

Feedback: <http://goo.gl/DpUBh>
#io2011 #TechTalk

Identity and Data Access: OpenID & OAuth

Ryan Boyd @ryguyrg

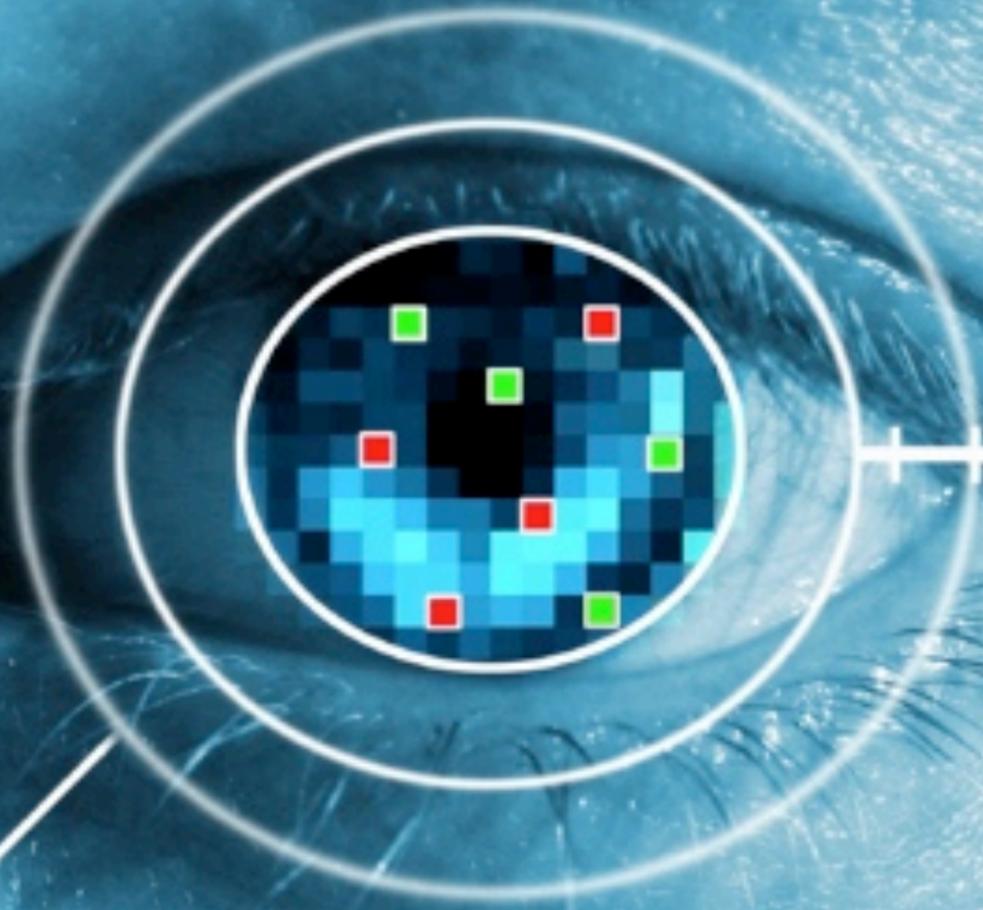
<https://profiles.google.com/ryanboyd>

May 11th 2011

Agenda

- 1 Terminology
- 2 OpenID and the Google Identity Toolkit
- 3 Mobile Authentication
- 4 OAuth for Individuals
- 5 OAuth for Businesses
- 6 The Future!
- 7 Resources and Q&A

Terminology



Authentication

Authorization





Bob

Administrative Assistant
@ Acme Corporation



http://mail.acme.com



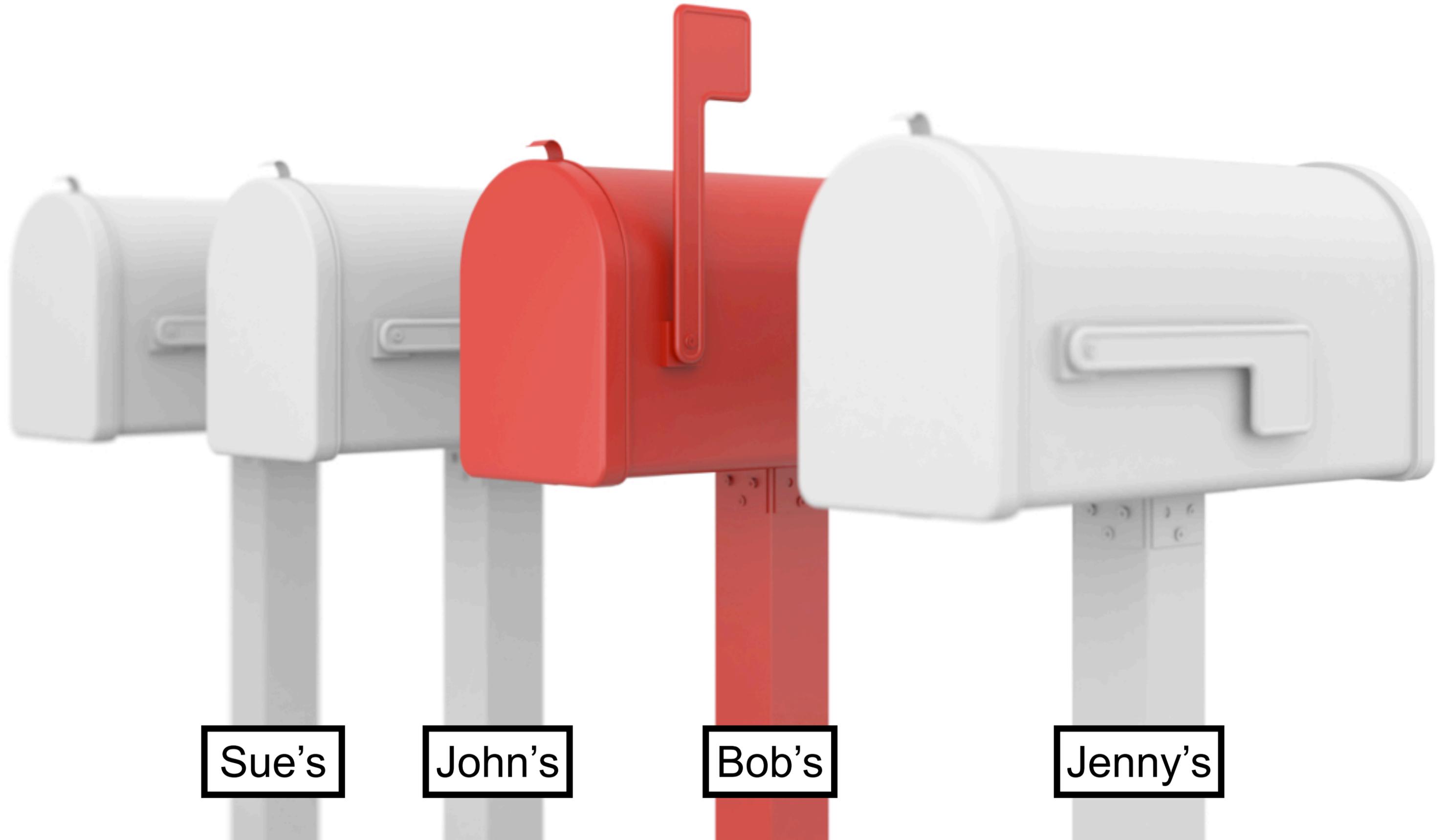
Login to your Acme Corp account

Username:

bob

Password:

Login



Sue's

John's

Bob's

Jenny's

Bob is
Authenticated
to his Acme Corp
account and has
Authorized
access to his
mailbox

Authentication and Authorization in Context

- 1 **OpenID** for **Authenticating** a user visiting a web site
- 2 **OAuth** for **getting Authorized access** to a user's data stored elsewhere

Authenticating users via OpenID

OpenID: Terminology

Identity Providers (IdP)

Department of Motor Vehicles

The image shows the exterior of a Department of Motor Vehicles building. The building is constructed from light-colored, rectangular stone blocks. A large sign on the upper part of the building reads "Department of Motor Vehicles" in a bold, blue, sans-serif font. The sign is mounted on a dark horizontal band. In the foreground, there is a paved walkway, some green landscaping with tall grasses, and a black metal railing. In the background, there are mountains under a clear blue sky. A tall, thin silver pole stands near the building.



Relying Parties (RP)

How does Federated Identity Work?

