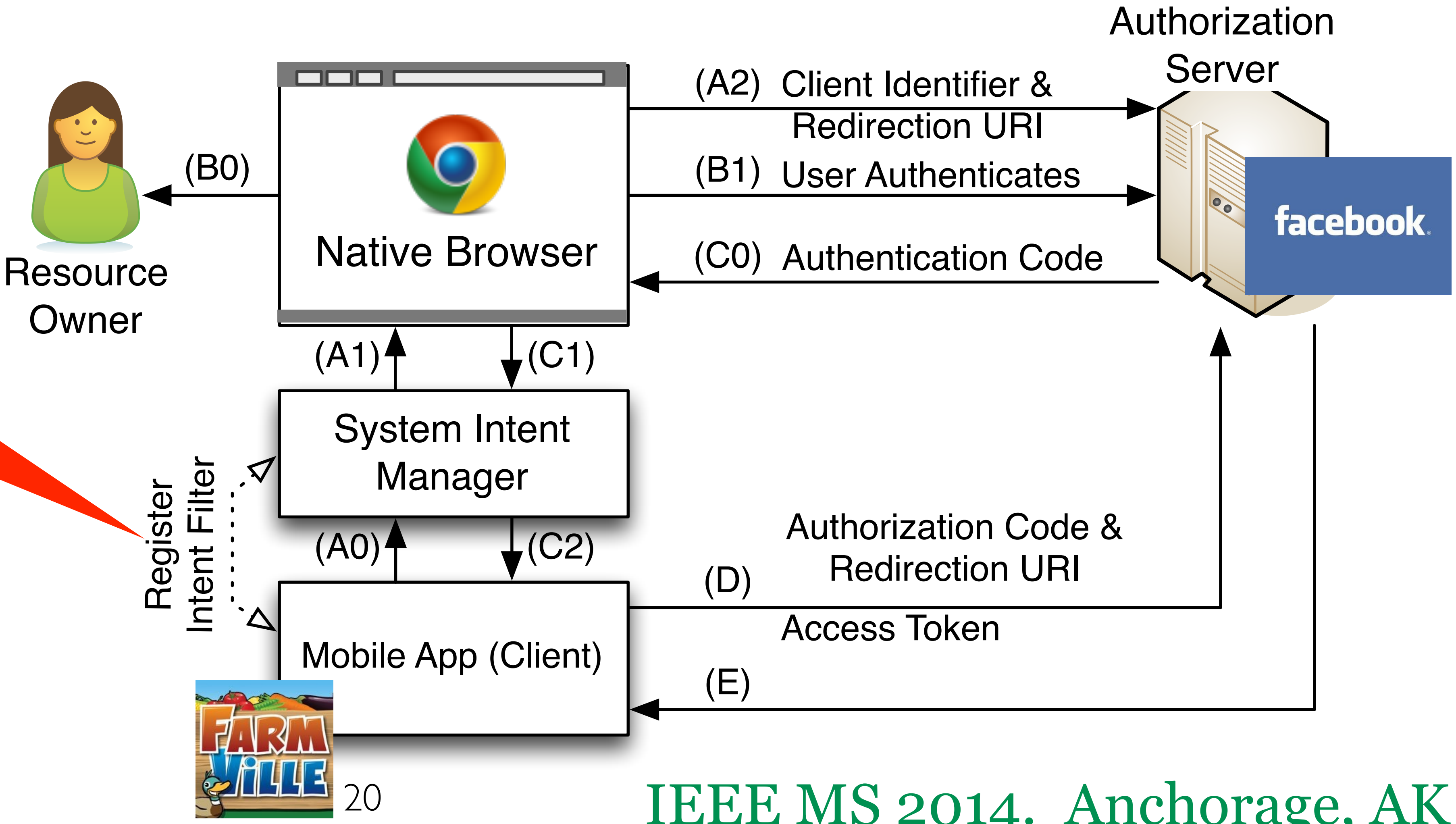
- **The client app registers a system wide handler to listen to specific data requests of type *"data-code://auth-token"***
- **The client app sends the user to the native web browser to perform the authentication and authorization stages.**
- **The browser sends the access token by invoking a request to a url *"data-code://auth-token"***



Type 2: Using the Native Browser

- **The client app registers a system wide handler to listen to specific data requests of type *“data-code://auth-token“***
- **The client app sends the user to the native web browser to perform the authentication and authorization stages.**
- **The browser sends the access token by invoking a request to a url *“data-code://auth-token“***

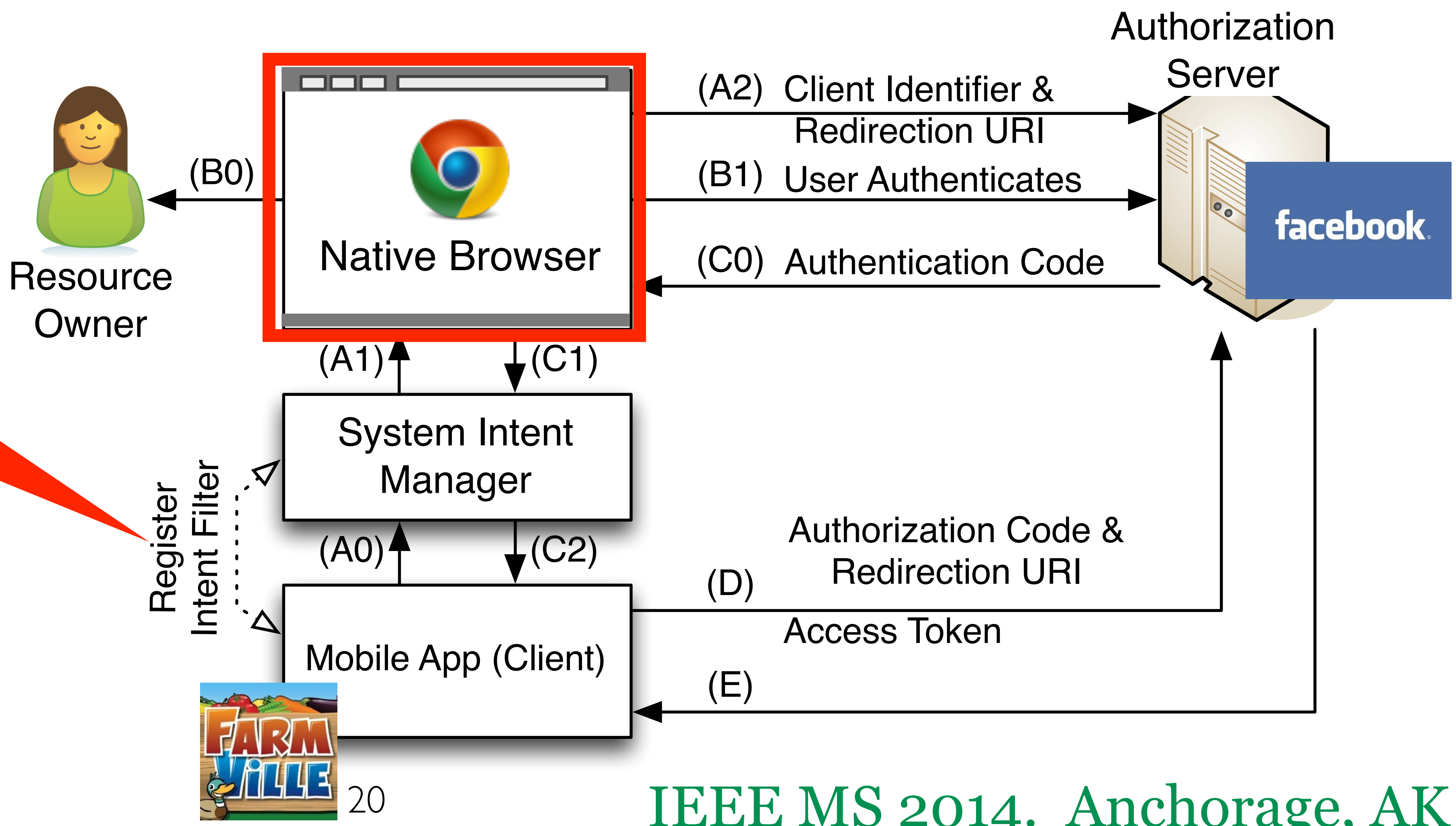
Message passing managed by the mobile framework.



