OAuth

Hva var problemet?

- Mashups bruker data fra andre kilder
- Dele ut passordet sitt på Twitter/Flickr/ whatever? Don't think so

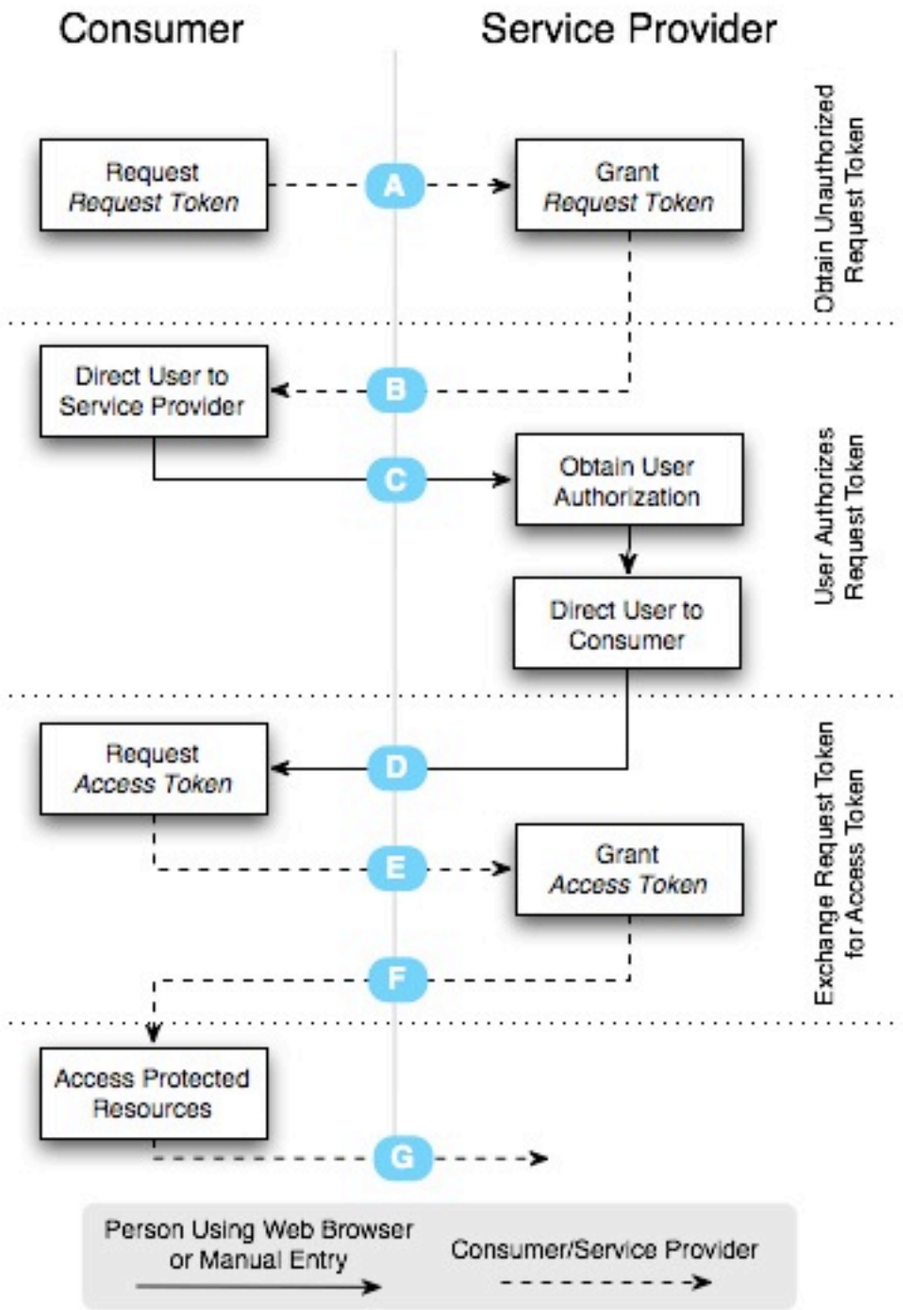
På veldig høyt nivå

- Få en nøkkel og hemmelighet fra siten du vil hente data fra
- Send nøkkelen over for å identifisere deg
- Signer dataene du sender med hemmeligheten dere begge vet om

Prosessen

- Request token
- Authorization URL
- Access token
- Rock&roll

OAuth Authentication Flow



A Consumer Requests *Request Token*

Request includes
oauth_consumer_key,
oauth_signature_method,
oauth_signature,
oauth_timestamp,
oauth_nonce,
oauth_version (optional).

B Service Provider Grants *Request Token*

Response includes
oauth_token,
oauth_token_secret.

C Consumer Directs User to Service Provider

Request includes
oauth_token (optional),
oauth_callback (optional).

D Service Provider Directs User to Consumer

Request includes
oauth_token (optional).

E Consumer Requests *Access Token*

Request includes
oauth_consumer_key,
oauth_token,
oauth_signature_method,
oauth_signature,
oauth_timestamp,
oauth_nonce,
oauth_version (optional).

F Service Provider Grants *Access Token*

Response includes
oauth_token,
oauth_token_secret.

G Consumer Accesses Protected Resources

Request includes
oauth_consumer_key,
oauth_token,
oauth_signature_method,
oauth_signature,
oauth_timestamp,
oauth_nonce,
oauth_version (optional).

Request token

- oauth_key
- oauth_signature (+metode)
- oauth_nonce
- oauth_timestamp
- alt annet

Persistering

- Et access token kan brukes om igjen, lagre key og secret i databasen, det er alt som trengs
- (…) så lenge det ikke er invalidert hos provideren

Signaturen

- alle parametre, sortert etter nøkkel
- hvis flere verdier i en nøkkel: sorter disse også
- skal kunne reproduseres i andre enden, må være deterministisk

Authorization URL

- Request token kan hente en URL for autorisering (ved å sende en signert HTTP request)
- redirect_to request_token.authorization_url

Access token

- Så snart brukeren har vært på besøk hos provideren kan du bytte inn et request token i et access token

Med et access token kan du

- hente alle ressurser, som om du var selve brukeren

Signering

- Støtter flere forskjellige signeringsalgoritmer:
  - HMAC:SHA1
  - RSA-SHA1
  - PLAINTEXT
  - (pluss eventuelt andre)

Hvordan?

```
consumer = OAuth::Consumer.new(KEY, SECRET, :site => SITE, :scheme
=> :GET)
request_token = consumer.get_request_token
redirect request_token.authorize_url
# ...
access_token = request_token.get_access_token
access_token.get('/friends.json')
```

Alternativer

- Authorization-header
- POST-body
- Query string

Hvor kan du bruke dette?

- Twitter
- Flickr
- Yahoo
- Google

I forhold til Open ID?

- Noe helt annet
- OAuth handler om å gi tilgang (autorisere)
- OpenID handler om å verifisere at noen er den de påstår å være

Begrensninger i Rubys OAuth

- Har ikke støtte for vilkårlige parametre bortsett fra ved henting av authorization_url
- Veldig mye kode
- Fungerer ikke asynkront (EM)
- Mikser inn i ActionController+Net::HTTP
- Avhengig av Railsversjon

Oauth-simple

- Basert på python-oauth
- Tilgang på protokollnivå
- API-kompatibelt med ruby-oauth

Demo time

Hva med Rails?

- OAuth plugin for Rails: http://code.google.com/p/oauth-plugin/
- Ut av boksen forutsettes acts_as_authenticated
- before_filter :login_or_oauth_required

Hva med Ruby 1.9?

- Funker

Spørsmål?