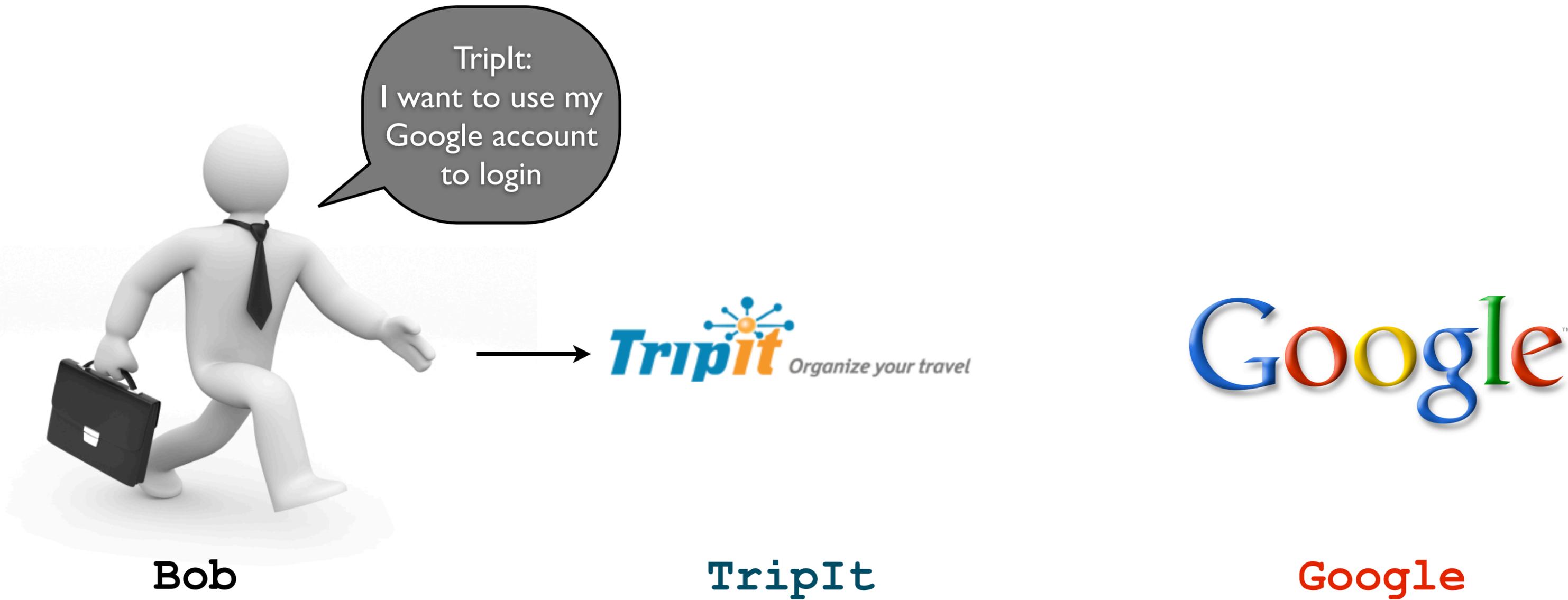


Bob

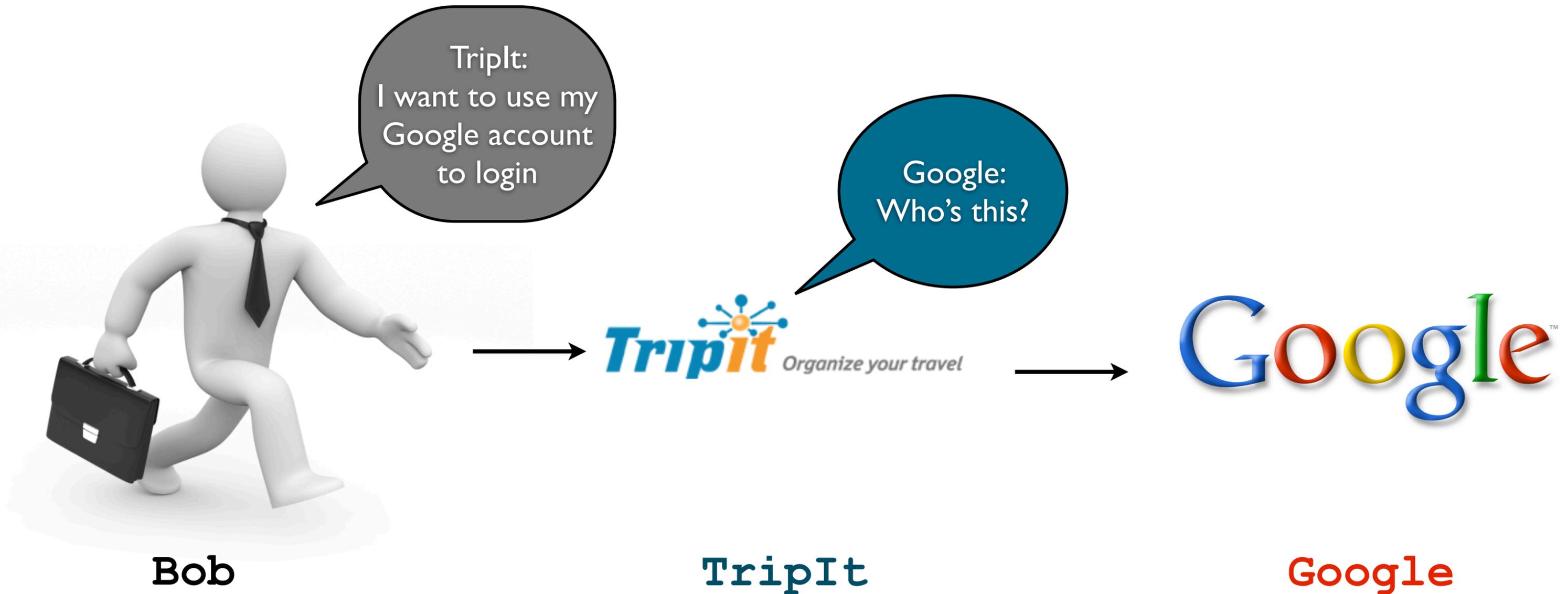


TripIt

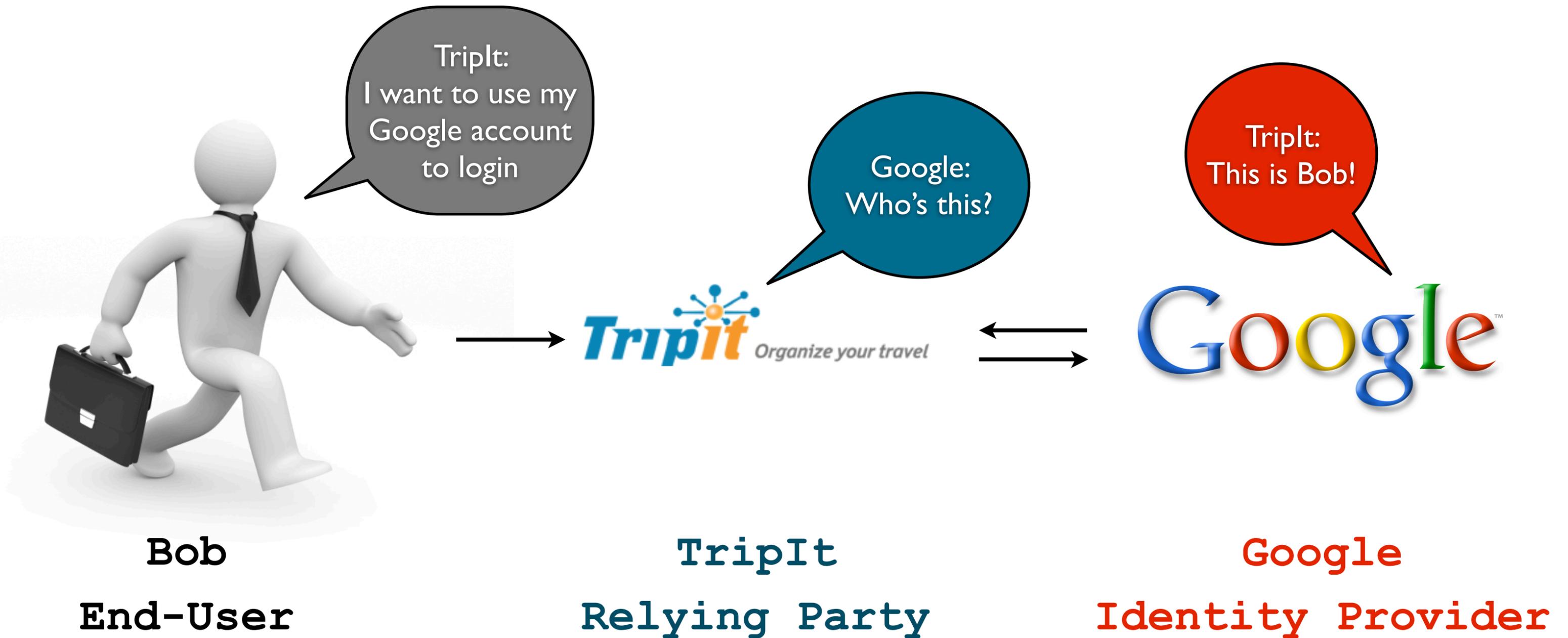
How does Federated Identity Work?



How does Federated Identity Work?



How does Federated Identity Work?



S U C C E S S !!

Bob is

Authenticated

to his

TripIt account

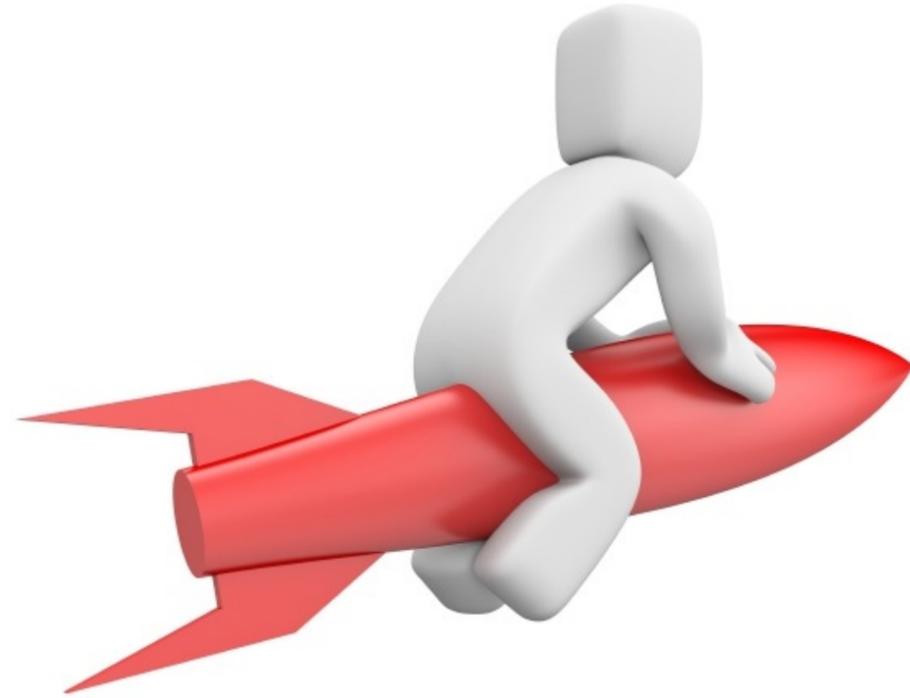
using his

Google identity

OpenID: Why?

Users can login to all sites

using their existing accounts



"OpenID is a **safe, faster, and easier** way to log in to web sites."

openid.net

**Faster
and
Easier**

Traditional Signup Form

The image shows a screenshot of a web form for 'SaaSy Payroll'. The header features the company name 'SaaSy Payroll' in a large, white, sans-serif font against a dark brown background, with the tagline 'your payroll solution, your way' in a smaller, orange font below it. A prominent red button with the text 'Login' is positioned to the left of the main form area. The main form itself is titled 'New Customer Form' and contains several input fields: 'First Name' with the value 'Ryan', 'Last Name' with 'Boyd', 'E-mail address' with 'ryan@ryguy.com', 'Password' (masked with dots), and 'Confirm Password' (also masked with dots). A 'Signup' button is located at the bottom left of the form. The entire form is set against a light gray background.

SaaSy Payroll
your payroll solution, your way

Login

New Customer Form

First Name:

Last Name:

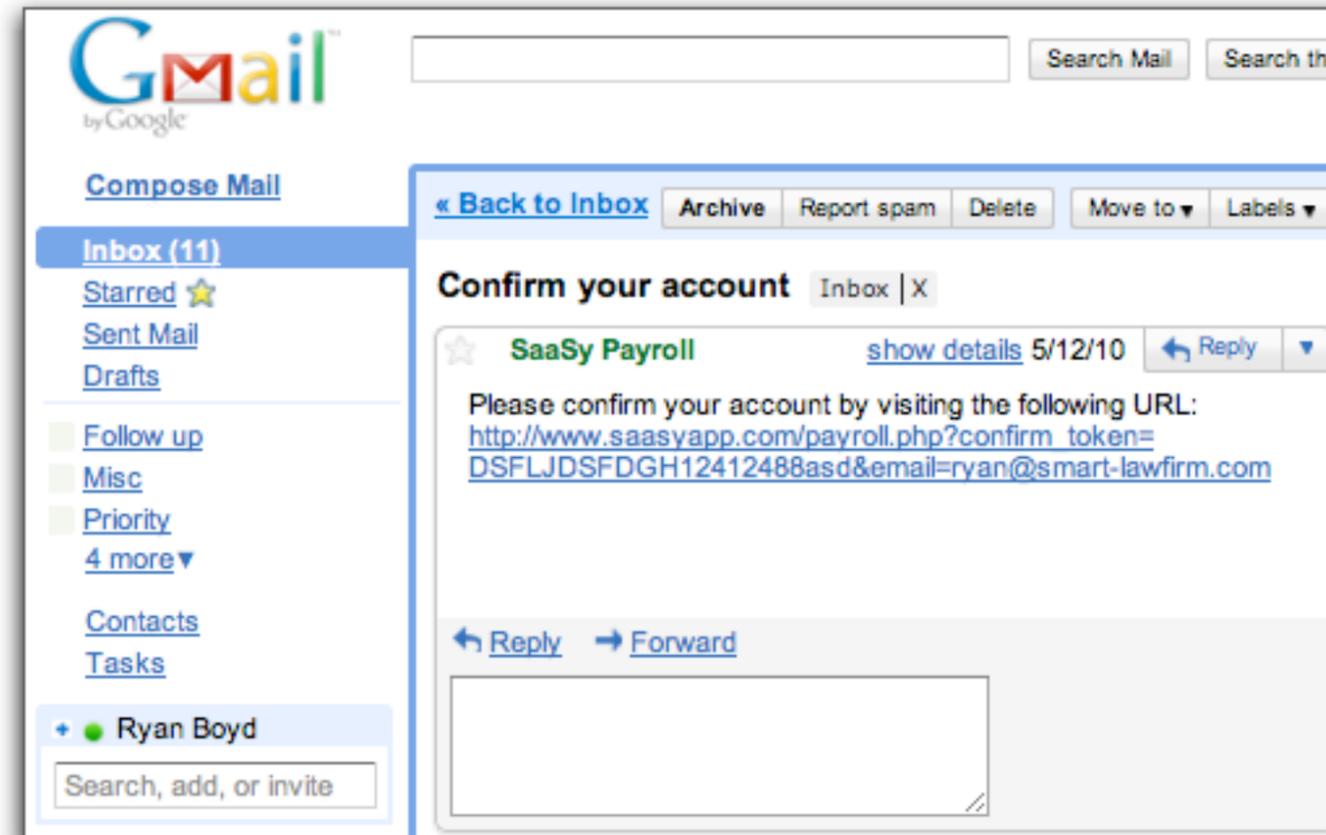
E-mail address:

Password:

Confirm Password:

Signup

Traditional Signup Form - e-mail confirmation

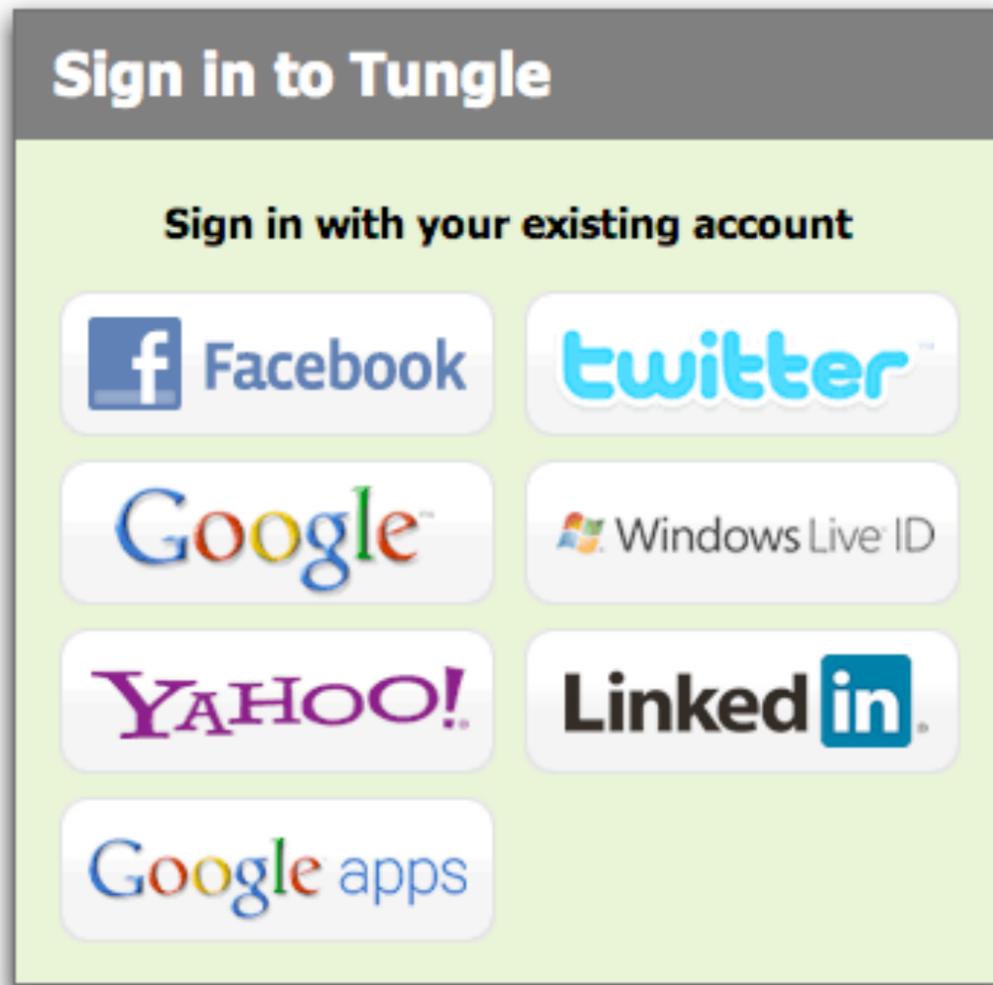


50 keystrokes

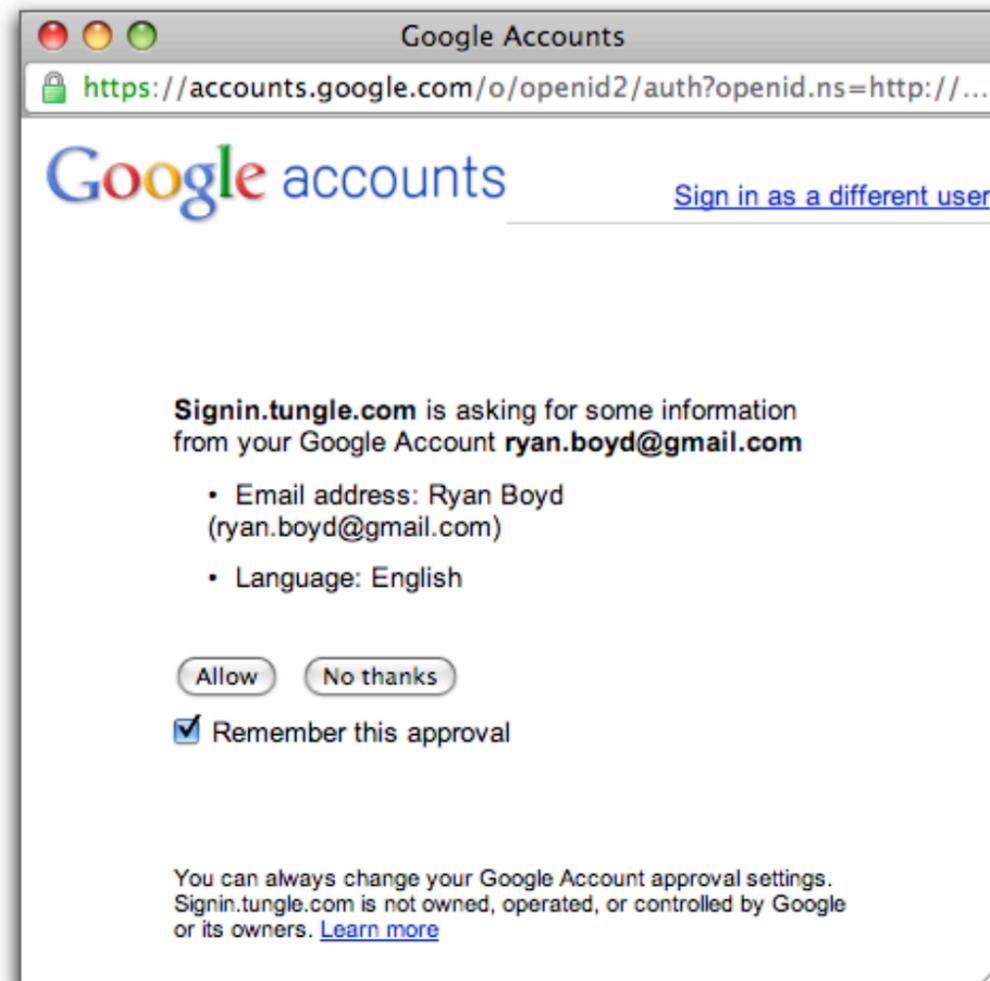
3 mouse clicks



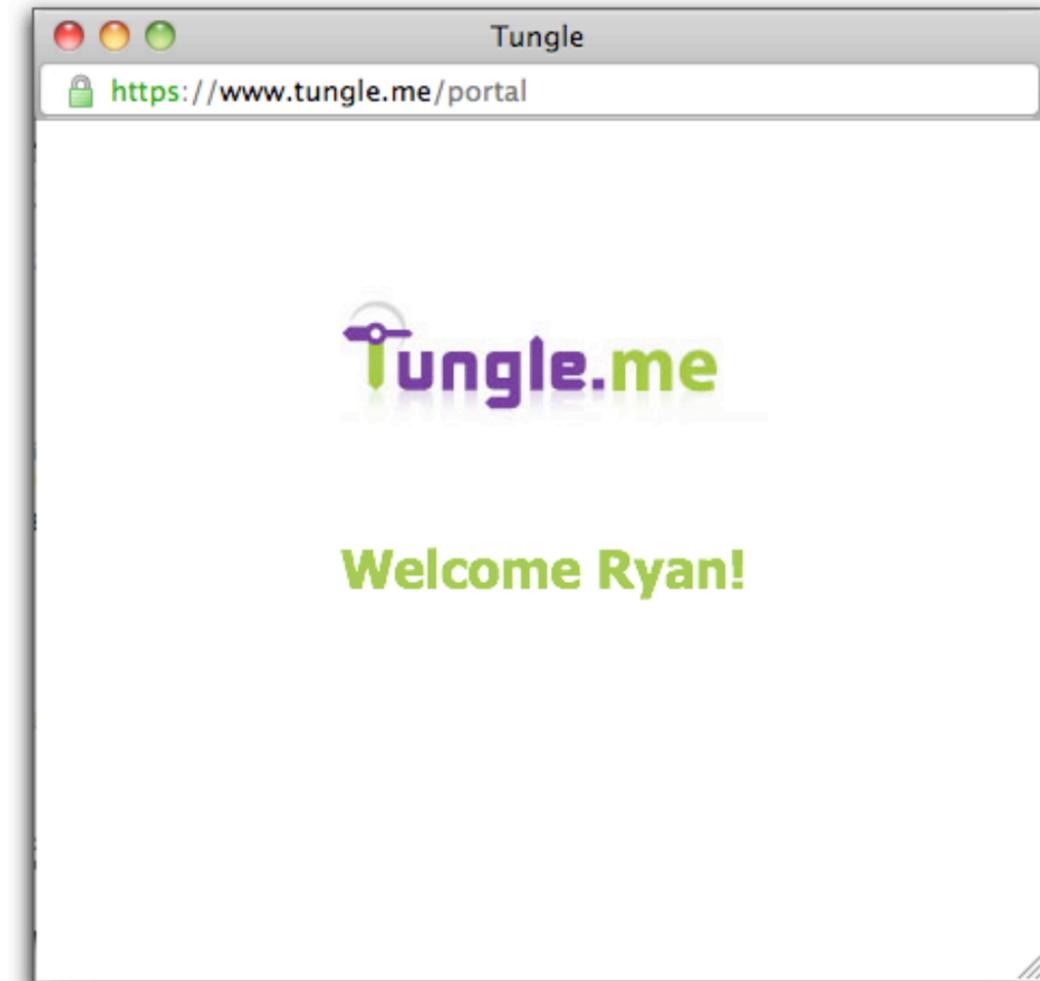
Improved UX with OpenID



1



2



3

0 keystrokes

2 mouse clicks



Safer

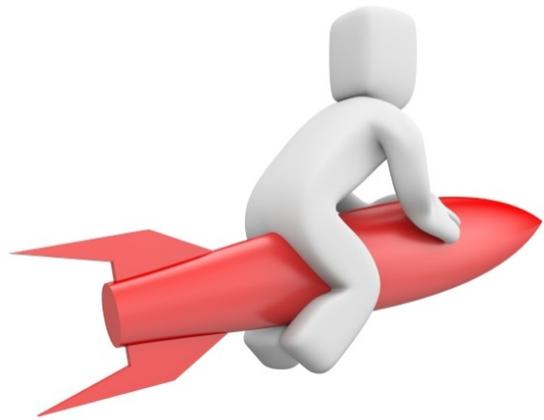
Ways OpenID is Safer

- ① One username and password
- ② Password can be ultra-secure
- ③ Password is only provided to Identity Provider
- ④ Two-factor auth and other protections

Safer, Faster and Easier



The user only provides their ultra-secure username/password to their identity provider



The user is often already logged into their identity provider



The user doesn't need to create, maintain and enter a password on every site

OpenID: becoming a Relying Party

OpenID is easy to implement

But not easy ENOUGH

Why? Edge Cases!

- Existing user wants to switch to using OpenID
- OpenID user wants to switch back to a password
- User changes their e-mail address
 - New address matches another account
- Handling deleted/suspended accounts

See <http://www.openidsamplestore.com/>

Introducing . . .

Google Identity Toolkit

Google Identity Toolkit

Sign in to DemoRP Account Connect with any account

Email

Password

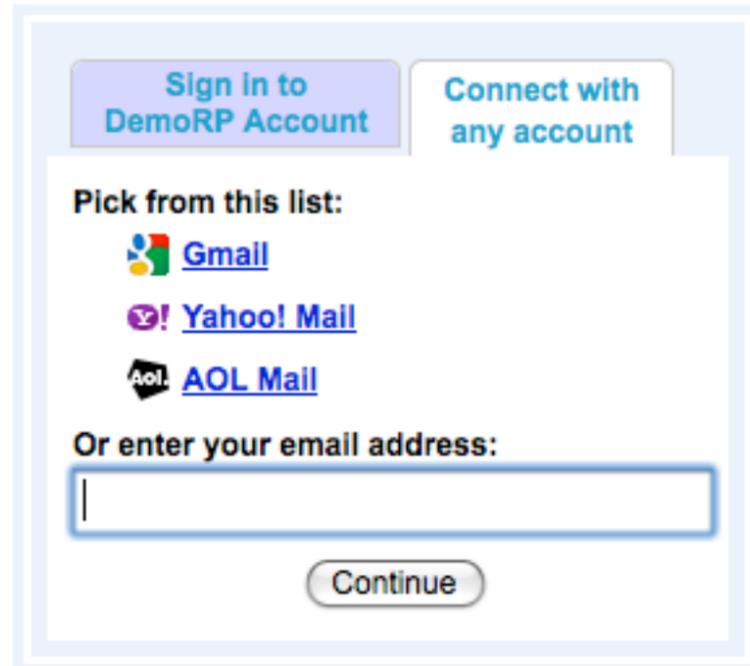
Stay signed In

Sign In

[Can't access your account?](#)

Don't have a DemoRP account?
[Create an Account Now »](#)

Google Identity Toolkit



The screenshot shows a login interface for a 'DemoRP Account'. At the top, there are two buttons: 'Sign in to DemoRP Account' and 'Connect with any account'. Below these, the text 'Pick from this list:' is followed by three links with icons: 'Gmail', 'Yahoo! Mail', and 'AOL Mail'. Underneath, the text 'Or enter your email address:' is followed by an empty text input field. At the bottom of the form is a 'Continue' button.

Provides:

- JavaScript UI Widgets
- Client Libraries
- Code on Google's servers

Supports:

- Signup and/or Login
- Multiple Identity Providers:
 - Gmail (including Google Apps)
 - AOL Mail
 - Yahoo Mail
 - Hotmail



Demo RP Site using GIT



Productivity for developers, performance for users

GIT provides developers with a powerful set of API endpoints to do federated login, attribute exchanges. This allows desktop, mobile, and web application developers to integrate with different identity provider, and facilitate the user to sign-up and sign-in on different sites.



[Try the Demo](#)

This site supports federated login by leveraging the GIT. Try federated login to this site by the 2-tab login widget on the right.

You can also try the [Create Account Wizard](#) to import attributes from IDP.



[Get Started](#)

Walk through the first steps needed to integrate GIT into your application. From there, work through the fundamentals of GIT development with an in-depth tutorial.



[Read the Docs](#)

Everything you need to know about how to use GIT.

Sign in to
DemoSite account

Connect with
any account

Email

Password

Stay signed In

Sign In

[Can't access your account?](#)

Don't have a DemoSite account?

[Create an Account Now >](#)

Authenticating Users on Mobile Devices



Allow users to create a 'Mobile Password'

Welcome To Concur for Mobile

Set PIN



To log in to Concur on your mobile device, you must enter your User Name and a Mobile PIN. Your User Name is **rboyd**. If you have not yet set up or need to change a PIN, enter a PIN (which can include letters, numbers, and special characters such as !,\$, or # but no spaces) in the fields below.