Type 2: Using the Native Browser

- **The client app registers a system wide handler to listen to specific data requests of type *"data-code://auth-token"***
- **The client app sends the user to the native web browser to perform the authentication and authorization stages.**
- **The browser sends the access token by invoking a request to a url *"data-code://auth-token"***

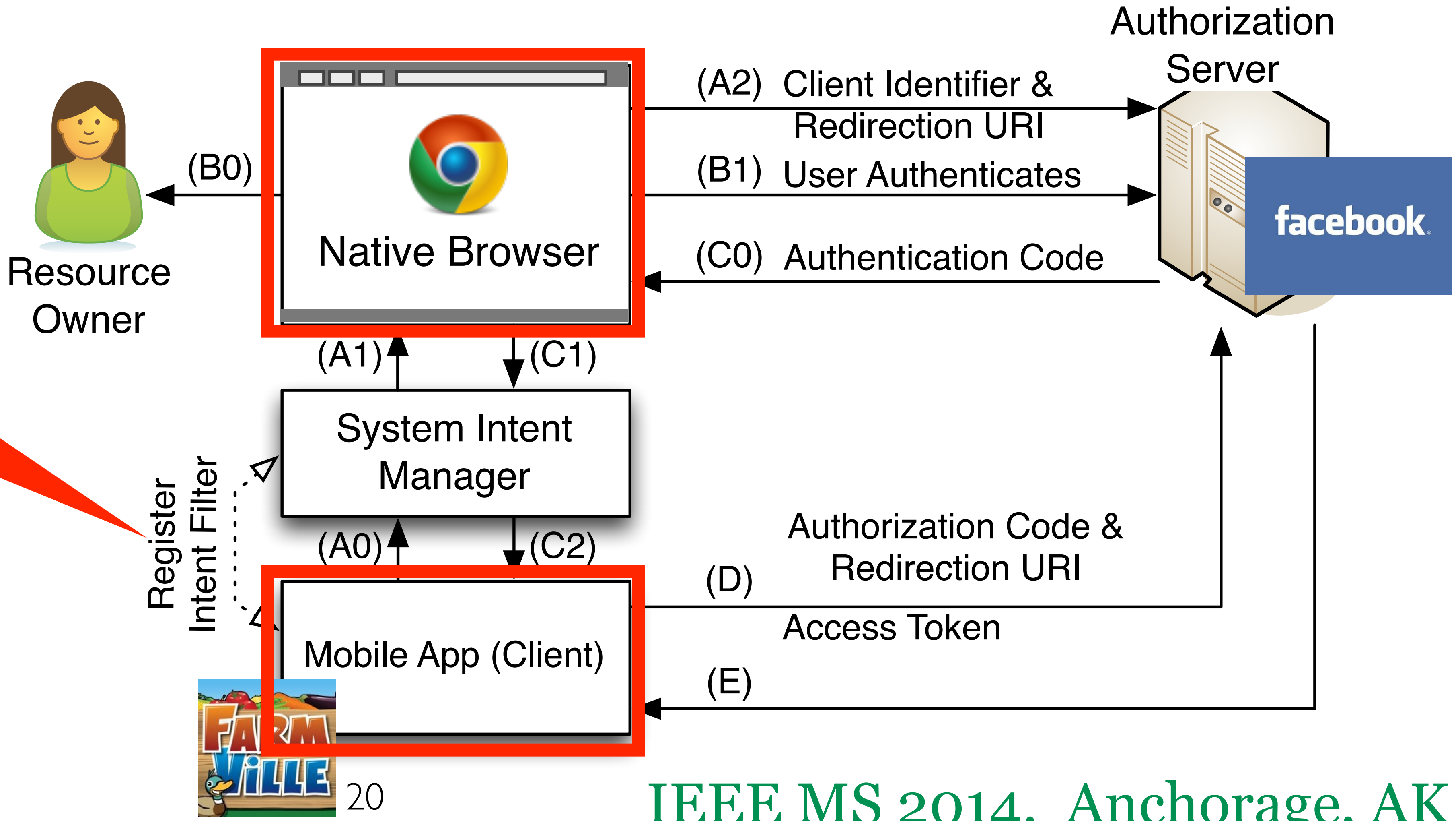
Message passing managed by the mobile framework.



Type 2: Using the Native Browser

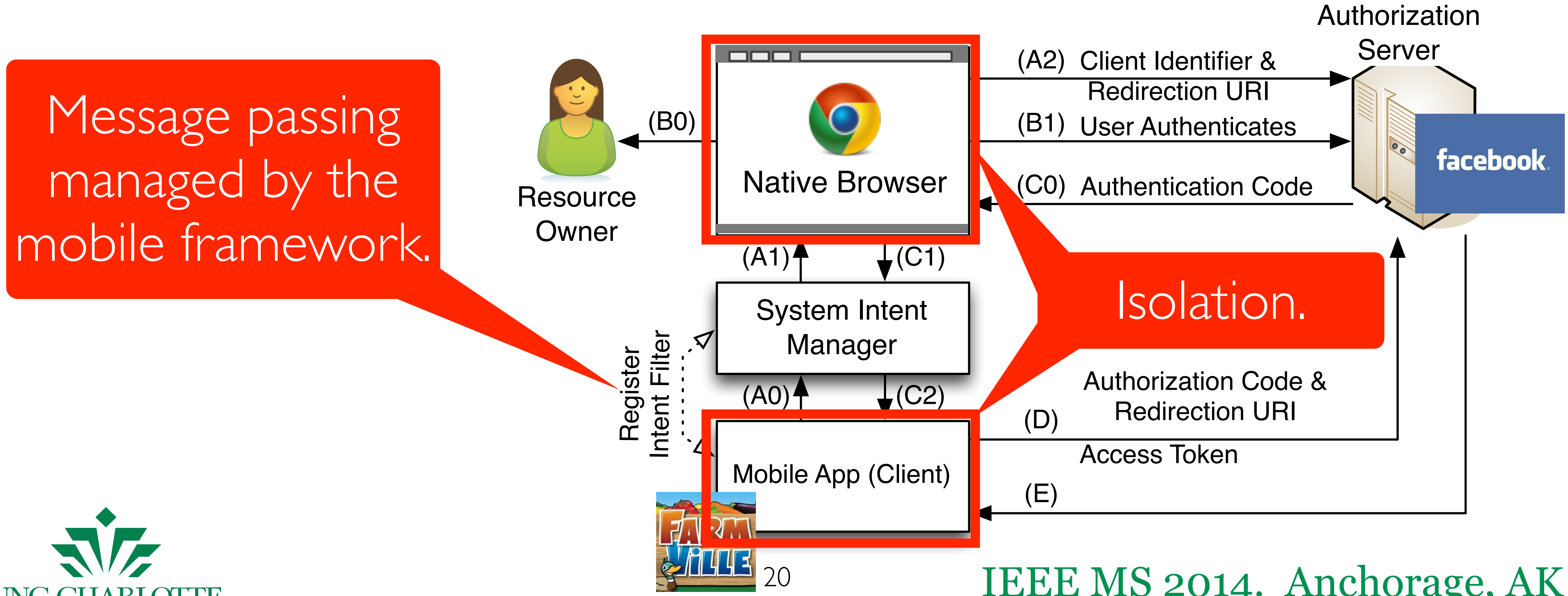
- **The client app registers a system wide handler to listen to specific data requests of type *"data-code://auth-token"***
- **The client app sends the user to the native web browser to perform the authentication and authorization stages.**
- **The browser sends the access token by invoking a request to a url *"data-code://auth-token"***

Message passing managed by the mobile framework.



Type 2: Using the Native Browser

- **The client app registers a system wide handler to listen to specific data requests of type *"data-code://auth-token"***
- **The client app sends the user to the native web browser to perform the authentication and authorization stages.**
- **The browser sends the access token by invoking a request to a url *"data-code://auth-token"***



Message passing managed by the mobile framework.

