Hunting the 6 most common price manipulation vulnerabilities in e-commerce websites

BY BLACKBIRD-EU · FEBRUARY 5, 2024 · LAST UPDATED ON FEBRUARY 3, 2026

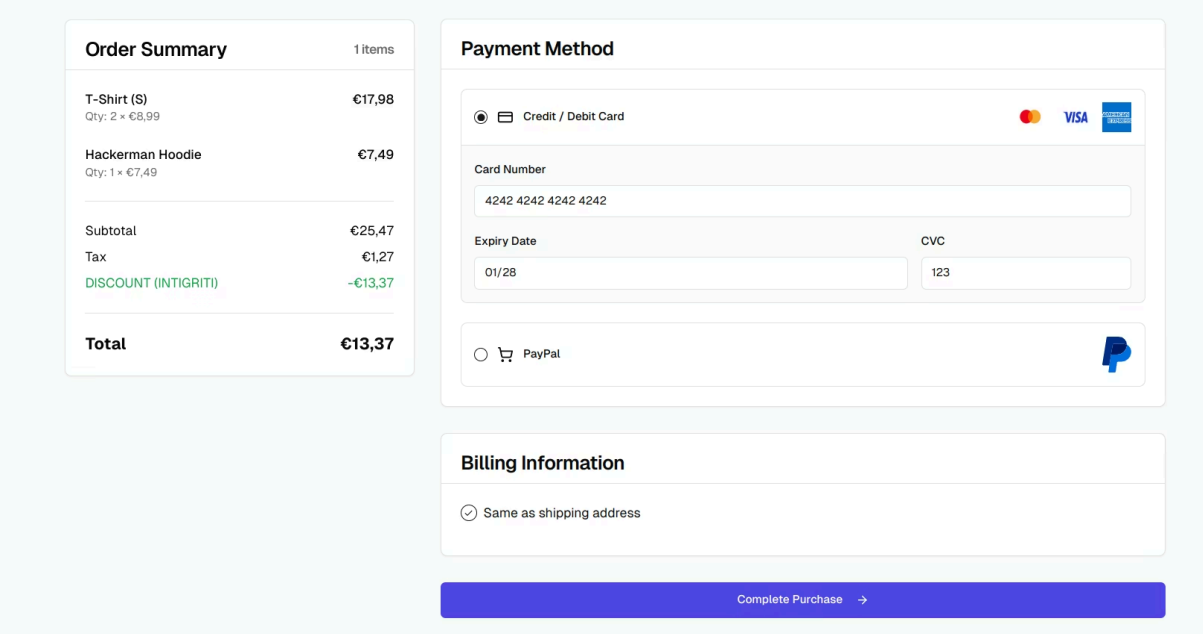
E-commerce platforms process millions of transactions daily, roughly 20% of all purchases made worldwide are happening online. Unfortunately, not every e-commerce target receives the same security attention as others. The chances of encountering vulnerable price manipulation flaws in e-commerce bug bounty programs are still probable. However, many bug bounty hunters get intimidated as e-commerce targets often implement complex checkout flows and discount systems. For this reason, most skip thoroughly testing these implementations.

In this article, we will cover several exploitation techniques to identify price manipulation (also referred to as 'formula injection') vulnerabilities in e-commerce targets.

Let's dive in!

What are price manipulation vulnerabilities?

Price manipulation vulnerabilities are security vulnerabilities in e-commerce targets, and since the rise of Web 3.0, also in decentralized finance (DeFi) platforms, allowing malicious attackers to alter the prices of products or goods to an arbitrary value.



Example of a checkout page on e-commerce targets

Price manipulation vulnerabilities can result in financial losses for the affected company and often stem from the following root causes:

- Formula injection
- Security misconfigurations
- Logic error vulnerabilities

Let's take a look at each one of these in detail.

Formula injection

Formula injection is a security flaw whereby unvalidated user input is directly concatenated in mathematical calculations or functions that determine the price of a certain item or service.

Depending on where and what is possible to inject, a malicious user may alter the calculation to reduce the price, or in the worst case, even set it to 0, allowing him/her to, for example, order an item for free.

Security misconfigurations

Most e-commerce companies make use of third-party payment system services. When checkout services are incorrectly implemented or configured, they could allow malicious actors to bypass the paywall altogether.

Logic error vulnerabilities

Developers may also introduce new changes or create new functionalities to cater to the e-commerce's needs (e.g., coupons, seasonal discounts, discounts on quantity, etc.). These new functionalities are not always safely designed and might be prone to logic error vulnerabilities that result in payment bypasses.

In this article, we will cover all 3 cases.