Mobile PIN

Verify Mobile PIN

Create/Update Mobile PIN

Generate a 'Mobile Password'

Application-specific passwords

Some mobile or desktop applications that work outside of a browser aren't yet compatible with 2-step verification. These applications are hard-coded to ask for a username and password, and do not prompt for a verification code. If you want one of these applications to access your Google Account, you must enter an **application-specific password**, not your Google Account password, when asked for a password. [Learn more](#)

Application-specific password generated

You may now enter your new application-specific password into your application. For security reasons, it will not be displayed again:

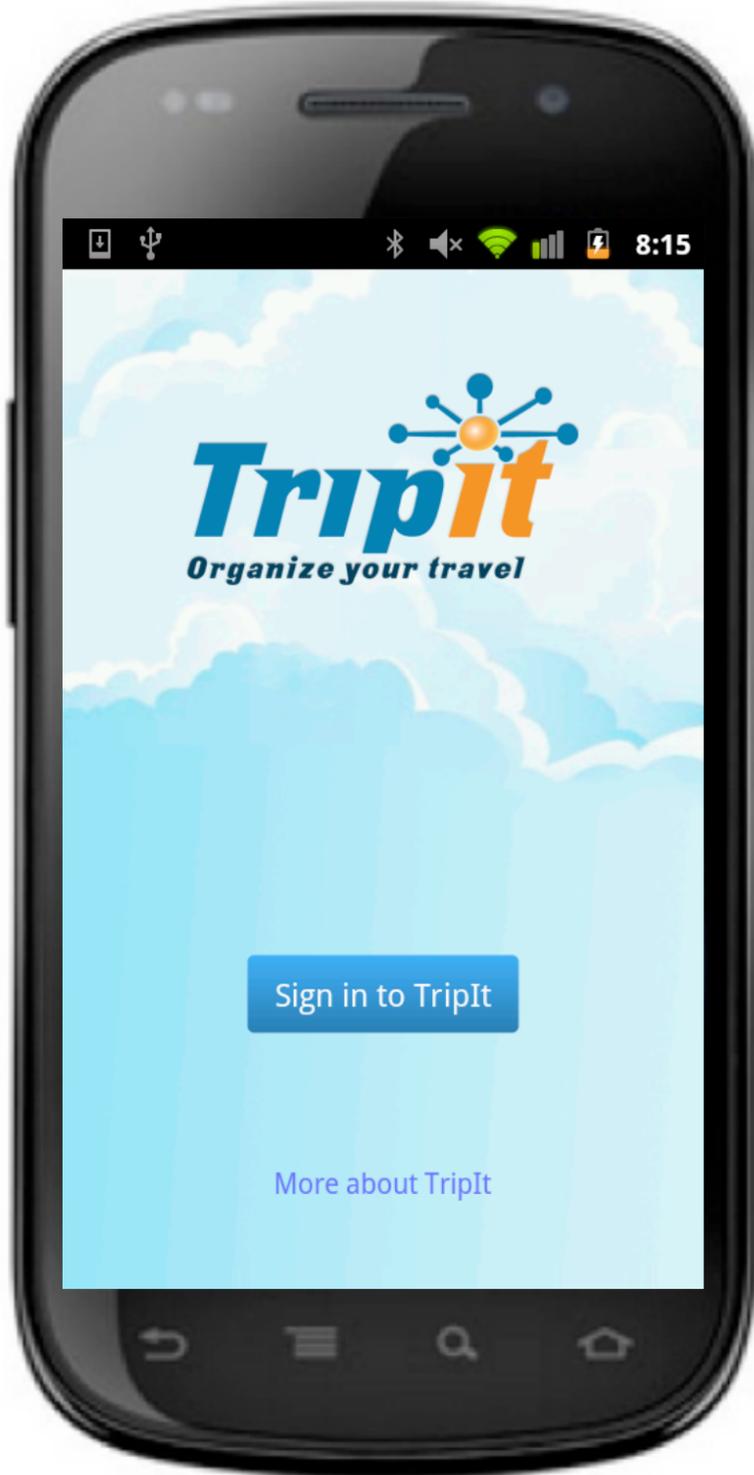
ldgi imqq qcaz cusq

Spaces don't matter.

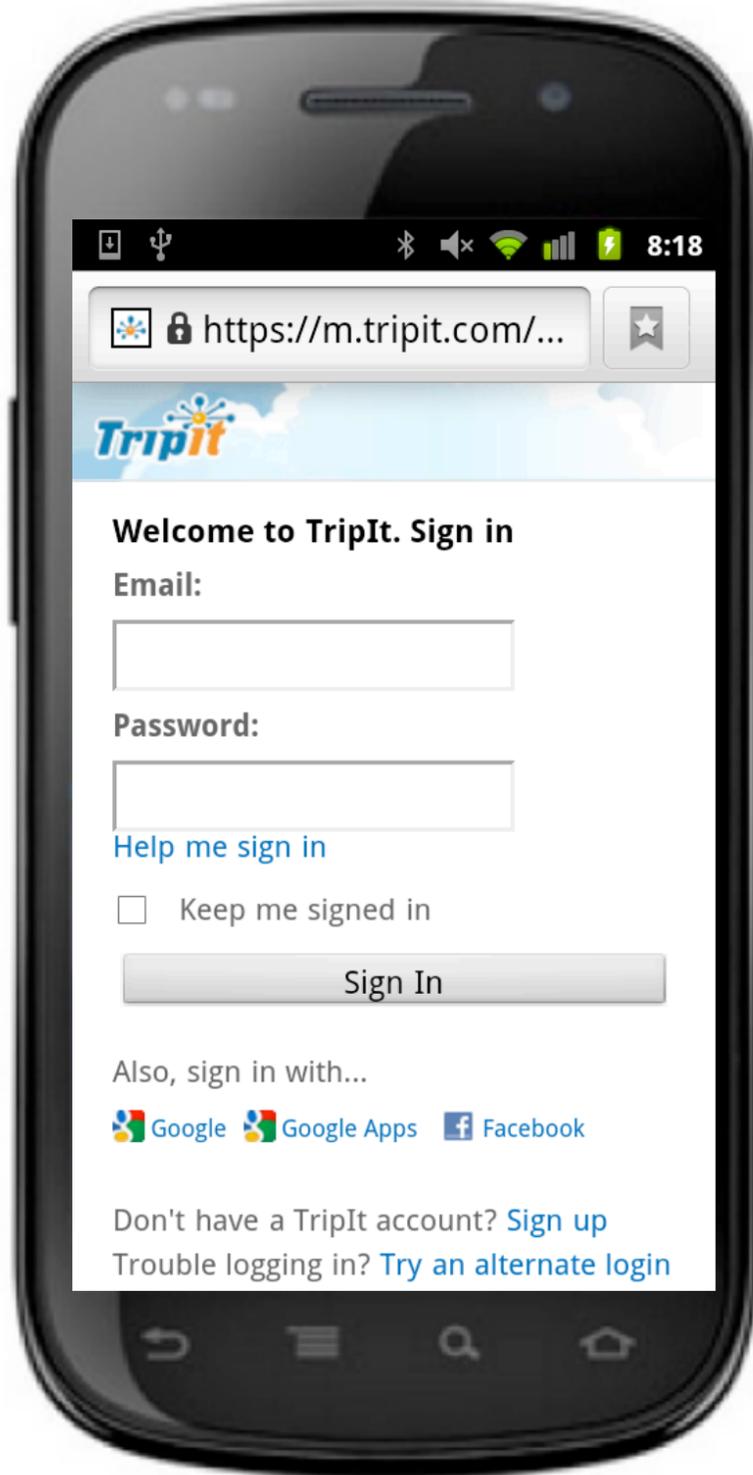
You should need to enter this password only once - no need to memorize it.

Hide password

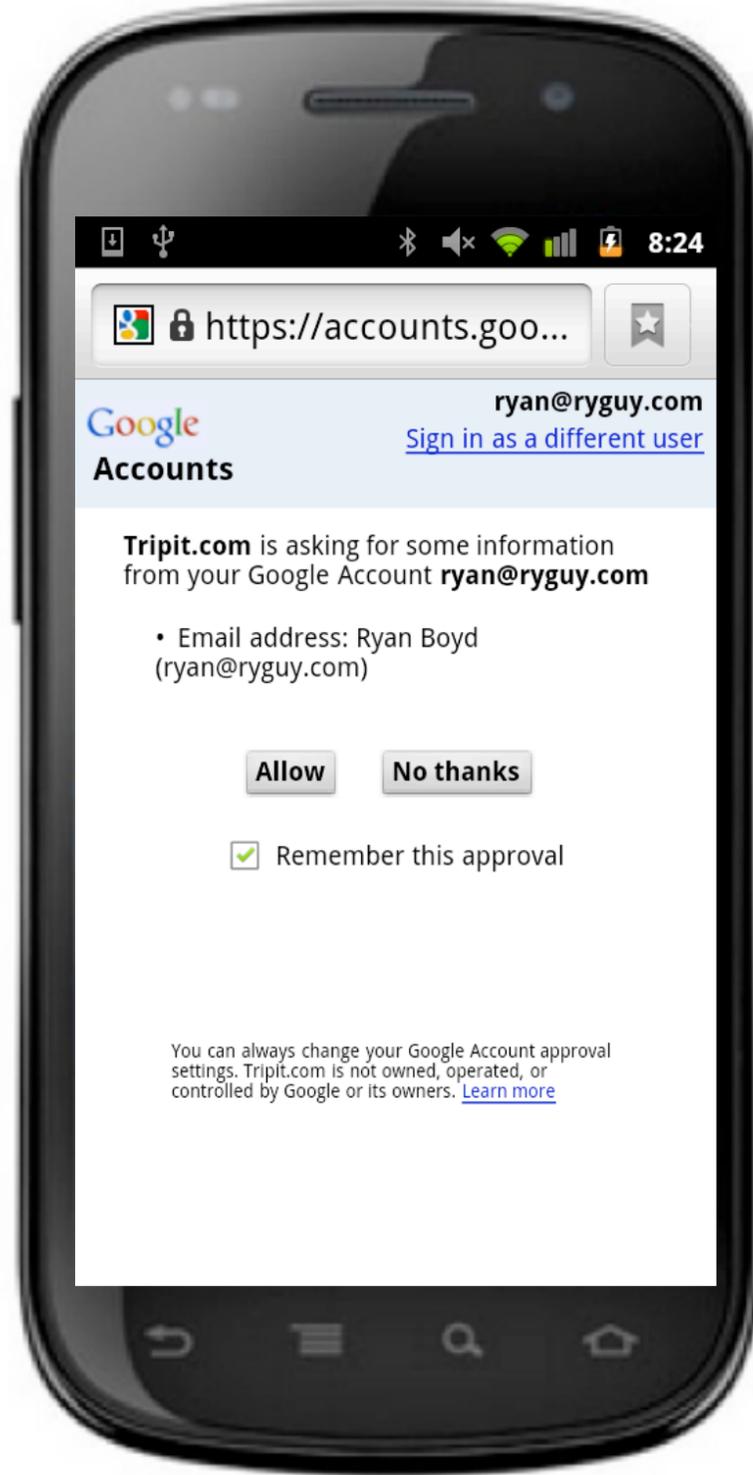
OpenID!



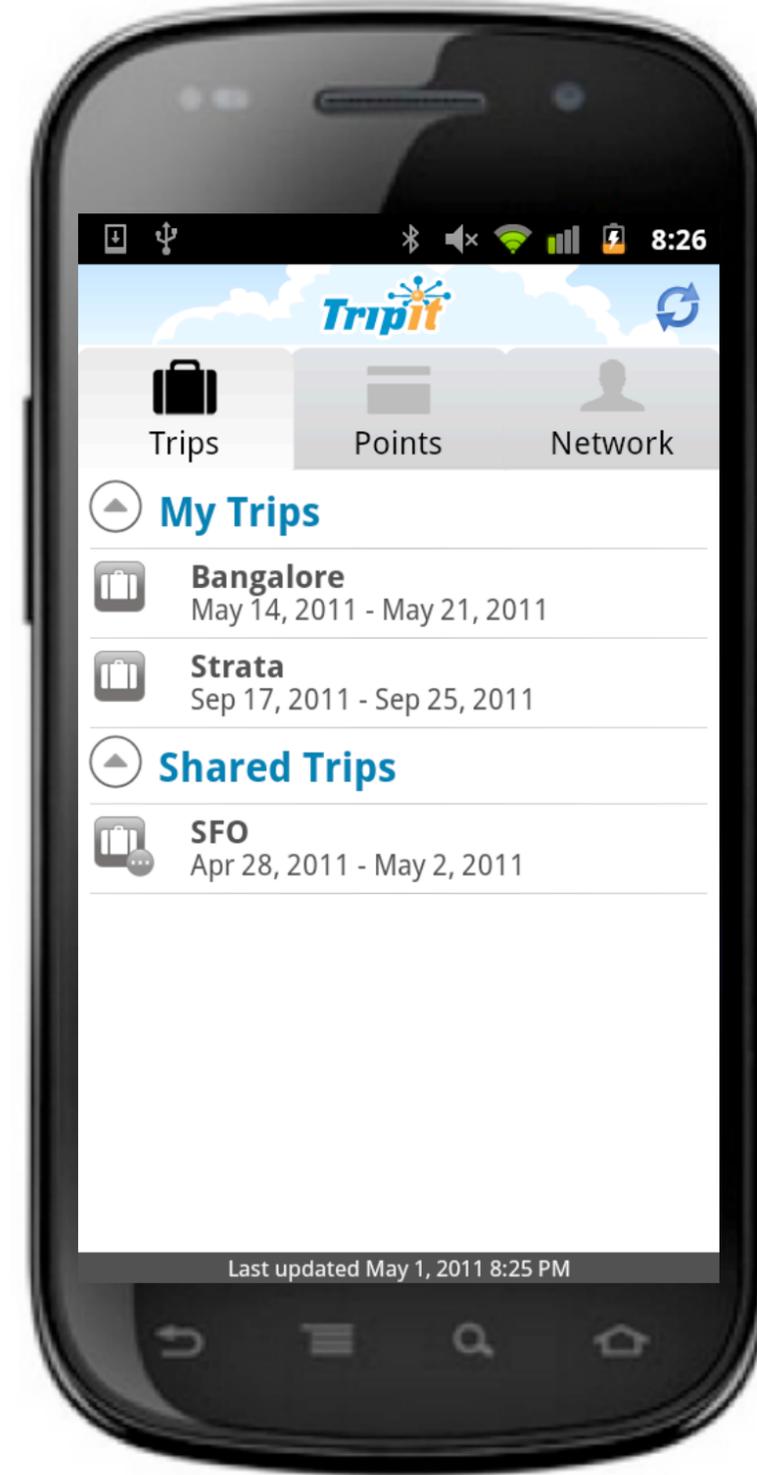
1



2



3



4

Getting authorized data access via OAuth

OAuth: Why?

