Using ArcGIS Online App Logins in Node.js

James Tedrick
ArcGIS Online – OAuth Logins

- Differentiates between application server, authentication server
- Authentication server logs user in, checks for user acceptance of application
- Application server *does not* see username/password as they are entered
- Application does get access token after authorization
ArcGIS Online – OAuth Logins

- **User Logins** – login to a user’s account
  - User must have ArcGIS Online login
  - Access user’s items
  - Use user’s organization credits for tasks
  - Web App, Mobile Device, Server login

- **Application Logins** – login with your (developer) account
  - User’s don’t need to have ArcGIS Online login (but you need to control access)
  - Access your items
  - Use your credits for tasks
  - Server login only
OAuth login key properties

- `redirect_uri` – resource to load when presenting new credentials
- `Client ID / appId` – unique ID of application in ArcGIS Online
- `Client Secret / appSecret` – secret key used with appId (appId’s ‘password’)

![OAuth login key properties](image_url)
Why Server Logins?

- Your application provides value via custom server-side resources
  - Access private data via your server
  - Apply custom analysis to user’s data
- You want to manage the user relationship
  - Use your credits for tasks for customers
  - Consolidate customer billing with you
Node – What? & Why?

http://nodejs.org

- Node.js – Use a JavaScript runtime as a server application environment
- Familiar language – like my web apps
- Asynchronous programming pattern – like my web apps
- Deployable in a variety of contexts (server, cloud)
- Strong developer community
- Many additional modules with easy install/update
App Logins
**Application Login**

1. Application requests authorization by opening [https://www.arcgis.com/sharing/oauth2/token](https://www.arcgis.com/sharing/oauth2/token) (normally done at server startup)
2. ArcGIS Online provides a token
3. Application loads into client
4. Client requests an operation that makes use of AGO resources
5. Server requests resources with token
6. ArcGIS Online provides response
7. Possible further processing; response is delivered to client

**Diagram:**

- Step 1: Application requests authorization
- Step 2: ArcGIS Online provides a token
- Step 3: Application loads into client
- Step 4: Client requests an operation
- Step 5: Server requests resources
- Step 6: ArcGIS Online provides response
- Step 7: Possible further processing
App Logins
User Logins
User Login – Server Applications

1. Application loads into client
3. ArcGIS Online redirects to organization login page
4. User logs in using login system
5. Login system redirects browser, providing authorization code as uri parameter
6. Server gets authorization code from uri
7. Application requests access token from AGO https://www.arcgis.com/sharing/outh2/token
8. AGO provides token

Your application server

ArcGIS Online

Identity Management
User Logins
Resources

• This demo: https://github.com/tedrick/agoServerLoginExample
• ArcGIS for Developer OAuth documentation: https://developers.arcgis.com/authentication/index.html
• Check language specific documentation at http://developers.arcgis.com

• Node.js – http://nodjs.org
  - Request module (HTTP) – https://github.com/mikeal/request
  - iisnode: Run node apps via IIS – https://github.com/tjanczuk/iisnode
Thank you…

• Please fill out the session survey:

   Offering ID:  2134

   Online – www.esri.com/ucsessionsurveys

• Related Sessions (Thursday):
  - ArcGIS Online: Security Aspects
    10:00 – 10:30 General Theater 1, Hall A
  - Securing your Portal for ArcGIS
    1:30 – 2:45 Room 4