Exploiting price manipulation vulnerabilities

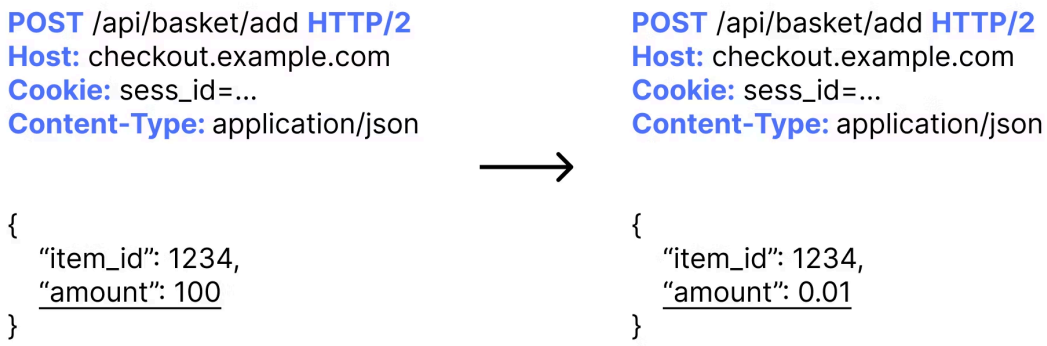
Checkout systems are used to allow customers to order goods and services online. When incorrectly implemented or configured, these may be prone to price manipulation vulnerabilities, allowing bad actors to bypass this paywall altogether and order items at a discounted rate or even for free.

Let's take a look at common price manipulation vulnerabilities.

1) Formula injection via price tampering

Formula injection occurs when arbitrary user input is a direct part of the mathematical calculation that is later used to determine the price. Payment systems that do not check the integrity of the item price are prone to these types of attacks.

A malicious user could easily change, for example, the "amount" parameter in a checkout request to any arbitrary number, such as "0.01," allowing him/her to order items at highly discounted prices.



Formula injection (price tampering) to manipulate price in e-commerce websites

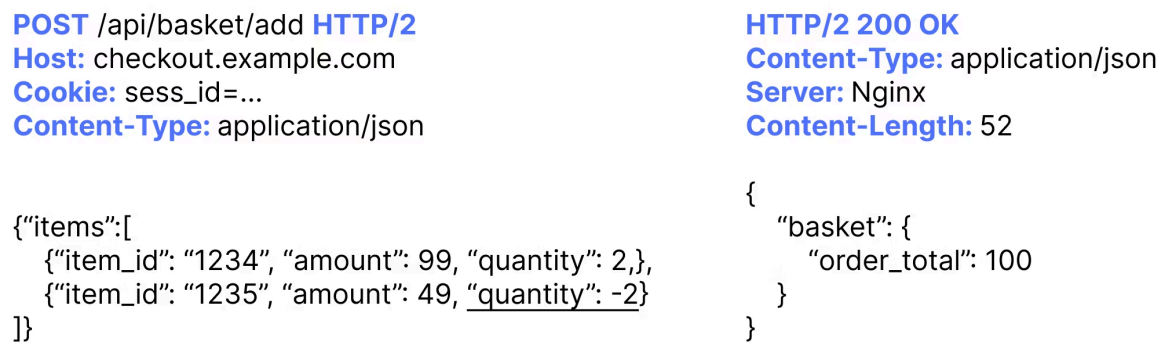
Let's take a detailed look into another type of formula injection.

2) Formula injection via quantity tampering

Another part of the order price calculation is the quantity of services or items. When payment systems do not correctly validate the quantity, it might allow us to again order items at discounted prices.

Negative quantity

If no validation is performed, we can set the quantity to a negative value to tamper with the formula that is used to determine our order price. Take a look at the following example:



Formula injection (quantity tampering) to manipulate price in e-commerce websites

We ordered 2 different items. The first item has a positive quantity and a total sum of 400 USD. The second item in our basket has a negative quantity. If no proper validation was performed, we could essentially subtract the sum amount of the second item (100 USD) from the order price (500 USD).

This would allow us to pay only 300 USD instead of the full 500 USD.

Note

Quantity tampering may not always work depending on how the value is validated and later on processed. A negative or decimal quantity can negatively affect the number of ordered items, sometimes resulting in zero items ordered while still paying an arbitrary amount.

Decimal quantity

Another way to test if quantities are properly validated is by sending a decimal value. If no validation is performed, we may be able to tamper with the order amount again by ordering 2 different items just as before:

```
POST /api/basket/add HTTP/2
Host: checkout.example.com
Cookie: sess_id=...
Content-Type: application/json

{"items":[
   {"item_id": "1234", "amount": 99, "quantity": 2,},
   {"item_id": "1235", "amount": 49, "quantity": 0.9}
]}
```

```
HTTP/2 200 OK
Content-Type: application/json
Server: Nginx
Content-Length: 55

{
   "basket": {
      "order_total": 242.10
   }
}
```

Formula injection (quantity tampering with decimal) to manipulate price in e-commerce websites

Tip!

Want to go more in-depth into some common price manipulation vulnerabilities like the ones we mentioned above? Check out @irsdl's research paper on Common Security Issues in Financially Oriented Web Applications!

3) Integer overflow

Most machines can only count or store a certain number, and when they go past that number, they wrap around and start counting from the beginning. Integer overflow attacks take advantage of this limitation.

If you set an amount or quantity to a very huge number that the backend can't handle, it will simply convert it to a negative number or reset it to 0 (depending on the underlying technologies used).