35+ APIs

What data can your app access?



Contacts



Calendar



**Picasa Web
Albums**



YouTube

Why not just ask for the user's password?

Random Website

← → × 🏠 🔍

Tell us your Gmail account

Username

Password

PS we PROMISE we won't tell anyone your password!

OAuth: Terminology & Concepts

OAuth Terminology & Concepts



Protected Resource

- Resides on server
- Requires authorization



Resource Owner

- Owns protected resource
- Approves access

OAuth Terminology & Concepts



Server

- Holds the protected resource

SaaSy Payroll
your payroll solution, your way

Client

- Web application
- Needs access to the protected resource

Who Owns the Data?



Individual owns the resource

- Individual user owns their own data, and decides whether to grant access



Company owns the resource

- Data is owned by a company and access is granted by IT guardians

OAuth Individual Grant Use Case (via OAuth 2.0)

Who Owns the Data?



Individual owns the resource

- Individual user owns their own data, and decides whether to grant access

SaaSy Payroll



SaaSy Payroll

1



The screenshot shows a web browser window with the URL payroll.saasyapp.com/paycheck.php. The page header features the SaaSy Payroll logo and a 'Login' button. The main content area displays 'Your Paycheck for 5/1/2011' with a summary of hours worked (74) and hourly rate (\$12). A table lists the breakdown of the paycheck, including total wages, 401(k) deduction, federal income tax, and state income tax. Below the table, there is a section for 'Future Pay Dates' listing dates from 5/8/2011 to 5/29/2011. At the bottom, a blue button with the number '31' and the text 'Add pay dates to your Google Calendar' is visible.

SaaSy Payroll
your payroll solution, your way

Login

Your Paycheck for 5/1/2011

You've worked a lot this week: **74 hours**
Your Hourly Rate is: **\$12**

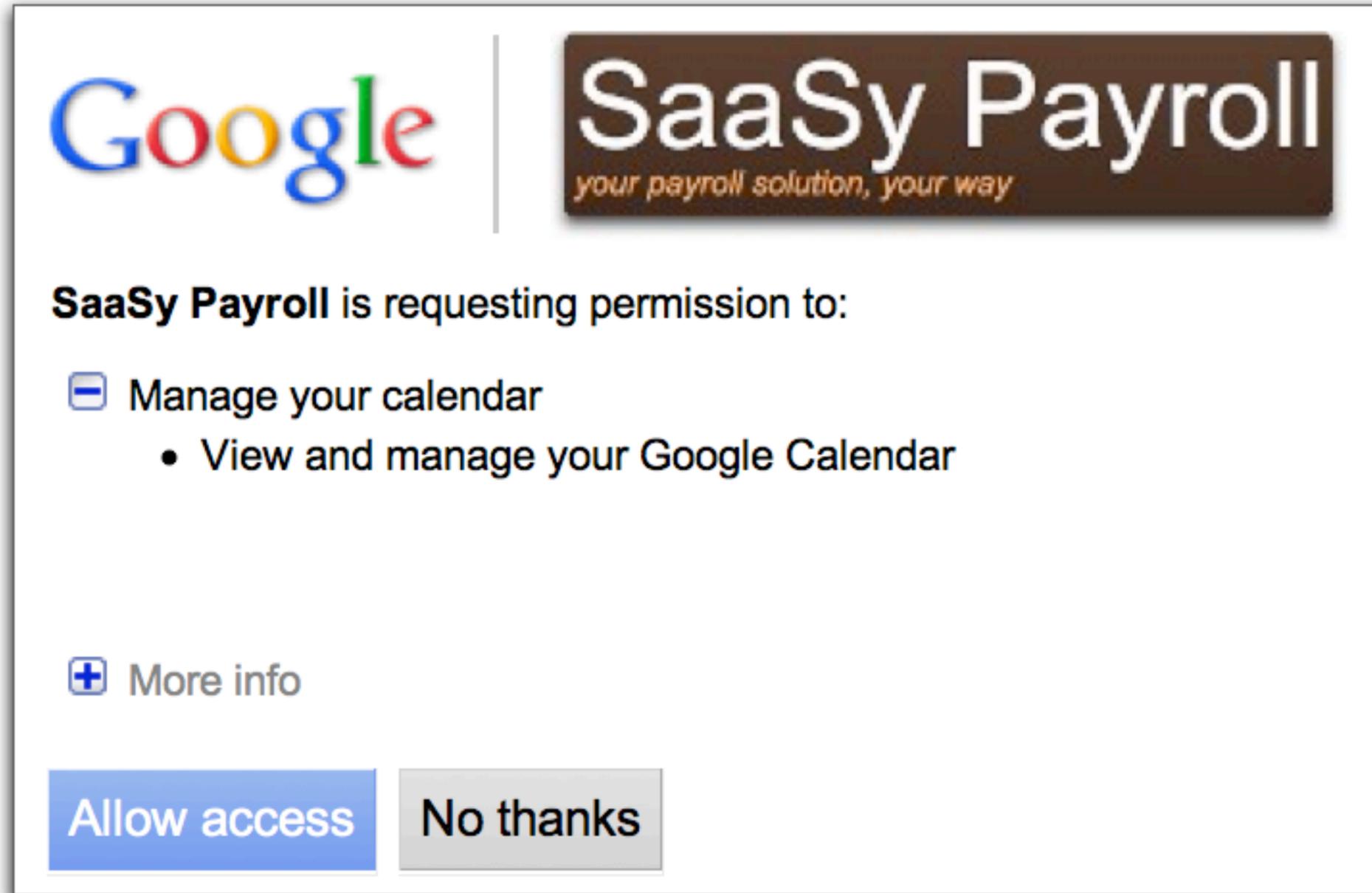
Item	\$
Total wages	888.00
401(k) deduction	(100.00)
Federal income tax	(244.00)
State income tax	(50.00)

Future Pay Dates:
5/8/2011
5/15/2011
5/22/2011
5/29/2011

31 [Add pay dates to your Google Calendar](#)

Access Control Grant

2



The screenshot shows a dialog box with the Google logo on the left and the SaaSy Payroll logo on the right. The SaaSy Payroll logo includes the text "SaaSy Payroll" and the tagline "your payroll solution, your way". Below the logos, the text reads "SaaSy Payroll is requesting permission to:". There are two main items listed: "Manage your calendar" with a minus sign icon, and "More info" with a plus sign icon. The "Manage your calendar" item has a sub-item: "View and manage your Google Calendar". At the bottom, there are two buttons: "Allow access" (blue) and "No thanks" (grey).

Google | **SaaSy Payroll**
your payroll solution, your way

SaaSy Payroll is requesting permission to:

- Manage your calendar
 - View and manage your Google Calendar
- More info

Allow access **No thanks**

Payroll on the Calendar

3

 Print			Day	Week	Month	4 Days	Agenda
Thu	Fri	Sat					
29	30	May 1					
6	7	8				Pay Day! (S)	
13	14	15				Pay Day! (S)	

Ryan's Calendar

OAuth 2.0 Flow

Registering an Application



0

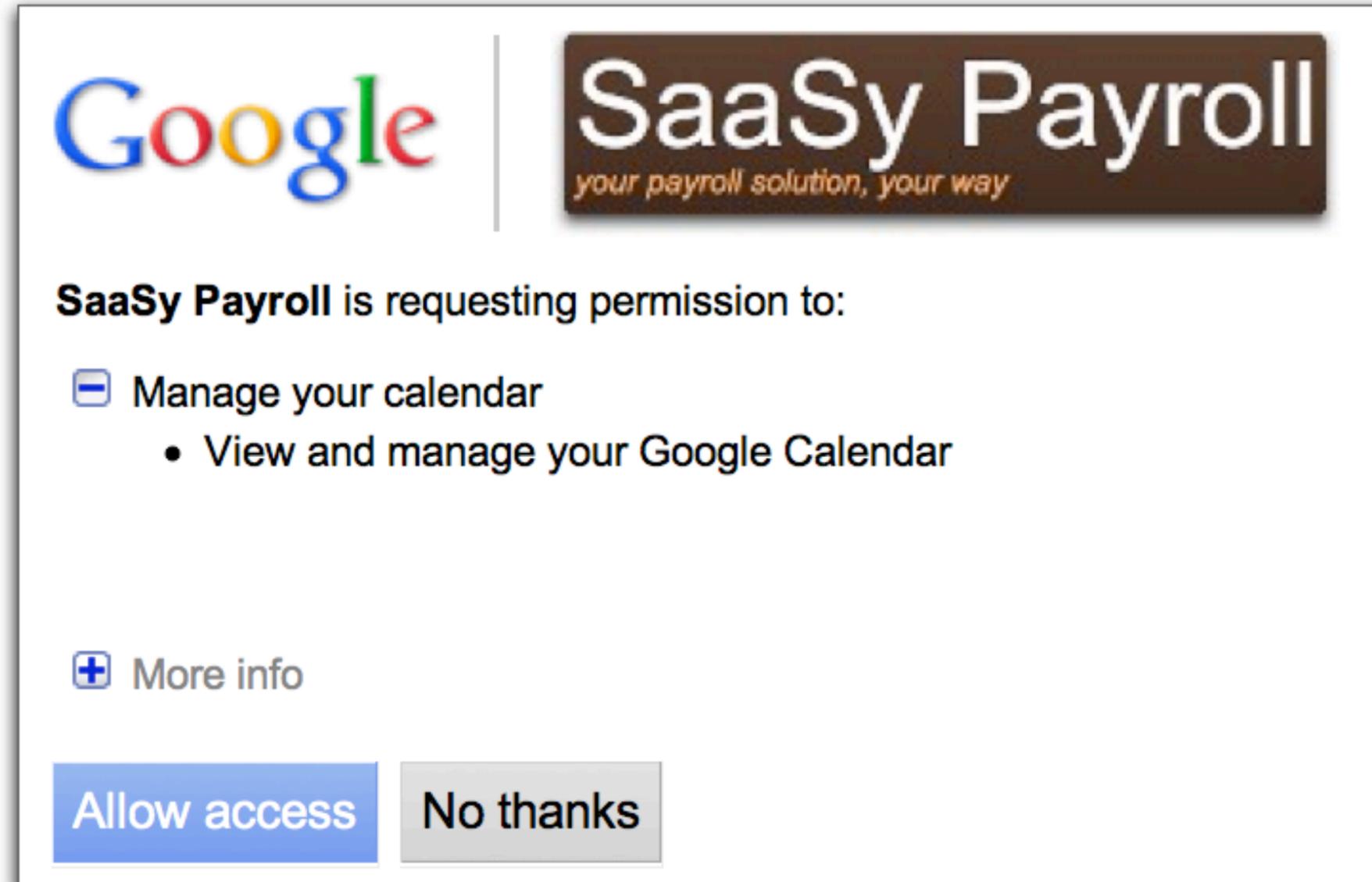
Developer registers application with Google, gets a `client_id` and `client_secret`

A screenshot of the Google APIs console interface. The browser address bar shows the URL: <https://code.google.com/apis/console/?pli=1#project:605902584318:overview:access>. The page header includes navigation links for Mail, Calendar, Documents, Sites, Video, and more, along with the user's email (ryan@ryguy.com) and links for My Account, Help, and Sign out. The main content area is titled "Google apis" and features a sidebar with a dropdown menu for "API Project" and a list of items: "Project Home", "APIs (2)", "Services", "API Access" (highlighted), "Billing", and "Team". The main content area has a heading "API Access" followed by a paragraph: "To prevent abuse, Google places limits on API requests. Using a valid OAuth 2.0 token or API key allows you to exceed anonymous limits by connecting requests back to your project." Below this is a section titled "Authorized API Access" with a text box containing: "OAuth allows users to share specific data with you (for example, contact lists) while keeping their usernames, passwords, and other information private. [Learn more](#)". To the right of this text is an OAuth logo. At the bottom right of the text box is a blue button that says "Create an OAuth 2.0 client ID...".

