SESSION ID: HT-W11

API Security Exposure for Gift Card Fraud: A 15-year old’s guide

Tanay Deshmukh
High School Student
Amador Valley High School
Pleasanton, CA
About Me

- High school sophomore
- Student at Amador Valley High School in Pleasanton
- Self taught and started coding at age 12
- Found major vulnerabilities for Chipotle, Spotify, NCR, and JambaJuice
- Built Chrome extensions for buying high demand items
- Participant in HackerOne
- Platinum tier for US Cyberpatriot
- Github: t4nay
What will you learn and how can you use the learnings?

What will I talk about?
- Securing API for services & gift cards

What will you learn?
- How hackers can exploit vulnerabilities

How can you apply the learnings?
- Use the best practices to secure APIs and learn new tools & techniques
Goals for my talk

– Understand how hackers exploit vulnerabilities using
  • Credential stuffing
  • SQL Injection
  • Web scraping

– Use techniques to protect by implementing
  • Captcha
  • Rate Limiting
  • Limiting public use, VPN access and increasing verifications
What is a Gift Card or Cash Card?

A gift card (also known as gift certificate in North America, or gift voucher or gift token in the UK) is a prepaid stored-value money card, usually issued by a retailer or bank, to be used as an alternative to cash for purchases within a particular store or related businesses.

Source - Wikipedia
What is a Subscription Account?

The subscription business model is a **business model** in which a customer must pay a **recurring price at regular intervals** for access to a product or service.

Source - Wikipedia
Why are Gift card and Subscription accounts vulnerable?

**Gift Cards:**

Gift cards are vulnerable due to how they are generated and how balance checks are handled.

**Subscription Accounts:**

Vulnerable from little to no protection from credential stuffing attacks on websites and old websites with recycled passwords.
How did I find the vulnerabilities?

While learning about web development and how to secure web applications to the best of my ability -- started finding some security holes

- Mobile API was not as secure as the browser
  - Mobile API’s typically do not have as much rate limiting as web applications do
- Credential stuffing is an attack which can easily be performed on most websites

My intentions were to learn and I found a way to help companies fight fraud
Common methods used by hackers

- Credential Stuffing
- Web Scraping
- SQL Injection
What is Credential Stuffing?

Credential stuffing is the **automated injection of breached username/password pairs** in order to fraudulently gain access to user accounts.
How does Credential Stuffing work?

1. Obtain credentials
2. Obtain proxies (optional)
3. Create/use existing config for website
Ways to prevent Credential Stuffing

• Rate limiting
  – Use a commercial or open source software
  – Build it in application logic
• Captchas
• 2-step verification
SQL Injection

- SQL Injection is the main method attackers use to gain access to credentials
  - Compromised databases are used for accessing credentials
Web Scraping

- Credentials can also be found on publicly on the internet easily
  - Links are scraped from keywords and are the program parses credentials
What is a Config?

- Configs are files which instruct the program the series of requests and responses to check for a valid login.
Tools used

- For credential stuffing a wide variety of tools can be used
  - **SentryMBA**
    - Original program for credential stuffing and the most popular
  - **SNIPR**
    - Costs money
    - Has a public and private repo for configs built in
    - Built in proxy scraper and leecher
  - **OpenBullet**
    - Can use selenium and has a simple system of making configs
  - **SQLi Dumper**
    - Has many different versions and is the primarily used program for SQL injection
Demo
How to fix the problems?

• Captcha
  – Captchas make it harder for bots to send automated requests
    o It adds additional steps to perform an attack
    o Forces the attacker to pay for a captcha solving service (ex: 2captcha)

• Rate Limiting
  – Prevent bots from sending requests at a faster rate
  – Prevent websites slowing down from bots sending a large amount of requests in a short period of time
What did I do after finding the vulnerabilities?

- Reached out to the customer service department and in some cases the InfoSec team on what I had found
- In some cases, used HackerOne to submit the vulnerabilities
- Sent a screenshot of the problem identified
- Made myself available for a call with the team
- Shared my findings and fixes that were needed
What’s next for me?

- Continue learning more about vulnerabilities, security tools and techniques
- Help companies with preventing fraud if I find something new
- Learn and share what I find
- Continue my high school for next three years -- enter a Computer Science program
- Learn more programming languages
- *Looking for short or long term research opportunities - I would love to talk*
Questions?

Thank you!

Contact: tanayemail@gmail.com