Working with Payment APIs

Tuomas Toivonen

March 30, 2010

Scred

- Scred.com: A service for informal and semi-formal groups to manage their money
  - Bands, Indie movie makers, event organizers, hobbyist groups, associations, ...
  - Meaningful Money = payments with context = automated accounting
- Flabat.fi: DIY event ticketing for the Finnish market
- Luottokunta (credit and debit cards), Finnish bank payment buttons, PayPal Adaptive Payments APIs

Web payments in Finland

- Bank payment buttons
- Credit cards: Luottokunta or DIBS
- Aggregators: Checkout, Suomen Verkkomaksut
- Value added payments: Klarna, Collector, Suomen Maksuturva
- E-wallets: PayPal and Moneybookers

Conceptual model for payments

- Actors:
  - Sender (person making the payment, payer)
  - Receiver (person or company receiving the payment, payee)
  - Caller (company initiating the payment instruction)
  - Processor (company receiving and processing the payment instruction)

Payment processing steps

1. Caller prepares and delivers payment instruction to Processor
2. Sender informs the Processor of payment instruction approval
3. Processor charges Sender and credits Receiver
4. Processor informs Caller of succesful payment processing

Security requirements

- Messaging integrity between Caller and Processor
- Mutual authentication between Caller and Processor
- Processor needs to authenticate Sender
- Sender needs to deliver authorization to Processor

Payment instruction delivery and authorization

HTTP POST with shared secret and message authentication code (MAC)

- POST to processor (SECRET=haukionkala):

POST https://www.bank.fi/payments.cgi HTTP/1.1
AMOUNT=42.00
MERCHANT=saippuakauppias
RETURN=http://www.shop.fi/return.cgi
MAC=677135ffd661d3e62e8ccc01edaeb821

- Return GET from processor:

GET https://www.shop.fi/return.cgi?AMOUNT=42.00&
MAC=d85090a0c7260948568db1a1ab79c65f

Payment instruction delivery and authorization

Background API with single use sessions and redirect

- Request one time token for a payment instruction:

```
POST https://www.pay.com/session.cgi HTTP/1.1
AMOUNT=42.00
USERNAME=saippuakauppias
PASSWORD=haukionkala
SENDER=shopper@customer.com
RECEIVER=merchant@webshop.com
RETURN=http://www.shoppe.com/return.cgi
```

- Redirect shopper:

```
302 Found
Location: https://www.pay.com/pay.cgi?token=xyz123
```

Case Nordea (1)

Return destinations and MACs

- Three types of return destinations:
  - SUCCESS
  - REJECT
  - CANCEL
- Nordea: Return MAC with all three
- Other banks: Return MAC with only SUCCESS

Case Nordea (2)

Nordea return parameters

- Nordea return GET parameters:
  - SOLOPMT_RETURN_VERSION = 0003
  - SOLOPMT_RETURN_STAMP = unique identifier
  - SOLOPMT_RETURN_REF = reference number
  - SOLOPMT_RETURN_PAID = bank archival reference
  - SOLOPMT_RETURN_MAC = authentication & integrity
- Problem: Same parameters with SUCCESS, REJECT and CANCEL
- Except: SOLOPMT_RETURN_PAID only with SUCCESS

Case Nordea (3)

Nordea flow and abuse

- Example site:
  - SUCCESS = http://www.saitti.fi/pay-ok.cgi
  - REJECT = http://www.saitti.fi/pay-fail.cgi
  - CANCEL = http://www.saitti.fi/pay-cancel.cgi
- Flow: Create order — Checkout — Cancel at bank
- Abuse: Post to CANCEL URL GET parameters to SUCCESS URL

Security tips

- Authenticate the message sender!
- Make sure the message makes sense!
- Verify message against payment model in the database
- Do not use sequential payment identifiers
- Use separate payment verify call if available
- Build in daily reconciliation procedures

PayPal Adaptive Payments

Released in late 2009 — still rapidly evolving.

- Adaptive Payments API
  - Actors: Caller, Sender, Receiver(s)
  - Payment model: one to one or one to many
  - Fees: sender or receiver pays
  - Pre-approvals support for direct debit
- Adaptive Accounts API
  - Provision PayPal accounts for your users
- Permissions API
  - Request access to a PayPal account with OAuth-like mechanism
  - Account statements, outbound payments, refunds, ...

Warning! Documentation weak. Approvals process confusing.

PayPal vs Amazon

	PayPal	Amazon
APIs:	Adaptive Payments	Flexible Payments Service
API caller:	Global	US only
Merchants:	Global	US only
Purchasers:	Global	Global
More:	www.x.com	aws.amazon.com/fps/

Thanks!

Questions?