Granting Data Access

1

Application redirects user to Google, specifying:
`client_id` obtained during registration
`redirect_uri` for user to return to
`scope` or APIs the app needs access to



Obtaining an Access Token and Refresh Token

- 2 Google redirects the user back to the application's `redirect_uri` and includes an `authorization_code` in the URL.

```
http://www.saasyapp.com/payroll/back?code=<authorization_code>
```

- 3 Application performs a HTTP `POST` request to Google, including the `client_id`, `client_secret` and `code`. Google returns an `access_token` and a `refresh_token`.

```
{  
  "access_token": "1/fFAGRNJru1FTz70BzhT3Zg",  
  "expires_in": 3920,  
  "refresh_token": "1/6BMfW9j53gdGX-tqf8JXQ"  
}
```

Calling an OAuth Protected API

- 4 Application makes a HTTP **GET** or HTTP **POST** request to the server containing the protected resource, including the **access_token** as a query param or header.

Query-param:

```
https://www.google.com/calendar/feeds/default/private/full?oauth_token=<access_token>
```

Header:

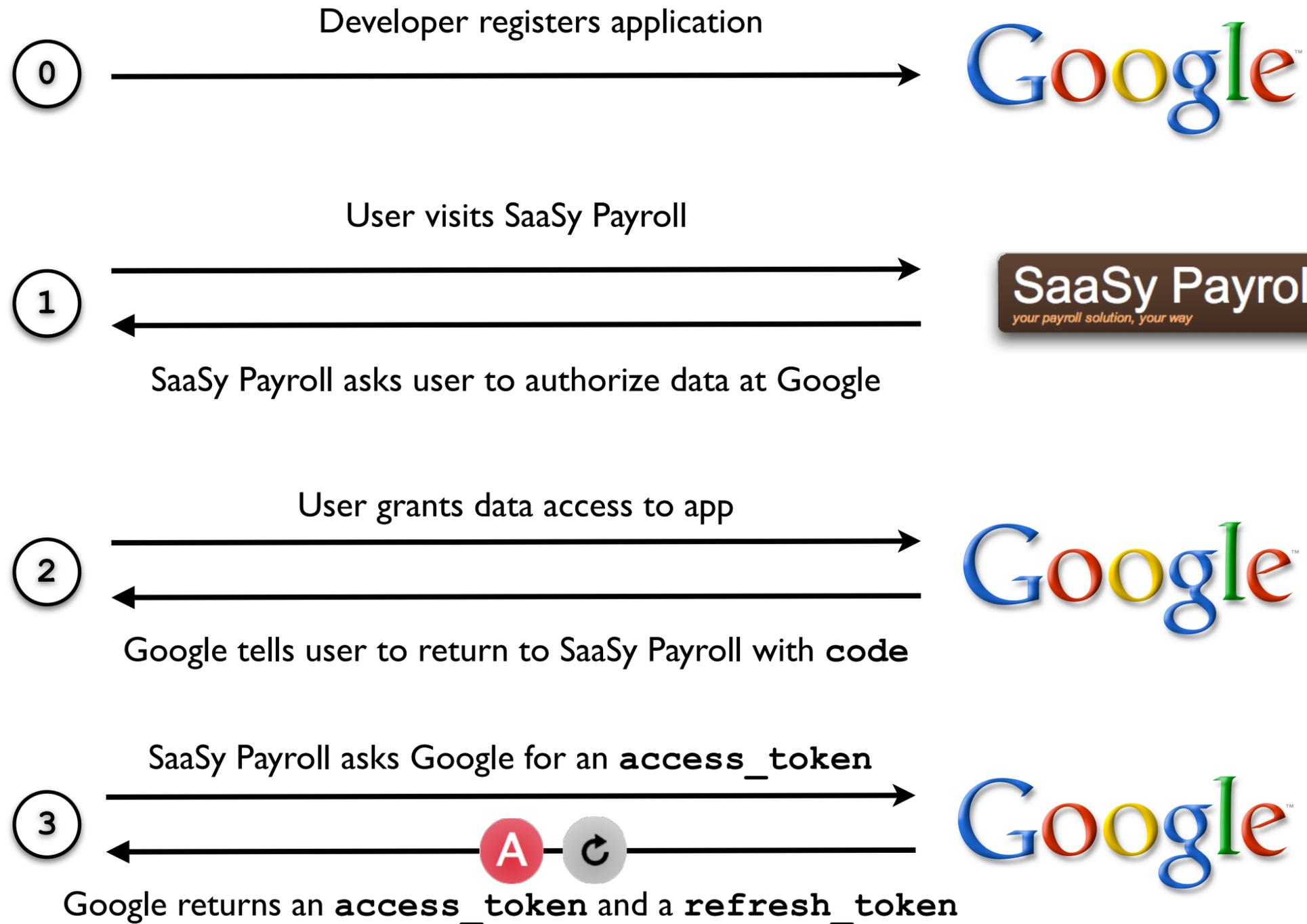
```
Authorization: OAuth <access_token>
```

Refreshing the Access Token

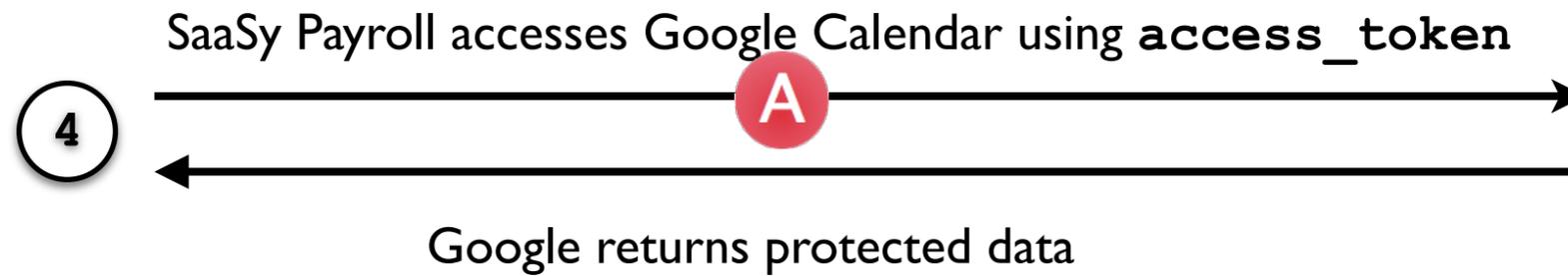
- 5 Application performs a HTTP **POST** request to Google, including the `client_id`, `client_secret` and `refresh_token`. Google returns an `access_token`. Refresh token remains the same, indefinitely until revoked.

```
{  
  "access_token": "1/fFAGRNJru1FTz70BzhT3Zg",  
  "expires_in": 3920  
}
```

OAuth 2.0: The Whole Flow



The Whole Flow (Continued)



Some time later



OAuth Business Use Case (via 2-legged OAuth 1)

Who Owns the Data?

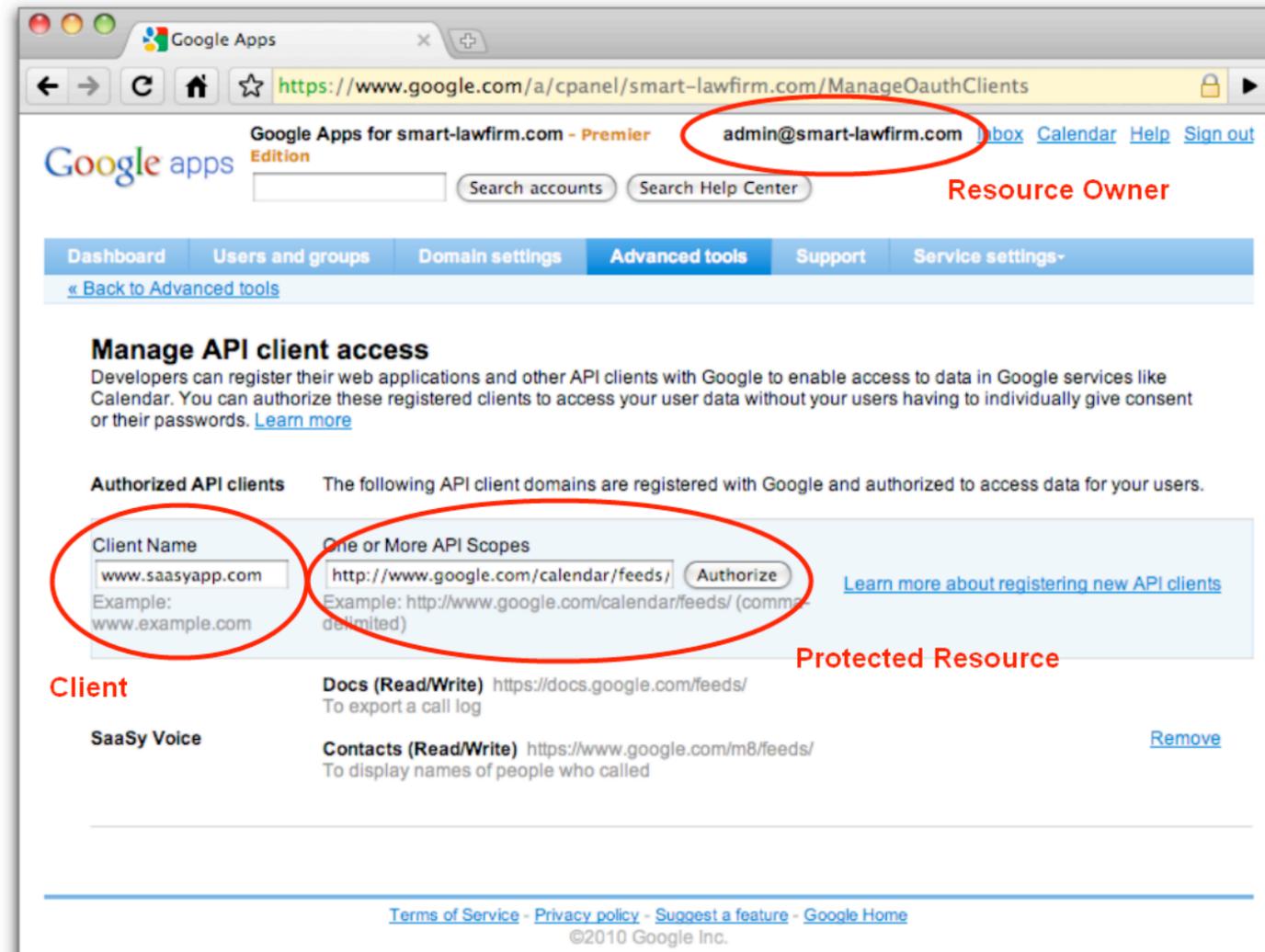


Company owns the resource

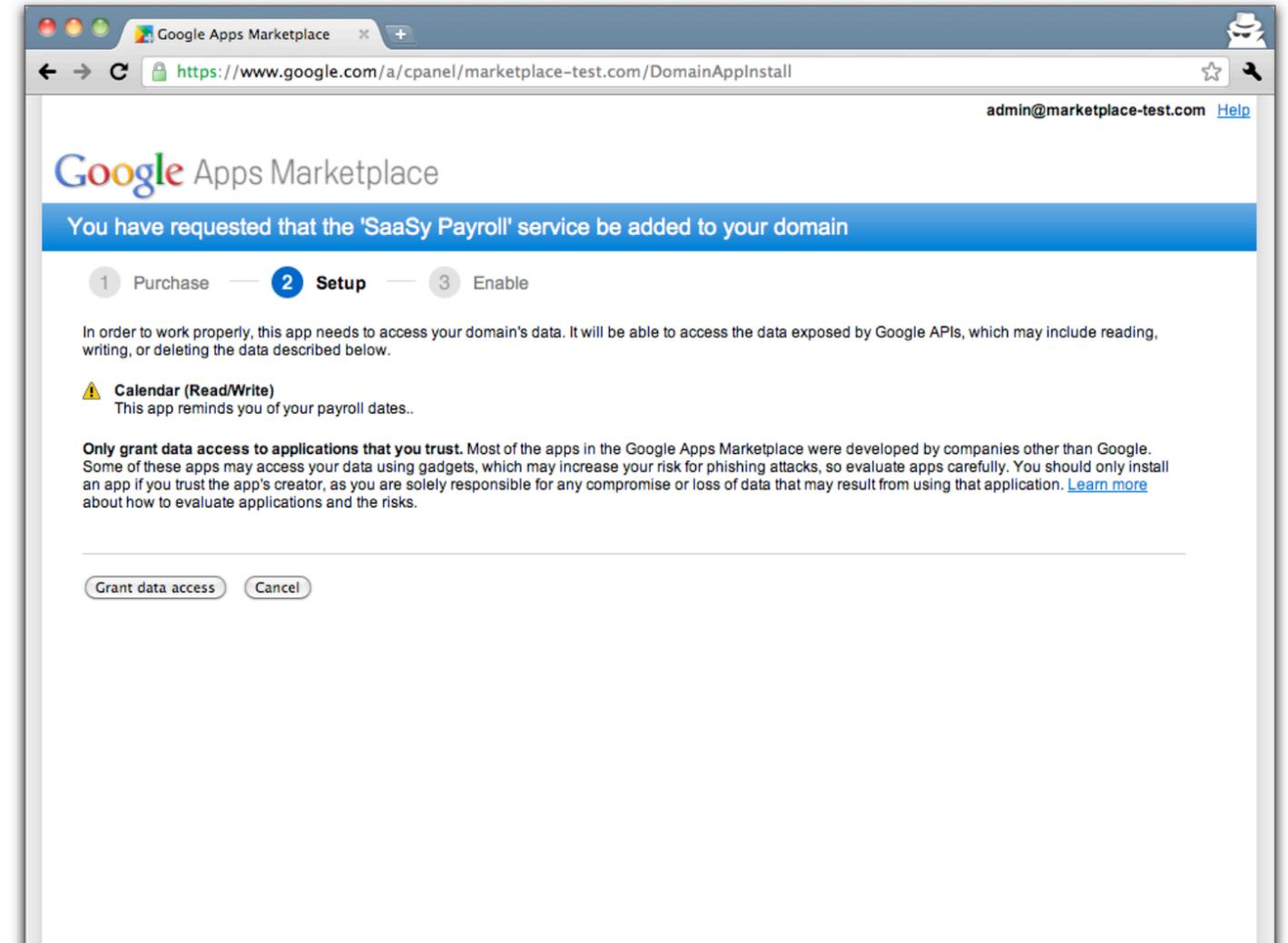
- Data is owned by a company and access is granted by IT guardians

