Security of Social Media APIs

Antti Nuopponen
Nixu

OWASP

The OWASP Foundation
http://www.owasp.org

Agenda

- Short Introduction to Social Media APIs
- Facebook Connect
  - End user experience
  - Security aspects
  - Implementation Pitfalls
  - Privacy Issues

Social Media APIs

- Several APIs available for developers
  - Idea is to allow 3rd party developers to create applications / web pages that interact with social media sites
- Common use scenarios
  - Widget / social mashup – light weight applications that run inside social media sites
  - Social application – user interface runs inside social media sites but functionality relies on external servers
  - External web sites – runs completely in external servers but interact with social media sites to get access to users' data
- This presentation focuses on the security aspects of APIs for external web site integration:
  - Facebook Connect, (Open Social, OpenID)

What Social Media APIs Provide

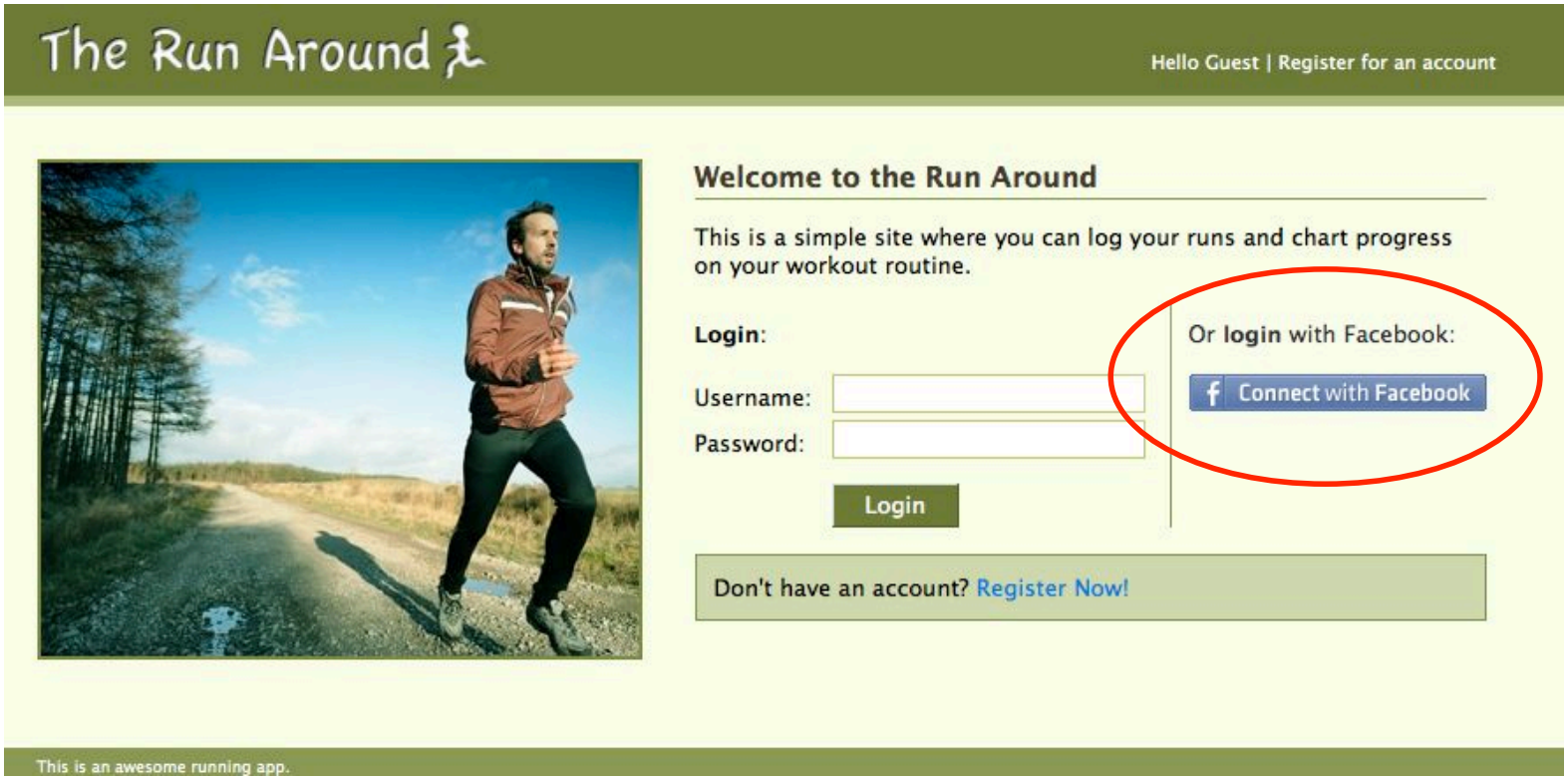
- Ability to federate user's identity from social media sites
  - I.e. authenticate users using users' existing social media site authentication
- Access to users' information in social media sites
  - Name, contact information, friends, posts, photos etc.
- Ability to post information to social media sites on user's behalf
  - Activities, comments, photos etc.
- Examples of existing integrations
  - ABC, NBC, XBox Live, TechCrunch, and many more