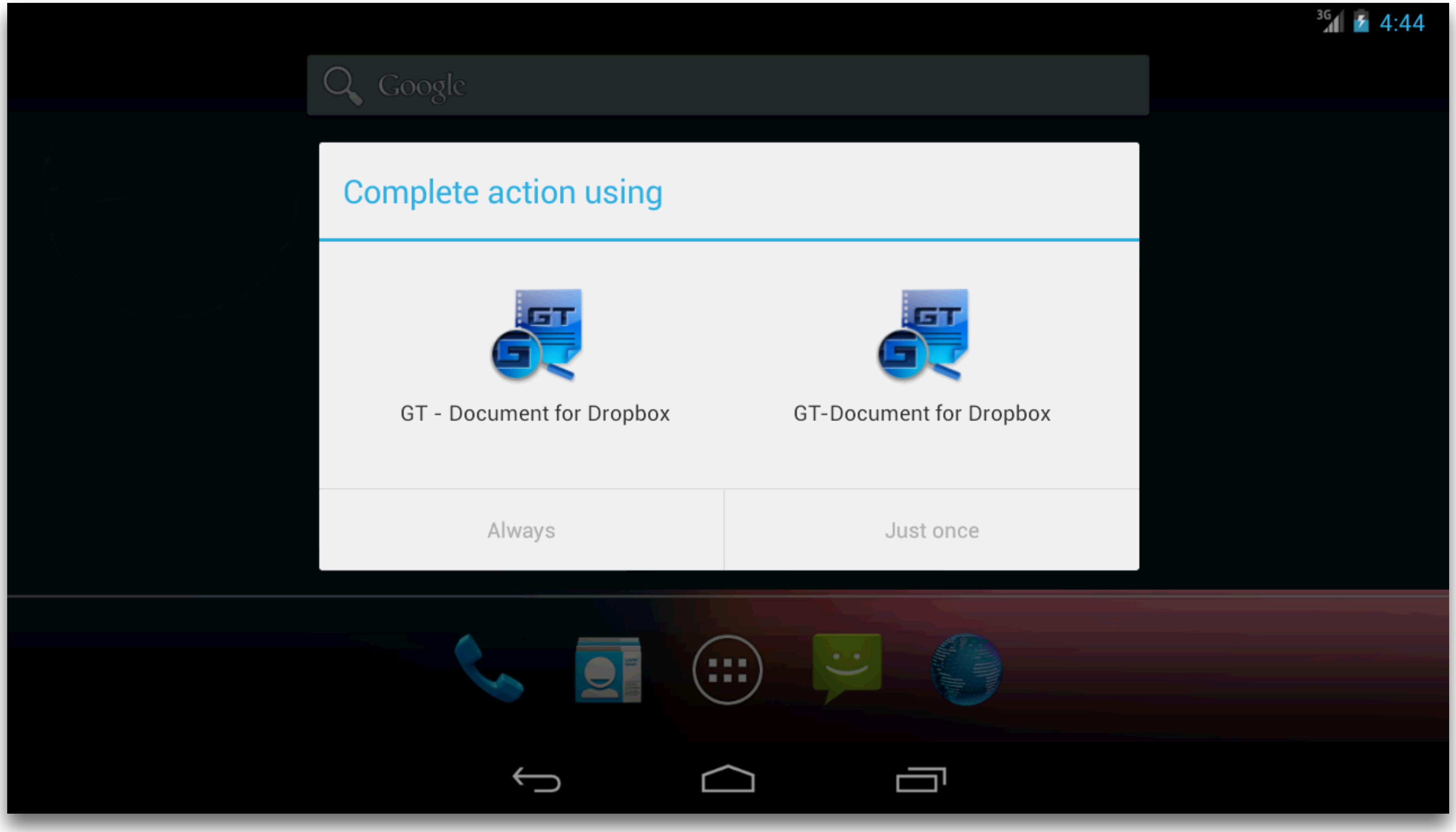
Isolation.

Type 2: Using the Native Browser

- **Using the native browser provides the required isolation, however the token can be stolen when it is being returned to the client app.**
- **A malicious app can exploit the channel between the browser and the client app.**
- ***Impersonation Attack:*** **A malicious app an register to listen to the same specific data request that the client app is registered to listen to, which could result in passing the access token to the malicious app.**

Type 2: Using the Native Browser

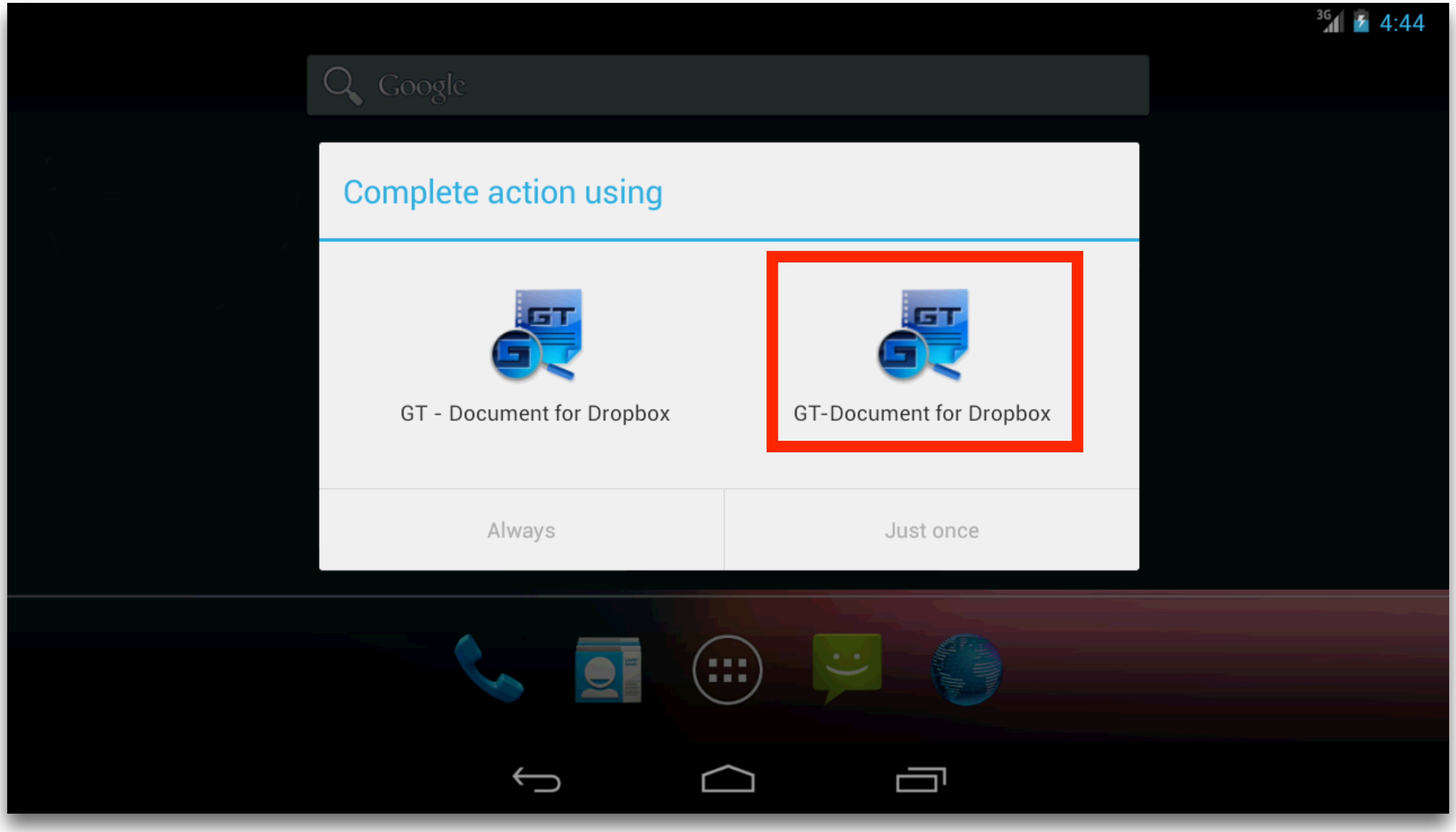
- **Because more than one app has registered to listen to the same data type, the user will be asked to choose which app to start.**



Complete action using

Type 2: Using the Native Browser

- **Because more than one app has registered to listen to the same data type, the user will be asked to choose which app to start.**