Access Control Grant

1



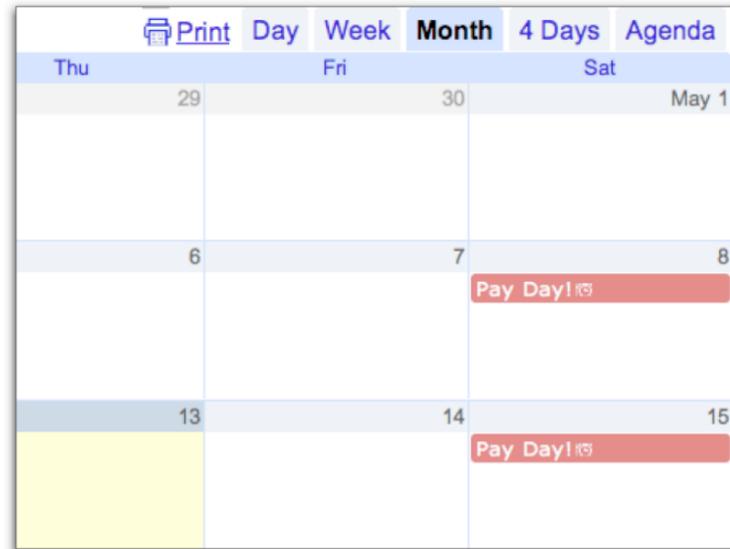
Google Apps Control Panel



Google Apps Marketplace

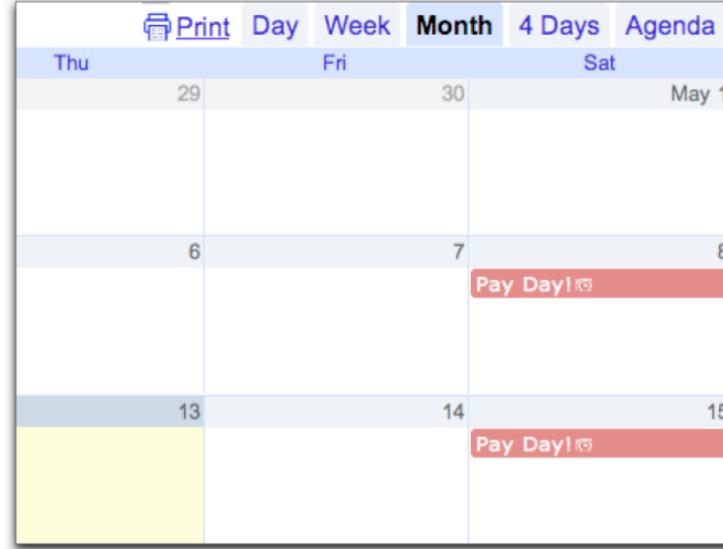
Payroll on All Employees' Calendars

2



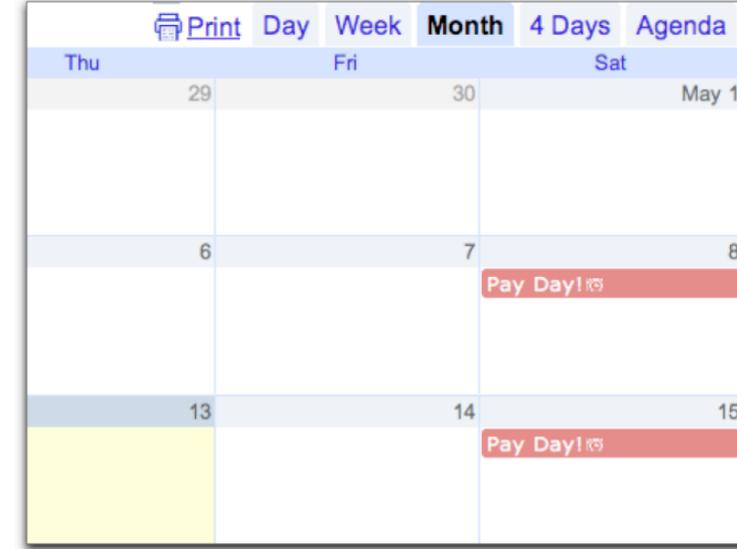
A monthly calendar for Ryan showing the month of May. The calendar is in a grid format with columns for days of the week (Thu, Fri, Sat) and rows for dates. The dates shown are 29, 30, May 1, 6, 7, 8, 13, 14, and 15. There are two red boxes labeled "Pay Day!" on the calendar, one on Friday, May 7, and one on Friday, May 14. The calendar has a navigation bar at the top with options: Print, Day, Week, Month, 4 Days, and Agenda. The "Month" option is selected.

Ryan's Calendar



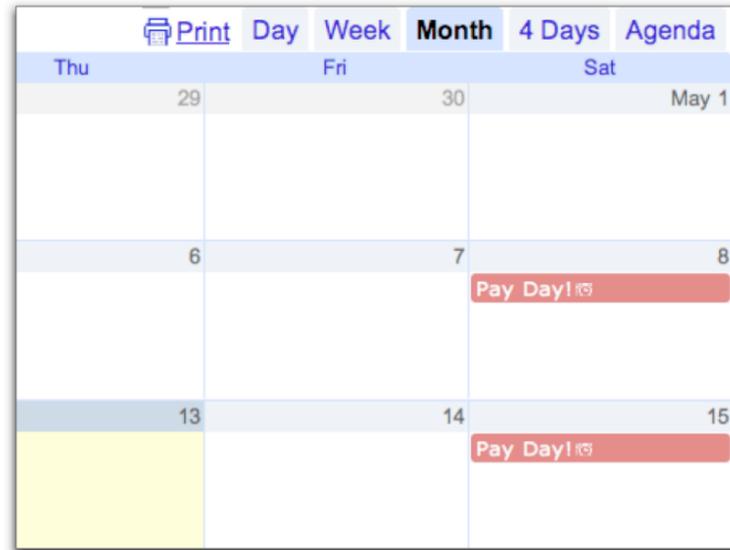
A monthly calendar for Tim showing the month of May. The calendar is in a grid format with columns for days of the week (Thu, Fri, Sat) and rows for dates. The dates shown are 29, 30, May 1, 6, 7, 8, 13, 14, and 15. There are two red boxes labeled "Pay Day!" on the calendar, one on Friday, May 7, and one on Friday, May 14. The calendar has a navigation bar at the top with options: Print, Day, Week, Month, 4 Days, and Agenda. The "Month" option is selected.

Tim's Calendar



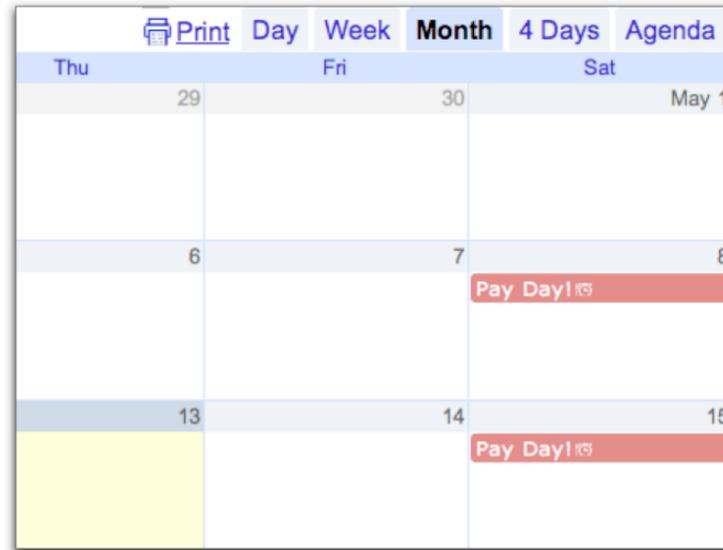
A monthly calendar for Steve showing the month of May. The calendar is in a grid format with columns for days of the week (Thu, Fri, Sat) and rows for dates. The dates shown are 29, 30, May 1, 6, 7, 8, 13, 14, and 15. There are two red boxes labeled "Pay Day!" on the calendar, one on Friday, May 7, and one on Friday, May 14. The calendar has a navigation bar at the top with options: Print, Day, Week, Month, 4 Days, and Agenda. The "Month" option is selected.

Steve's Calendar



A monthly calendar for Julia showing the month of May. The calendar is in a grid format with columns for days of the week (Thu, Fri, Sat) and rows for dates. The dates shown are 29, 30, May 1, 6, 7, 8, 13, 14, and 15. There are two red boxes labeled "Pay Day!" on the calendar, one on Friday, May 7, and one on Friday, May 14. The calendar has a navigation bar at the top with options: Print, Day, Week, Month, 4 Days, and Agenda. The "Month" option is selected.

Julia's Calendar



A monthly calendar for Scott showing the month of May. The calendar is in a grid format with columns for days of the week (Thu, Fri, Sat) and rows for dates. The dates shown are 29, 30, May 1, 6, 7, 8, 13, 14, and 15. There are two red boxes labeled "Pay Day!" on the calendar, one on Friday, May 7, and one on Friday, May 14. The calendar has a navigation bar at the top with options: Print, Day, Week, Month, 4 Days, and Agenda. The "Month" option is selected.

Scott's Calendar



A monthly calendar for Dan showing the month of May. The calendar is in a grid format with columns for days of the week (Thu, Fri, Sat) and rows for dates. The dates shown are 29, 30, May 1, 6, 7, 8, 13, 14, and 15. There are two red boxes labeled "Pay Day!" on the calendar, one on Friday, May 7, and one on Friday, May 14. The calendar has a navigation bar at the top with options: Print, Day, Week, Month, 4 Days, and Agenda. The "Month" option is selected.

Dan's Calendar

2-Legged OAuth 1 Flow

Registering an Application



0

Developer registers application with Google, gets a **consumer_key** and **consumer_secret**

Google accounts

Manage payroll.saasyapp.com (Active)

To register your domain, provide the following information. Once you've registered with an authentication certificate, you will be able to use secure tokens when communicating with a Google service.

Target URL path prefix:
Must be the prefix of the *next* parameter used in AuthSub.
e.g. <http://example.com/authsub>

Domain description: (Optional)
Tell us about your domain

OAuth Consumer Key: **payroll.saasyapp.com**
OAuth Consumer Secret: **0/HBCxRybXH+TD7011OypUdM**

We do not have a certificate for your domain.

Upload new X.509 cert: (Optional) No file chosen
File must be in PEM format. [Learn More](#)

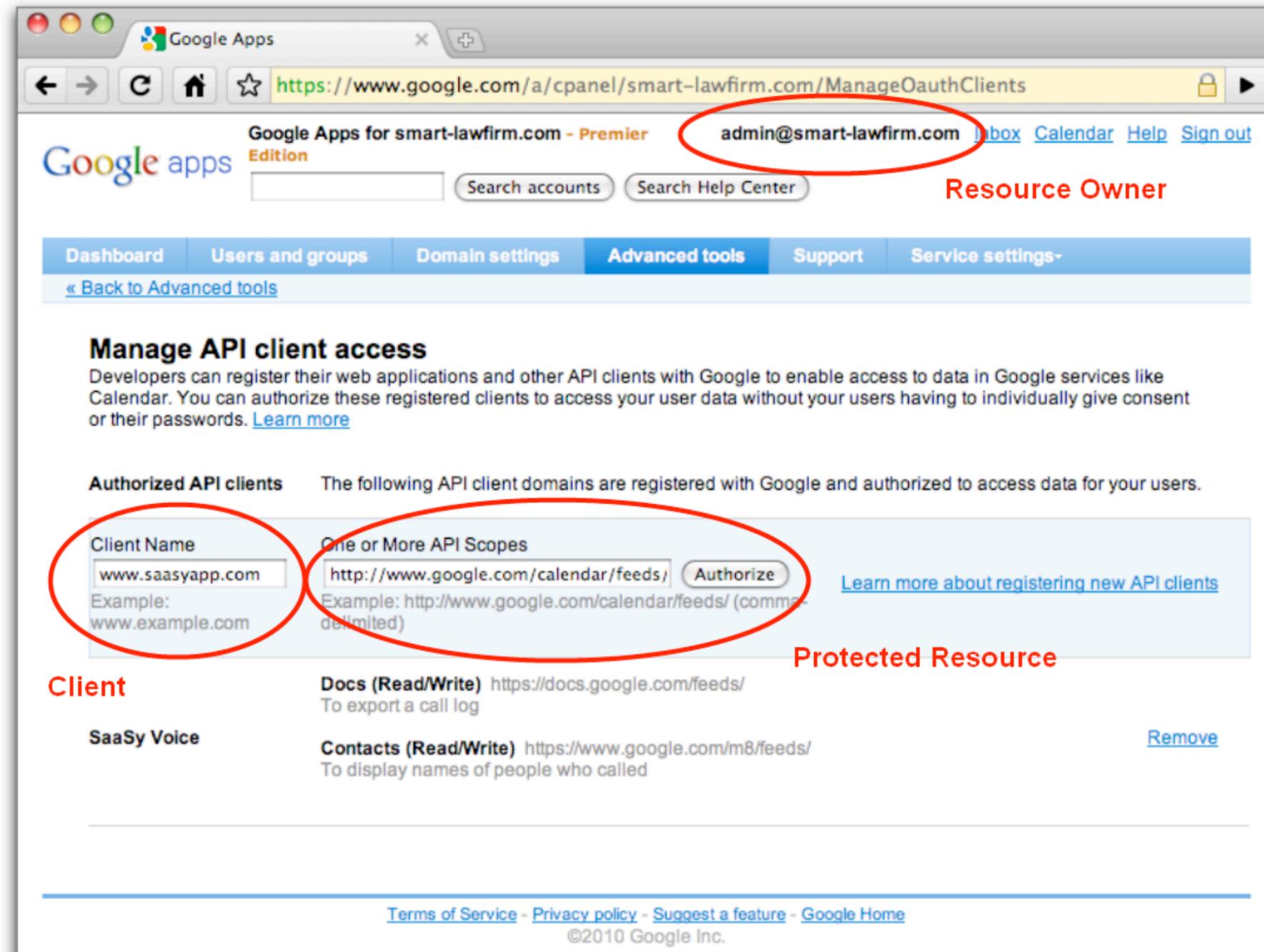
Test your AuthSub registration [here](#).