Facebook Connect

- Facebook's API for external web site integration
  - User authentication via Facebook
  - Access to user's Facebook profile
- Proprietary solution
- JavaScript based approach, but closely integrated with Facebook's other APIs
  - Documentation incomplete
  - Under constant development
  - Makes it harder to understand and use

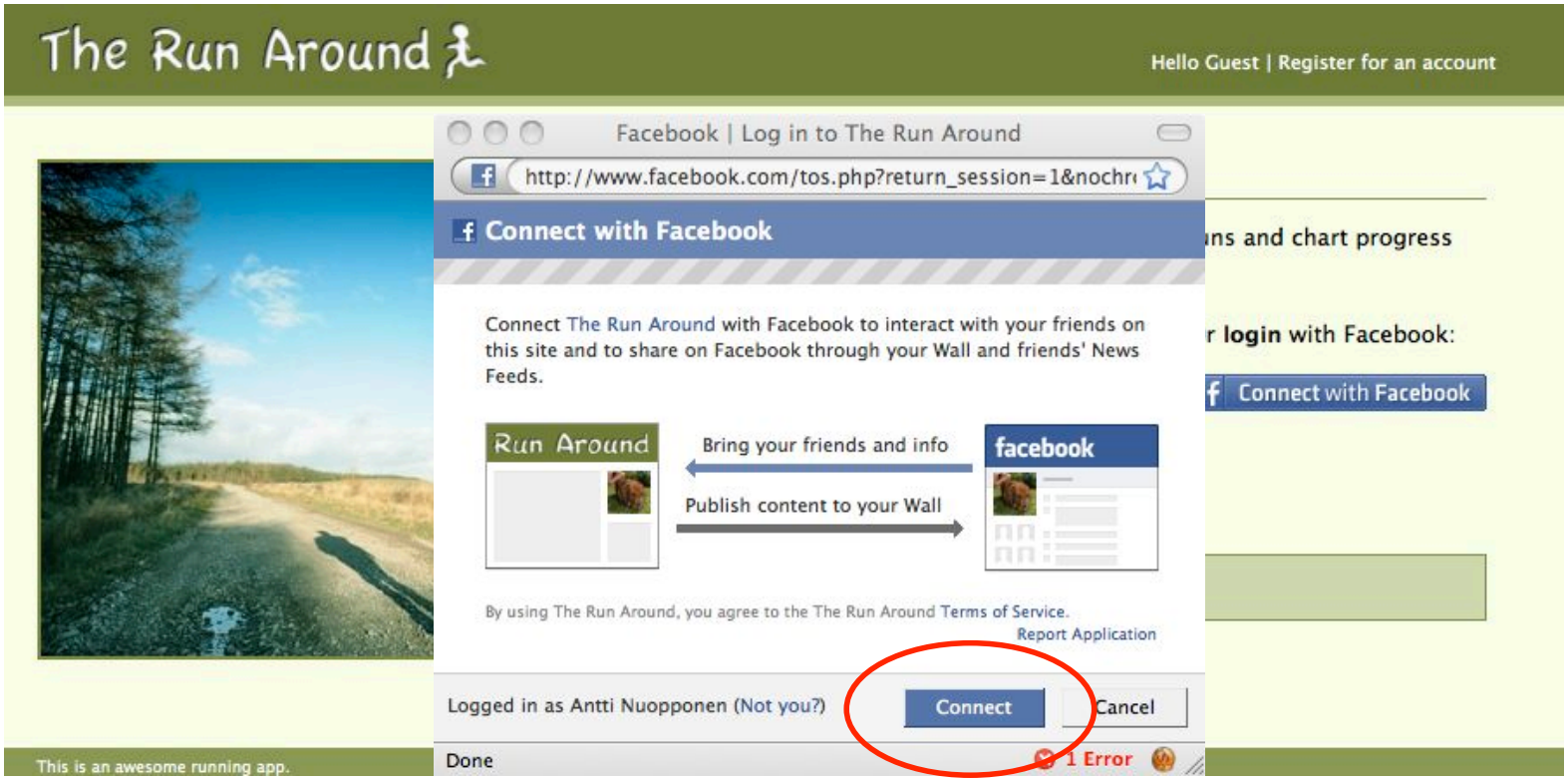
FB Connect - End User Experience (1)