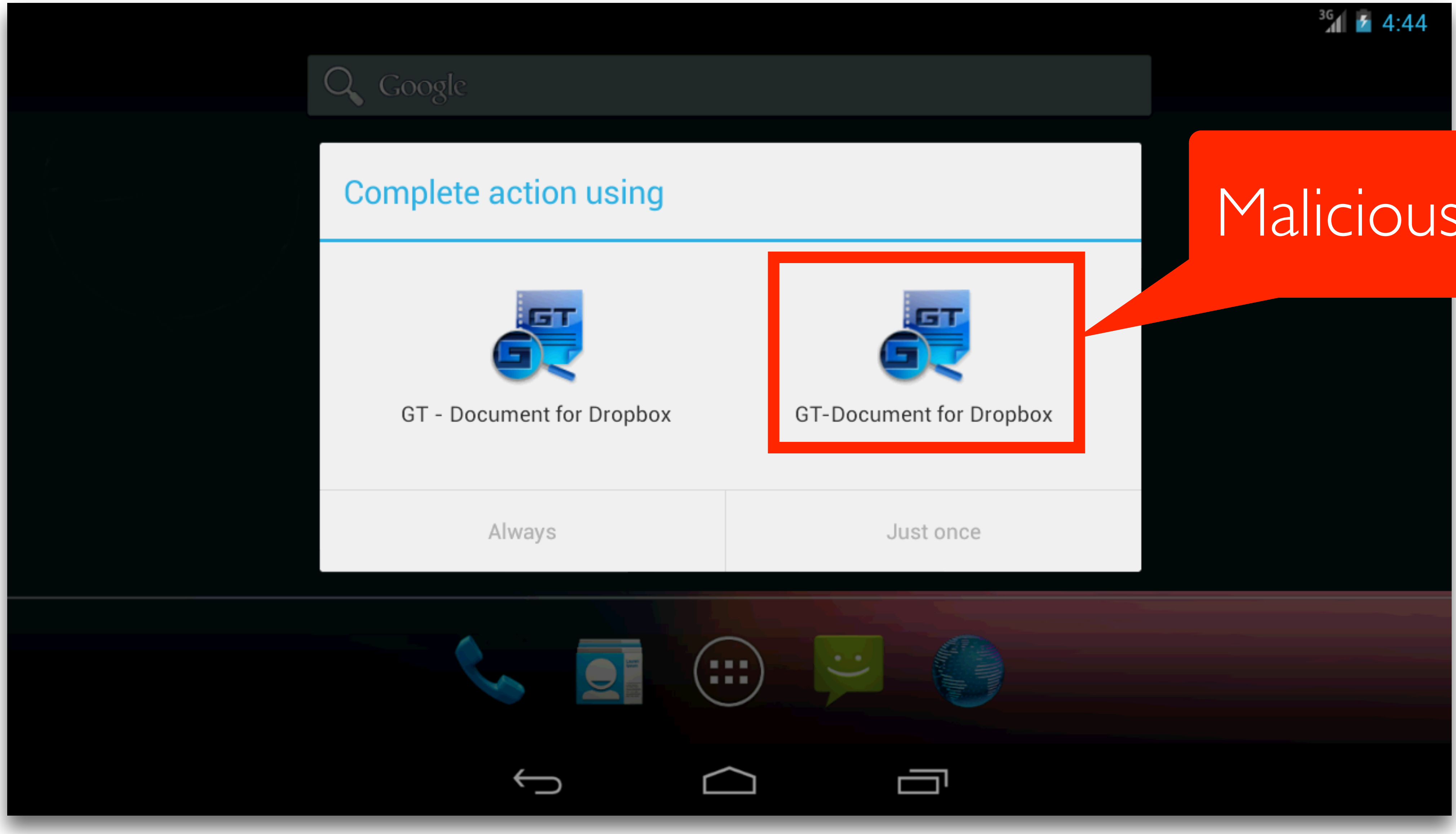
Complete action using

GT - Document for Dropbox

GT-Document for Dropbox

Type 2: Using the Native Browser

- **Because more than one app has registered to listen to the same data type, the user will be asked to choose which app to start.**