Granting Data Access

1 In an offline process, Google Apps domain administrator grants access to the app, specifying the **consumer_key** for the app **scope** or APIs the app needs access to

OR

The Google Apps domain administrator approves data access for the app during the installation from the Google Apps Marketplace.



Calling an OAuth Protected API

2

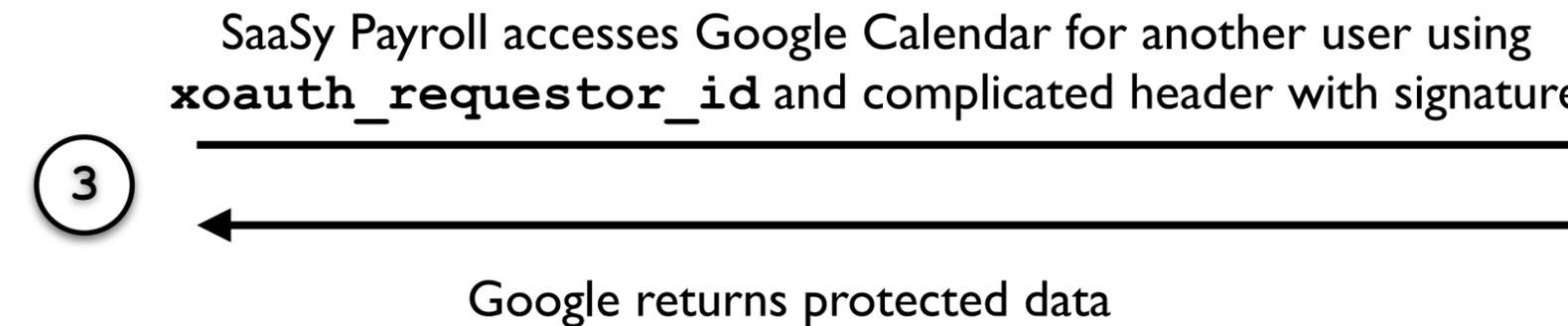
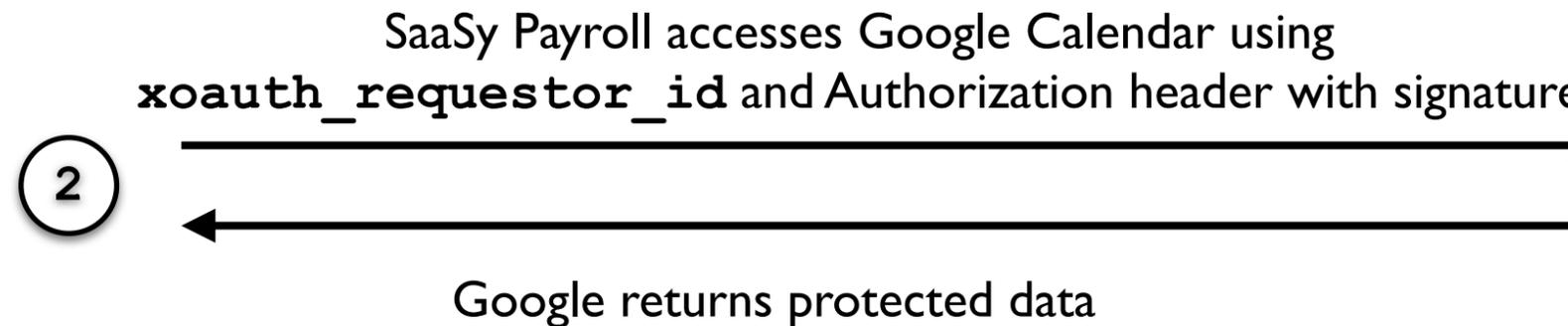
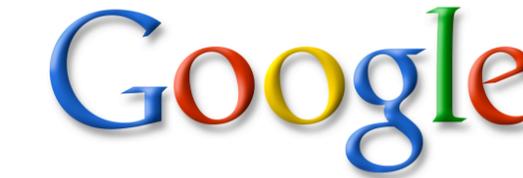
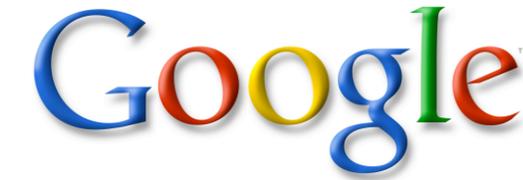
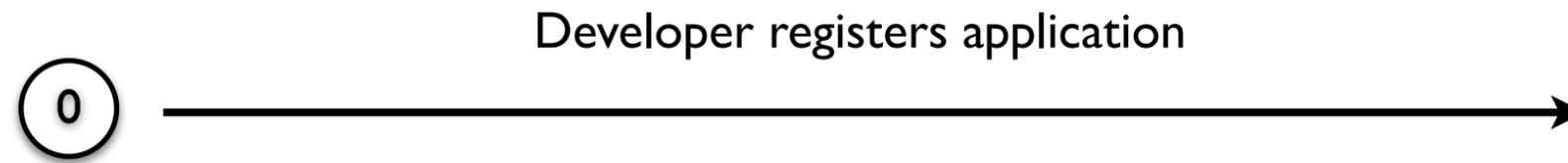
Application makes a HTTP **GET** or HTTP **POST** request to the server containing the protected resource, including an **Authorization** header. Additionally, the application specifies which user's data it is trying to access via a **xoauth_requestor_id** query parameter.

```
https://www.google.com/calendar/feeds/default/private  
/full?xoauth_requestor_id=<email address>
```

Header:

```
Authorization: OAuth  
  oauth_version="1.0",  
  oauth_nonce="1cbf231409dad9a2341856",  
  oauth_timestamp="123456789",  
  oauth_consumer_key="<consumer_key>",  
  oauth_signature_method="HMAC-SHA1",  
  oauth_signature="1qz%2F%2Bfwtsu0"
```

2-legged OAuth 1: The Whole Flow



Recap

Recap

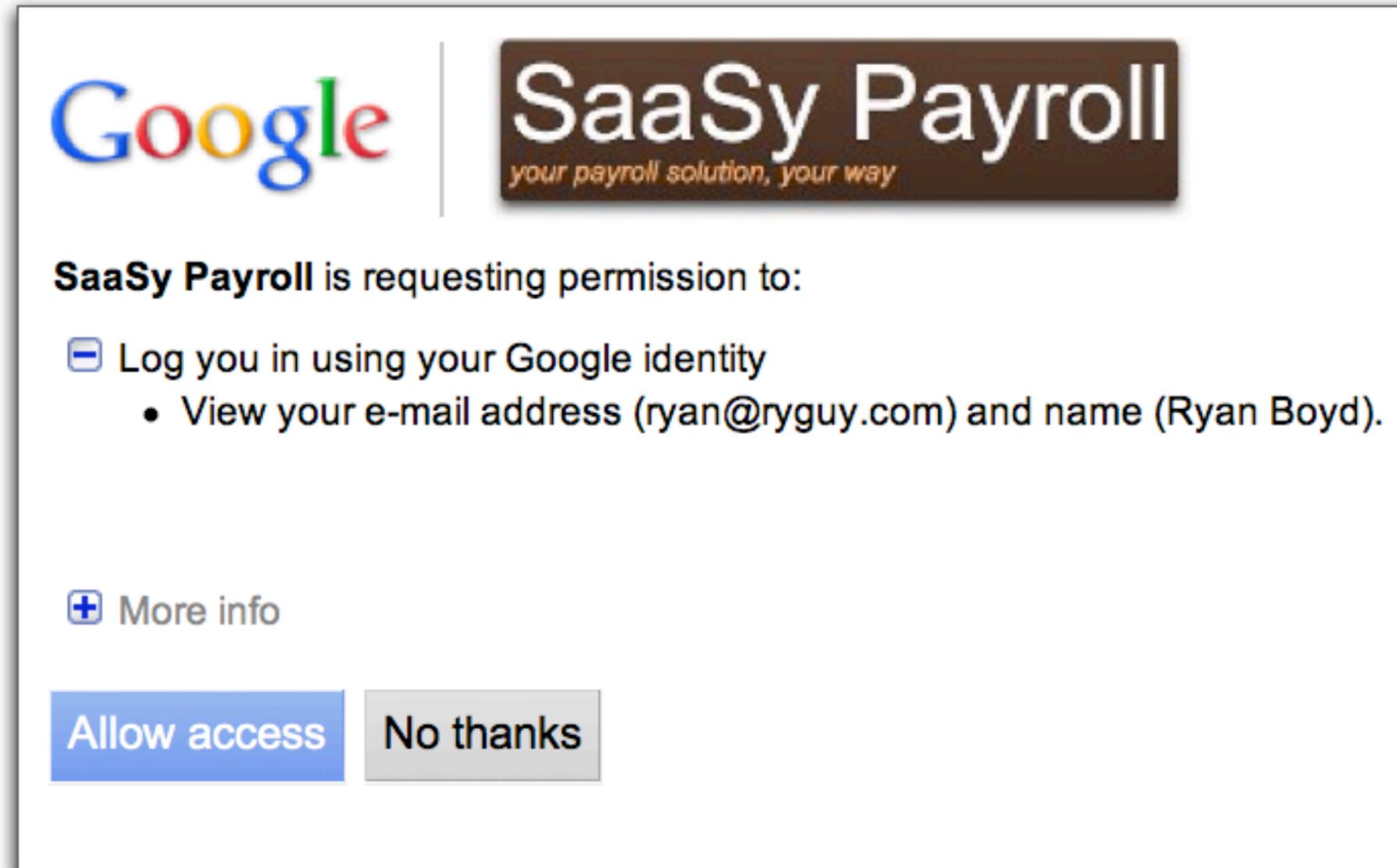
- 1 Terminology
- 2 OpenID for Authentication
- 3 Mobile Authentication
- 4 OAuth for Authorizing access to data owned by Individuals
- 5 OAuth for Authorizing access to data owned by Businesses

The Future

**One Protocol rules
all use cases**

Authentication
Authorization

One Protocol Rules all use cases



The screenshot shows a Google OAuth consent screen. At the top left is the Google logo. To its right is the SaaSy Payroll logo, which consists of the text "SaaSy Payroll" in white on a dark brown background, with the tagline "your payroll solution, your way" in orange below it. Below the logos, the text reads "SaaSy Payroll is requesting permission to:". Underneath, there is a list of permissions, each preceded by a minus sign icon. The first permission is "Log you in using your Google identity", which has a sub-bullet point: "View your e-mail address (ryan@ryguy.com) and name (Ryan Boyd)". Below the permissions list is a plus sign icon followed by the text "More info". At the bottom of the screen are two buttons: "Allow access" in a blue box and "No thanks" in a grey box.

Google | **SaaSy Payroll**
your payroll solution, your way

SaaSy Payroll is requesting permission to:

- [-] Log you in using your Google identity
 - View your e-mail address (ryan@ryguy.com) and name (Ryan Boyd).

[+] More info

Allow access **No thanks**

Resources

Resources

- 1 Google Identity Toolkit (<http://goo.gl/Tkklz>). Talk to esachs@google.com for tester access.
- 2 OAuth Playground for OAuth 1 (http://googlecodesamples.com/oauth_playground/)
- 3 Google's Auth docs (<http://code.google.com/apis/accounts/docs/>)
- 4 ClientLogin #FAIL I/O session from yesterday (<http://goo.gl/b78jJ>)
- 5 Ryan's Twitter ([@ryguyrg](https://twitter.com/ryguyrg))

Feedback: <http://goo.gl/DpUBh>
#io2011 #TechTalk

Feedback: <http://goo.gl/DpUBh>
#io2011 #TechTalk

Q & A

Google™