- A "Connect with Facebook" – button is shown to user as alternative login / registration method



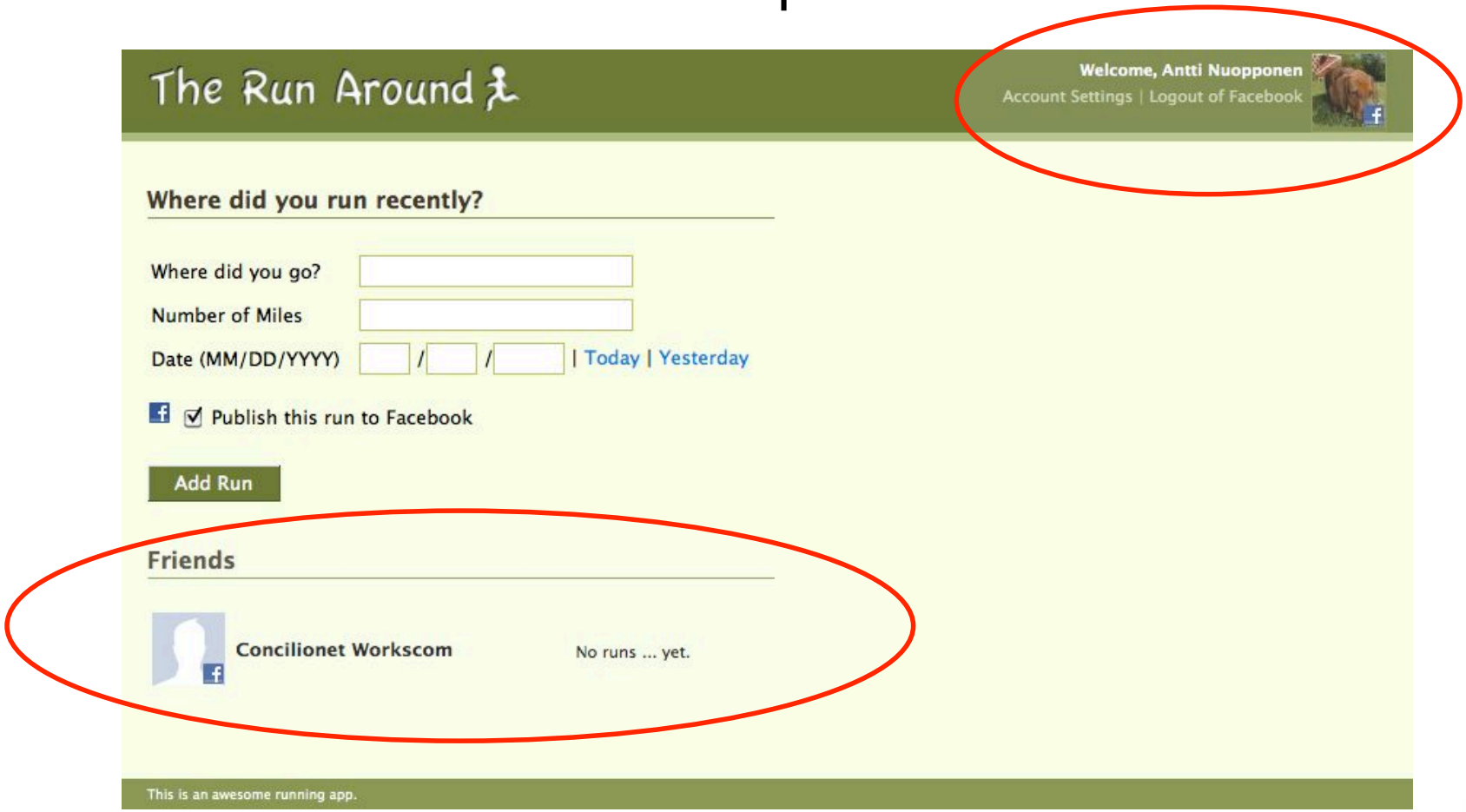
FB Connect - End User Experience (2)