Complete action using

GT - Document for Dropbox

GT-Document for Dropbox

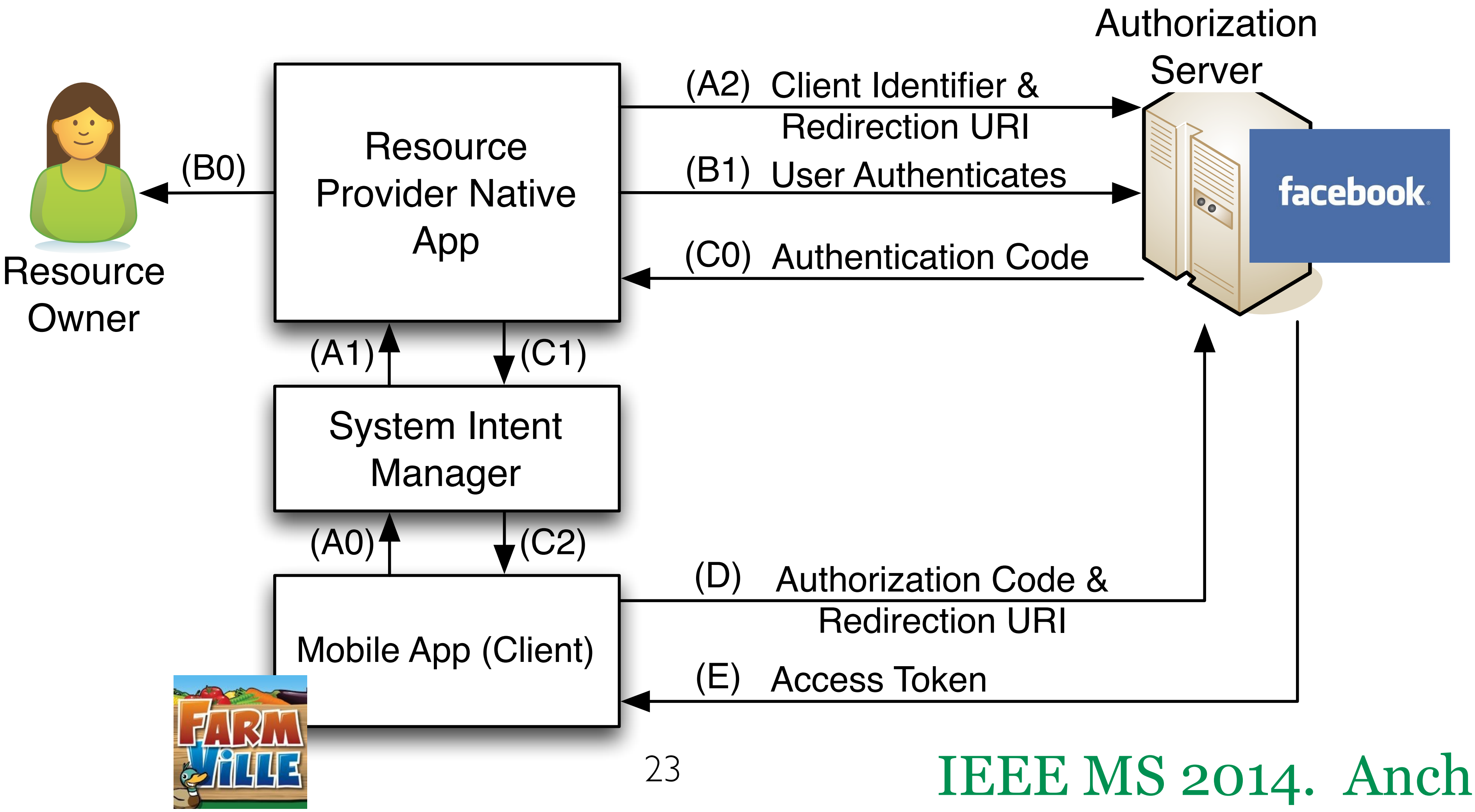
Always

Just once

Malicious App

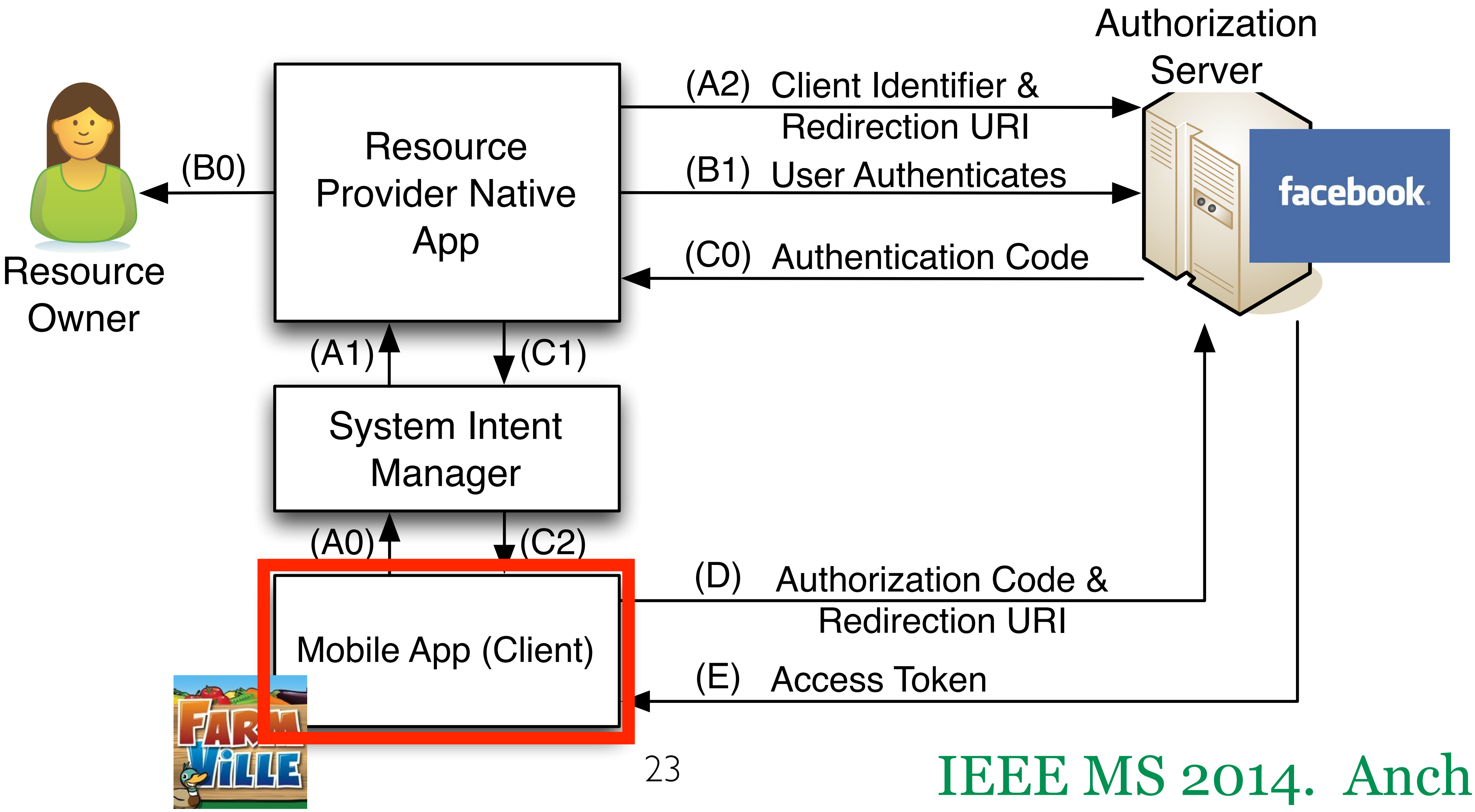
Type 3: Using the Resource Provider's App

- **This approach requires the resource provider's native app to be an installed on the smart phone. It is assumed that that the provider's app is trusted.**
- **The client app sends the user to the provider's native app to perform the authentication and authorization stages.**



Type 3: Using the Resource Provider's App

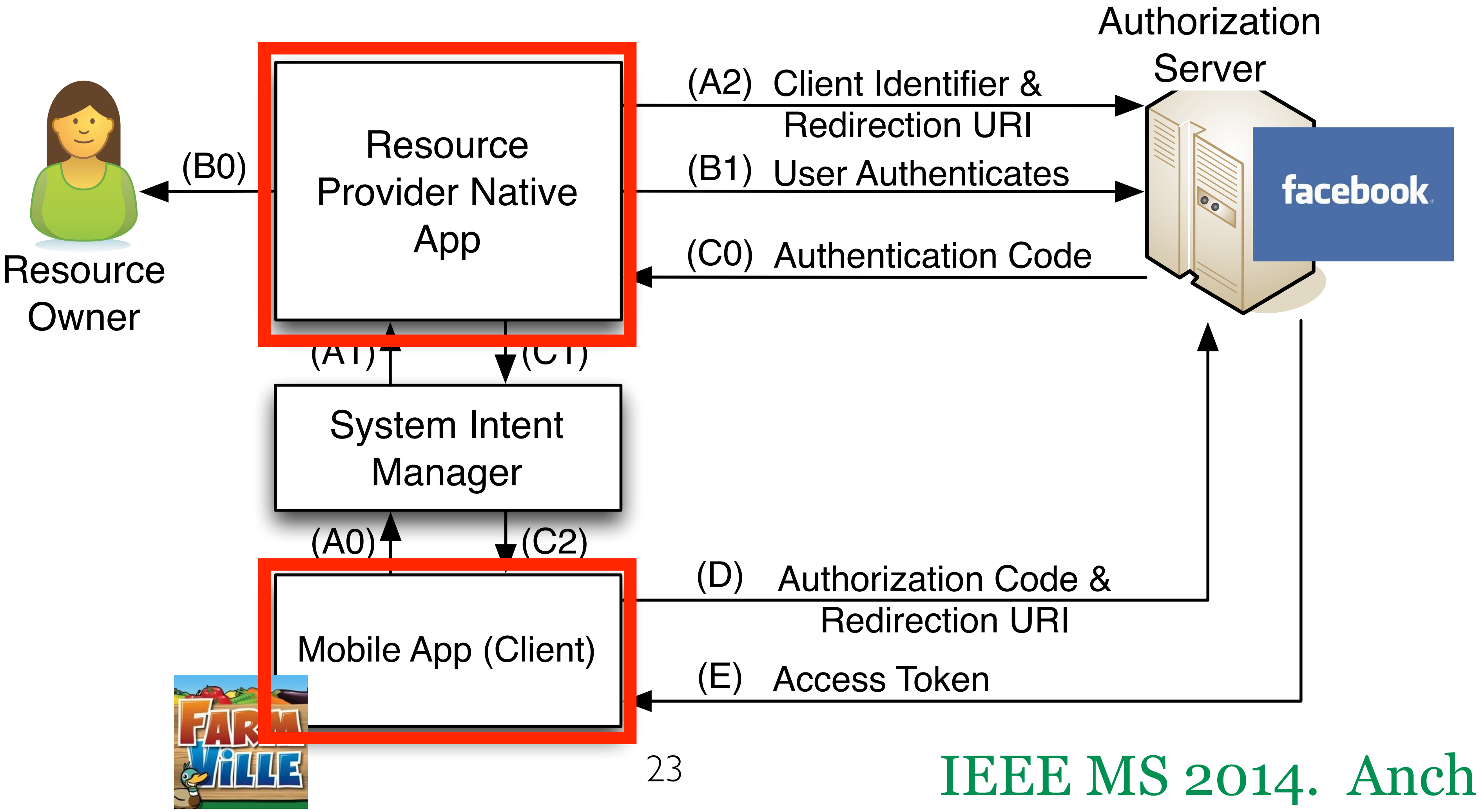
- This approach requires the resource provider's native app to be an installed on the smart phone. It is assumed that that the provider's app is trusted.
- The client app sends the user to the provider's native app to perform the authentication and authorization stages.



Type 3: Using the Resource Provider's App

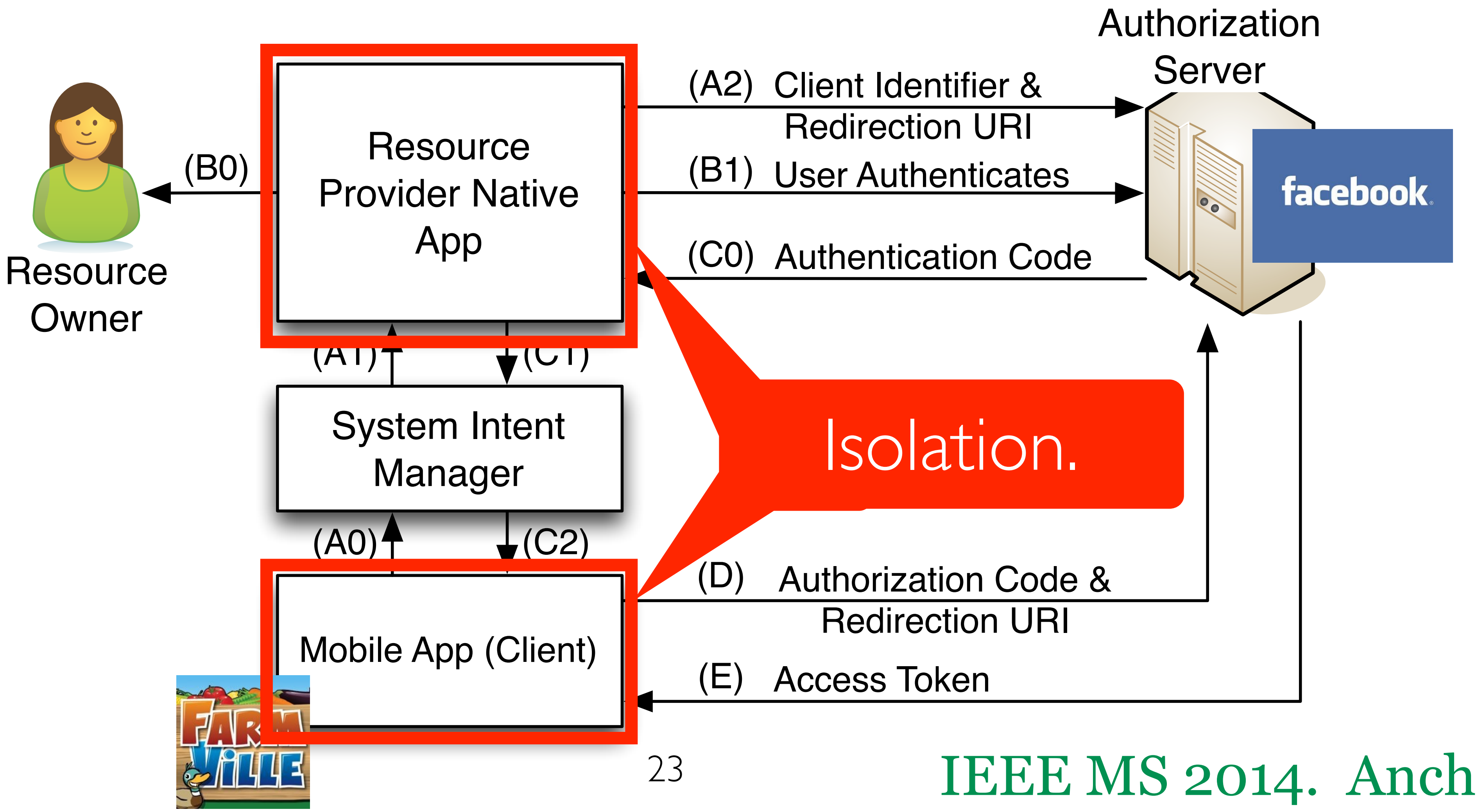
- **This approach requires the resource provider's native app to be an installed on the smart phone. It is assumed that that the provider's app is trusted.**

