Credit Card Payments

## Scenario

- You deploy a wonderful eCommerce site
- Users start to browse your store
- One user actually decides to buy something!
  - They add items to their cart
  - Navigate your checkout process
  - Enter their credit card information
  - Submit their order
- Now what?
- How do you get their money?

- Parts of a credit card number
  - Numbers do not have to be 16 digits
  - First 5 digits: Issuer Identification Number (IIN) • Remaining digits except the last: Account number
- - Last digit: Checksum



- Use the Luhn algorithm algorithm to verify the card
  - Double every other digit starting with the first
  - Sum up all the digits
  - - 4007 7028 3553 2454
    - 8007 14048 65103 44104
- 8+0+0+7+1+4+0+4+8+6+5+1+0+3+4+4+1+0+4

60

• If the sum is not divisible by 10, it's not a valid card

- IIN identifies the issuer (Bank) and type of card
- Use the first few digits to determine the type of card
  - Visa: Starts with 4
  - MasterCard: Starts with 51-55 or 2221-2720
  - AmEx: Starts with 34 or 37



 Once you determine the type of card Connect to the appropriate API

• This is a Visa card so we'll send the transaction to Visa using their API



- **1.** Purchase 3 **User's Bank** 2 VISA
- User decides to make a purchase • They enter their credit card information and submit Your server receives the requests and processes the order

5





4

**Visa API** 

- Your server sees that the user has a Visa credit card
- Connect to the Visa API and request that the appropriate funds are sent to your account





Your Bank

- Visa checks the IIN of the card to determine the issuing bank
- Visa checks with the bank to verify the transaction
- The bank can decline the transaction at this point
  - le. Fraud detection, insufficient funds, etc



**3: Authorization** 

2

5

**User's Bank** 

3

4

**ISA** 

**Visa API** 



### 4: Settlement • If the transaction is approved, the funds are added to your account



**Your Server** 



### **5: API Response**

2

5

- - a receipt



• Visa reports back to your server that the transaction is complete • Your server will update all records and finish processing the order • Eg. Update accounting records, prepare for shipping, generate

**Visa API** 

**User's Bank** 

3

4



### **6: Finalize Transaction** Report back to your user with confirmation that their

2

5

- purchase is complete
  - Eg. Display their receipt, send a confirmation email, followup with tracking numbers



**User's Bank** 

3

4

**ISA** 

**Visa API** 



## Problems

• There are **many** complications with the process we just went through

- The example used Visa, b companies
- For each company:
  - Create a dev account
  - Study their API documentation
  - Write code integrating their API
- This is a lot of code maintain

The example used Visa, but there are many credit card

ntation heir API ain

## Problems

- Your bank, etc.
  - These APIs are very powerful!

    - trusted businesses
- is tedious

Must get approval from Visa, MaterCard, Discover, AmEx,

 You have the power to move any amount of money Access to these APIs must be secure and restricted to

Getting/maintaining approval from every payment method

## Problems

If you want to save users' credit cards for later purchases
You are in charge of storing this sensitive information
You cannot secure that database!
Discussion: Why not?

• It's often a better solution to use a payment processor • Eg. PayPal, Square, Stripe

- The processor works directly with your customer to approve the order
  - Eliminates the need to trust you
- If the customer agrees to the purchase, the processor connects to the appropriate credit card API
  - Eliminates the need for you study multiple APIs
- Processor can maintain customer cards
  - No need for you to store credit cards
- Handles fraud and refunds for you
  - More safety for the customer

### Payment Processors

You connect to one API only and tell the processor to handle the transaction

### Payment Processors - Fees

- Why wouldn't you use a payment processor?

  - directly to credit card APIs
- PayPal: 3.49% + \$0.49
- Square: 2.9% + \$0.30
- Stripe: 2.9% + \$0.30

• Credit Card: 1.3% - 2.6% + \$0.05 - \$0.10

 They'll want a cut for the work they do for you The fees will often double as opposed to connecting

• Small to mid-sized companies Almost always better to pay the fees and use a processor

• Large companies

- to connect to the credit card APIs
- Very expensive to develop/maintain this software

### Payment Processors

• Sometimes more economical to write your own software • Must have very large volume to offset this cost