- When user clicks the button a popup dialog asking permissions is displayed
- The dialog is loaded from Facebook server



FB Connect - End User Experience (3)

- Once user clicks the "Connect" button the website gets access to user's Facebook profile



Setting Up Facebook Connect

- An application is created inside Facebook
  - Provides the identity inside Facebook for the website (application ID, API key and API secret)
  - Used to tell Facebook where the site lives
- Loading of Facebook JavaScript API and cross domain communication receiver is added to the web site
  - JavaScript API is initialize with the API key
  - Facebook uses the API key to identify the correct application
- Messages between Facebook and your site travel via user's browser

What Actually Happens Behind Scene?

1. User clicks the "Connect with Facebook" button
   - This triggers loading of the permission popup dialog from Facebook site
2. When user clicks the "Connect" button the dialog is closed and Facebook Connect JavaScript library makes a callback to website's cross domain receiver
3. The cross domain receiver loads JavaScript from Facebook and redirects user's browser to original website
   - At this point the site has access to user's Facebook profile

   → Site can get user's information

Security Building Blocks

- User authentication uses Facebook's normal user authentication and session management
  - User needs to be logged in to Facebook in order for the Connect to work
  - If user is not logged in the Connect procedure asks user to log in first
- External website is authenticated using Facebook application that provides shared secret for Facebook and the website
- External website can authenticate information that comes from Facebook using the shared secret

Security – Web Site Perspective

- It is JavaScript – how can we trust it?
- The are two cases:
  1. If web site does not store and later access data from Facebook Connect there is no need for trust (not usually the case).
  2. If web site stores data coming from Facebook Connect the web server MUST authenticate the data
- Authentication is done with cookies that the JavaScript API library sets in user's browser
  - For all replies Facebook calculates a keyed MD5 hash with the application secret
  - The JavaScript API library gets these replies and sets the values to cookies for the web site
  - Web site gets these cookies and can verify data by calculating the same keyed hash

Security – Web Site Perspective (2)

- From the web site perspective trusting data coming through Facebook Connect comes down to:
  - Trusting Facebook user authentication and management (e.g. no two users have the same user id)
  - Trusting keyed MD5 based authentication of data
    - Shared secret used as the key is 128bits long
- Things to consider
  - Facebook session lifetime is very long – would this be a problem for your users?
  - When users are logged in to Facebook they will be automatically logged in to your site when they come to it

Facebook Connect – Implementation Pitfalls

- It is easy to build the integration in a way that information coming from Facebook is not verified
  - Use of JavaScript API alone will give you the end user functionality but no security!
  - Server must authenticate the information that is set with cookies
  - JavaScript API provides functions to check user's login status.
    - For example "*ifUserConnected*" function redirects users to different pages depending on their login status
    - If the page for logged in users does not properly verify user's credentials from cookies set by Facebook

      → a backdoor to the system is created.
- Handling of the secret key
  - Trying to implement signature verification in JavaScript
  - Keeping it in a place that can be accessed from outside

Privacy Issues – End User Perspective

- What happens when user grants 3rd party applications or sites access to his/hers social media site profile?
- In Facebook granted privileges define what is seen
- With "*read_stream*" permission 3rd party applications can see all posts that appear in user's wall
  - This includes posts of user's friends
  - User's friend can not know if their friend has given access to 3rd party applications
  - → Popular applications like FarmVille have access to posts of millions of users
  - → This means that popular application can have access to post of hundreds of millions of people without them knowing about it
- If application gets "*offline_access*" they can read user's information even if user is logged out from Facebook

Thank you!