- **The client app sends the user to the provider's native app to perform the authentication and authorization stages.**



Type 3: Using the Resource Provider's App

- **This approach requires the resource provider's native app to be an installed on the smart phone. It is assumed that that the provider's app is trusted.**
- **The client app sends the user to the provider's native app to perform the authentication and authorization stages.**



Type 3: Using the Resource Provider’s App

Type 3: Using the Resource Provider's App

- **Isolation is provided through the system message passing system.**

Type 3: Using the Resource Provider's App

- **Isolation is provided through the system message passing system.**
- **The main weakness of this approach is that:**
  - The user has to install the provider's app for each service provider they want to use this approach for.
  - Each provider app has a different OAuth flow which makes it difficult for the user to comprehend the OAuth stages.
  - In some cases a malicious app can impersonate the provider's app and in such case it can retrieve the user's username/password and can completely control the OAuth flow.

SDKs and Apps Study

SDKs and Apps Study

- **Major services providers offer software development kits (SDKs) that can be included in the mobile apps to seamlessly integrate them with their services.**

SDKs and Apps Study

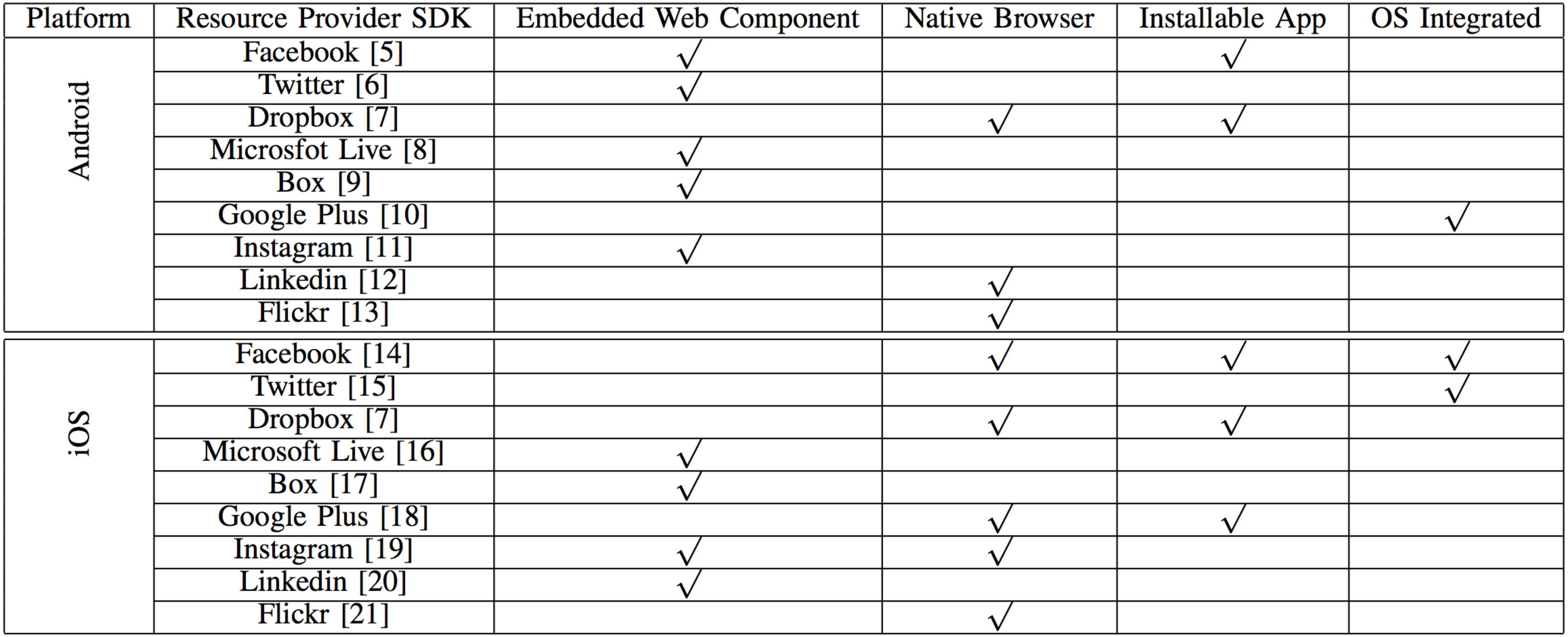
- **Major services providers offer software development kits (SDKs) that can be included in the mobile apps to seamlessly integrate them with their services.**

- **We conducted an empirical study on the current OAuth implementation trends followed by different service providers and by the OAuth development choices made by application developers.**

SDKs and Apps Study

- **Major services providers offer software development kits (SDKs) that can be included in the mobile apps to seamlessly integrate them with their services.**
- **We conducted an empirical study on the current OAuth implementation trends followed by different service providers and by the OAuth development choices made by application developers.**
- **In this study:**
  - We used 9 resource providers' SDKs.
  - We investigated the two most popular platforms (iOS and Android).
  - We downloaded, decompiled and analyzed:
    - 231 Facebook integrated apps.
    - 202 Dropbox integrated apps.

OAuth SDK Implementations

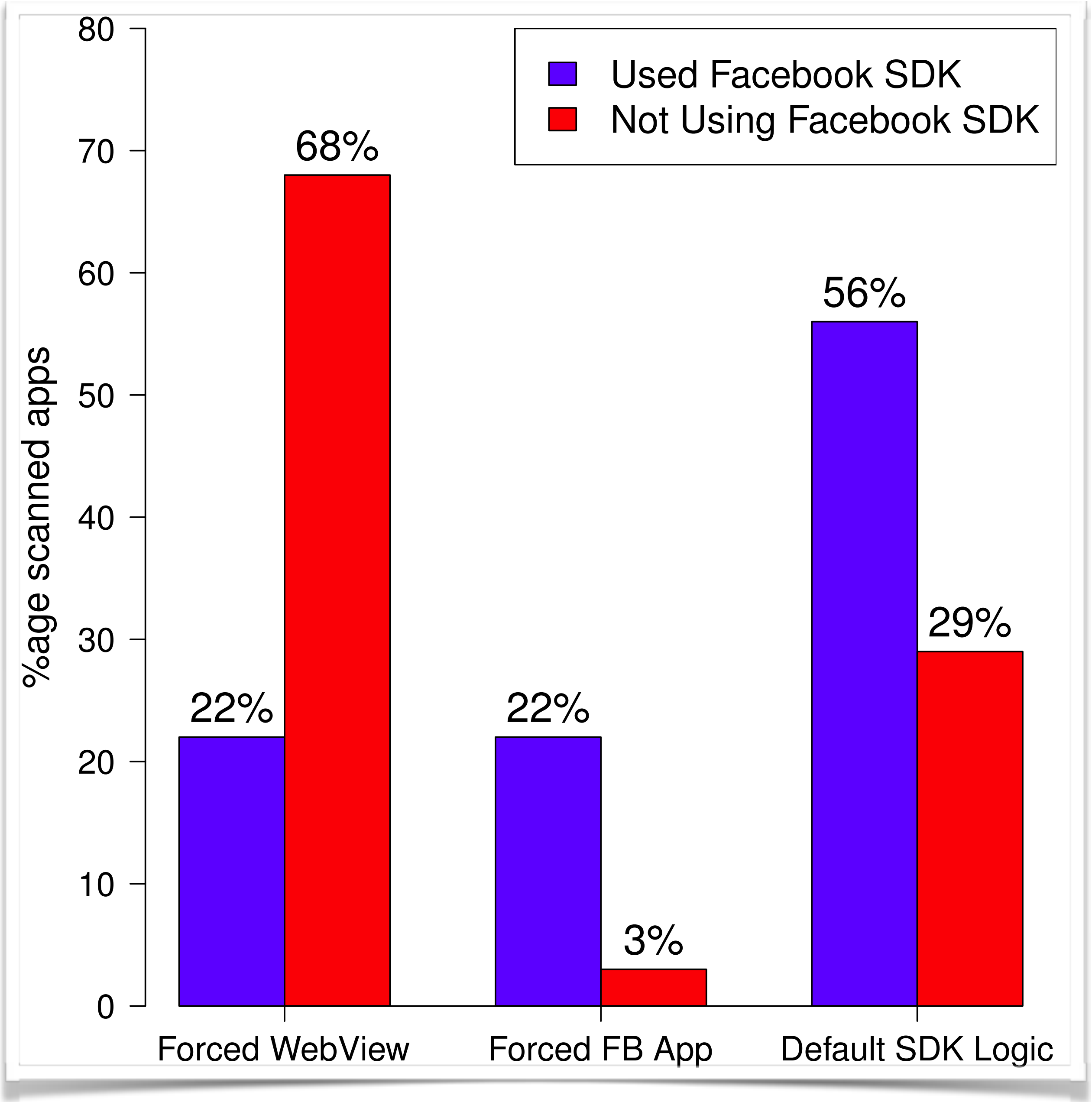


Platform	Resource Provider SDK	Embedded Web Component	Native Browser	Installable App	OS Integrated
Android	Facebook [5]	√		√	
	Twitter [6]	√			
	Dropbox [7]		√	√	
	Microsfot Live [8]	√			
	Box [9]	√			
	Google Plus [10]				√
	Instagram [11]	√			
	Linkedin [12]		√		
	Flickr [13]		√		
iOS	Facebook [14]		√	√	√
	Twitter [15]				√
	Dropbox [7]		√	√	
	Microsoft Live [16]	√			
	Box [17]	√			
	Google Plus [18]		√	√	
	Instagram [19]	√	√		
	Linkedin [20]	√			
	Flickr [21]		√		

OAuth SDKs and Authentication Models

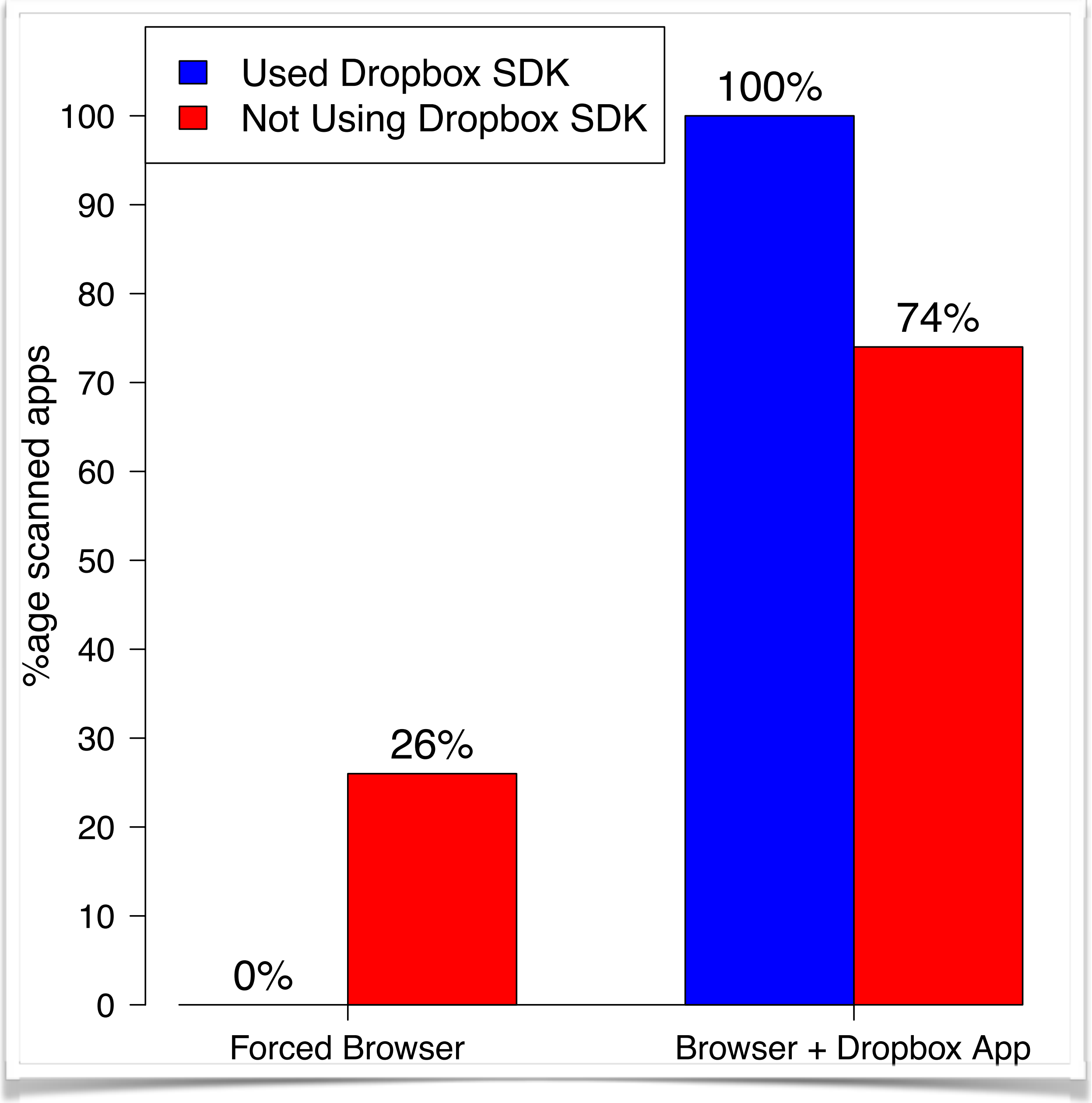
OAuth Implementation Stats

- **We downloaded and analyzed 231 Facebook integrated apps from Google Play.**



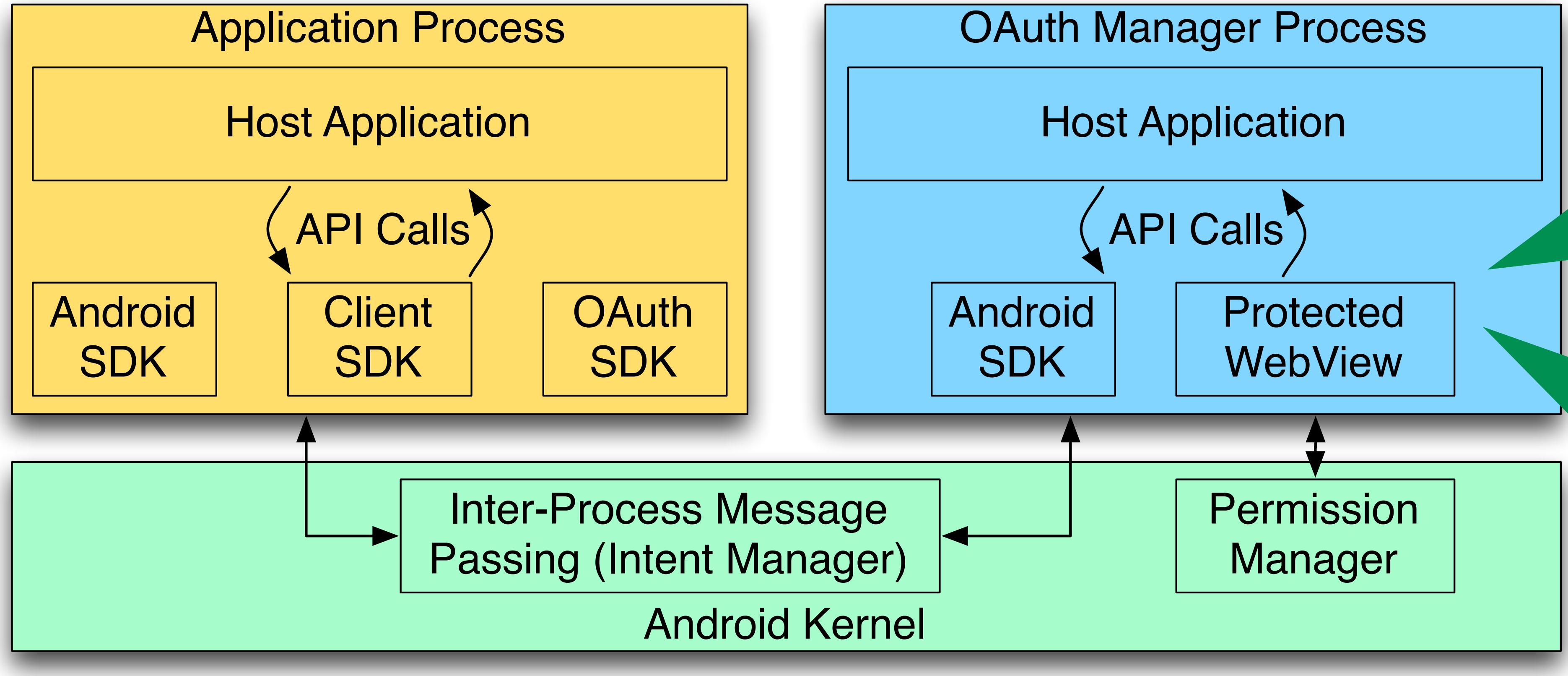
OAuth Implementation Stats

- **We downloaded and analyzed 202 Dropbox integrated apps from Google Play.**



Proposed Approach (*OAuth Manager*)

- **We propose to use the privilege separation concept to ensure that the client application has no control over the user-agent.**
- **Based on privilege separation, we removed the critical OAuth components and implemented it in a separate application (secure sandbox).**



Provides required isolation

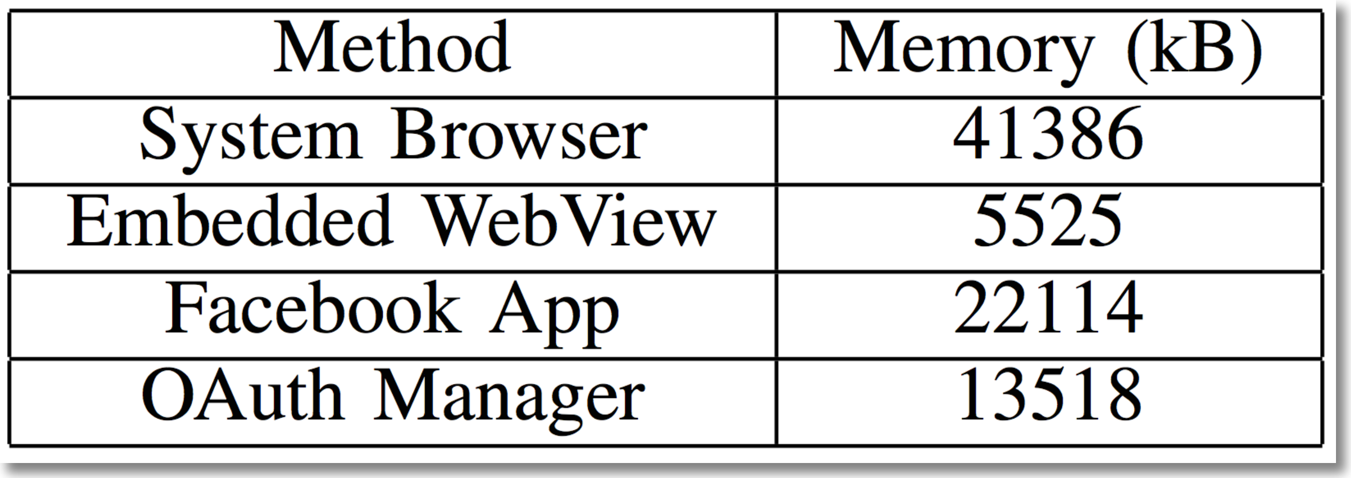
Detects impersonation attacks.

Proposed Approach (OAuth Manager)

- **We conducted performance analysis on our prototype, we measured the response time and the memory usage.**
- **We performed our experiments on a standard Android device, the Nexus S, that has android version 4.1.2, 1007.89 MB internal memory, 13624.34 MB SDCard, 343 MB RAM, system browser version 4.1.2-485486.**
- **We also analyzed the security of our framework:**
  - Detects impersonation attack.
  - Prevents from stealing and modification attacks.

OAuth Manager Memory Analysis

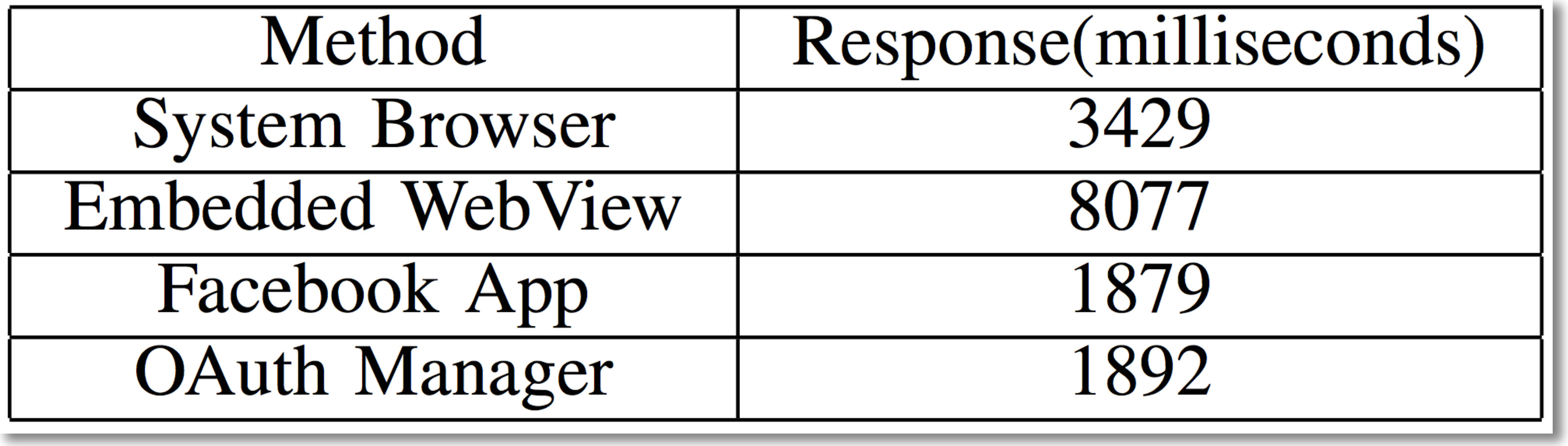
- **We used the Android Debug Bridge (adb) to measure memory overhead.**
- **We ran out test application multiple times and each time we used different authentication method. We recorded the memory consumption for each method (proportional set size).**



Method	Memory (kB)
System Browser	41386
Embedded WebView	5525
Facebook App	22114
OAuth Manager	13518

OAuth Manager Response Time Analysis

- **We performed benchmarking to estimate the overhead of OAuth Manager on displaying pages.**
- **We used Android Logging System, we added hooks to the code to record the time samples immediately after the user clicks the login button and promptly after successfully loading the authentication page.**



Method	Response(milliseconds)
System Browser	3429
Embedded WebView	8077
Facebook App	1879
OAuth Manager	1892

OAuth Manager Security Analysis

- The OAuth flow based on OAuth Manager is more secure than the other flows, it provides the measures to prevent from the aforementioned attacks.
- It isolates the user-agent and the client apps. It provides a secure WebView that is not accessible to the client app.
- It detects impersonation attacks by scanning the installed packages and detecting possible malicious registered handlers.

Conclusion and Future Work

- **Conclusion:**
  - We described the design and security assumptions of each of the main OAuth implementations in smart phone apps.
  - We demonstrated the attacks that can be performed on the different implementations and discussed their effects.
  - We conducted an empirical study on the current OAuth implementation trends followed by the service providers and by the OAuth development choices made by app developers.
  - We proposed an application-based OAuth Manager framework, that provides a secure, light, and fast OAuth flow.

- **Future Work:**
  - Investigate OAuth management at the OS or Core library levels.
  - Investigate methods to enhance the awareness of secure OAuth implementation and usage.

Thank You.

mshehab@uncc.edu