Next time when testing checkout systems, try altering the price or quantity of your ordered items and set them to an excessively high number. If no validation is present, it may reset your quantity or price to a negative number or simply set it to 0, allowing you to bypass the checkout system altogether.

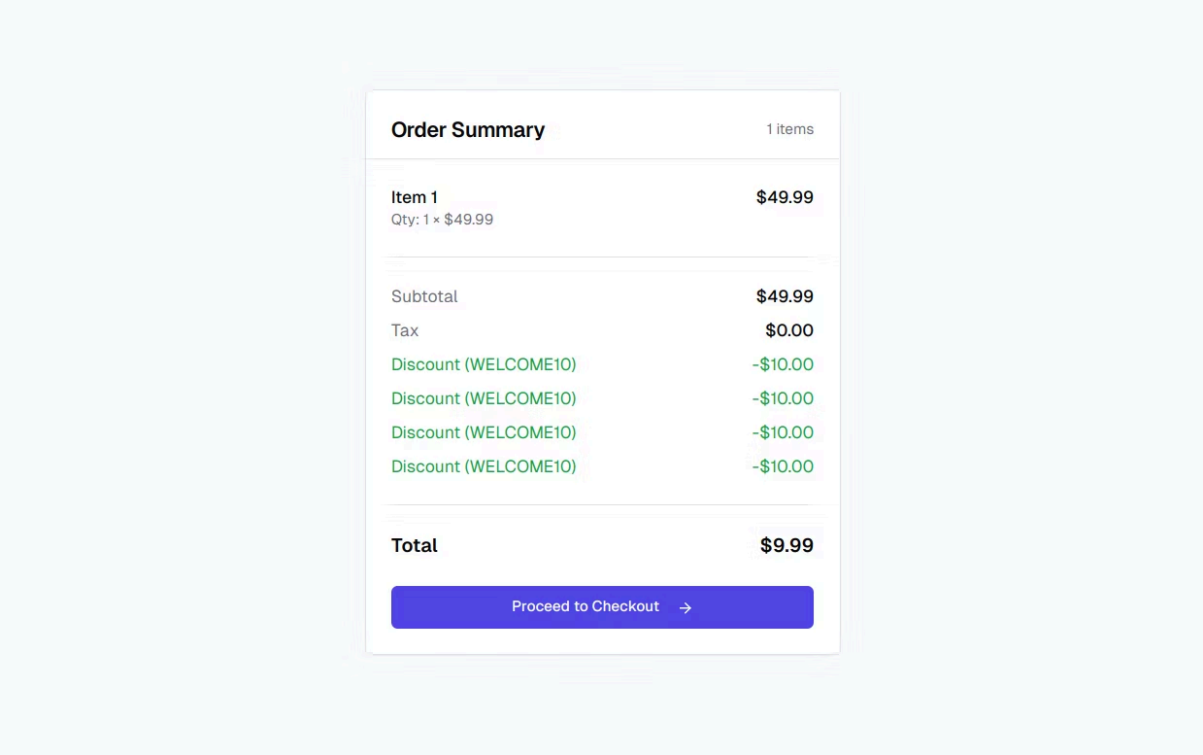
4) Coupons

Coupons and discounts are widely used on e-commerce websites as they're an integral part of marketing and sales. When incorrectly set up, coupons can introduce a way for malicious users to alter the order

price to an arbitrary value. Let's take a look at a few logic errors that could help us bypass an insecure payment wall.

Multiple redemptions

Coupons are almost always only redeemable once per customer. Lack of validation can allow us to redeem the same coupon multiple times. Especially when the coupon validation system is prone to race condition vulnerabilities!



Example of a logic error vulnerability in coupon redeem systems

Lack of expiration

Seasonal coupons (such as limited promotions, Black Fridays, etc.) are all temporary coupons. Even in this scenario, if no validation is performed on the coupon, you should be able to use coupons and discounts that were issued a few months or even years ago.

Personal coupons

Some e-commerce targets send you personal gifts in the form of coupons on your birthday. Lack of validation can allow malicious users to change the birthday on their profile multiple times to receive multiple coupons.

This bug bounty tip shared by @intidc doubles down on how logic errors can be exploited:

5) Currency confusion

Support for multiple currencies is often implemented by e-commerce companies to cater to the needs of customers. Currency confusion is an attack vector whereby we, as malicious users, take advantage of exchange rate differences between different currencies to order items at a low price.

Currency confusion stems from a lack of validation. Whenever inspecting an HTTP request, check if you can alter the currency from USD to, for example, INR or JPY while leaving the price or amount parameter untouched.

If your ordered item is priced at 100 USD, you'd eventually pay 100 INR, the equivalent of 1.10 USD, instead.

```
POST /api/basket/add HTTP/2
Host: checkout.example.com
Cookie: sess_id=...
Content-Type: application/json

{
   "item_id": 1234,
   "amount": 100,
   "currency": "USD"
}
```

→

```
POST /api/basket/add HTTP/2
Host: checkout.example.com
Cookie: sess_id=...
Content-Type: application/json

{
   "item_id": 1234,
   "amount": 100,
   "currency": "INR"
}
```

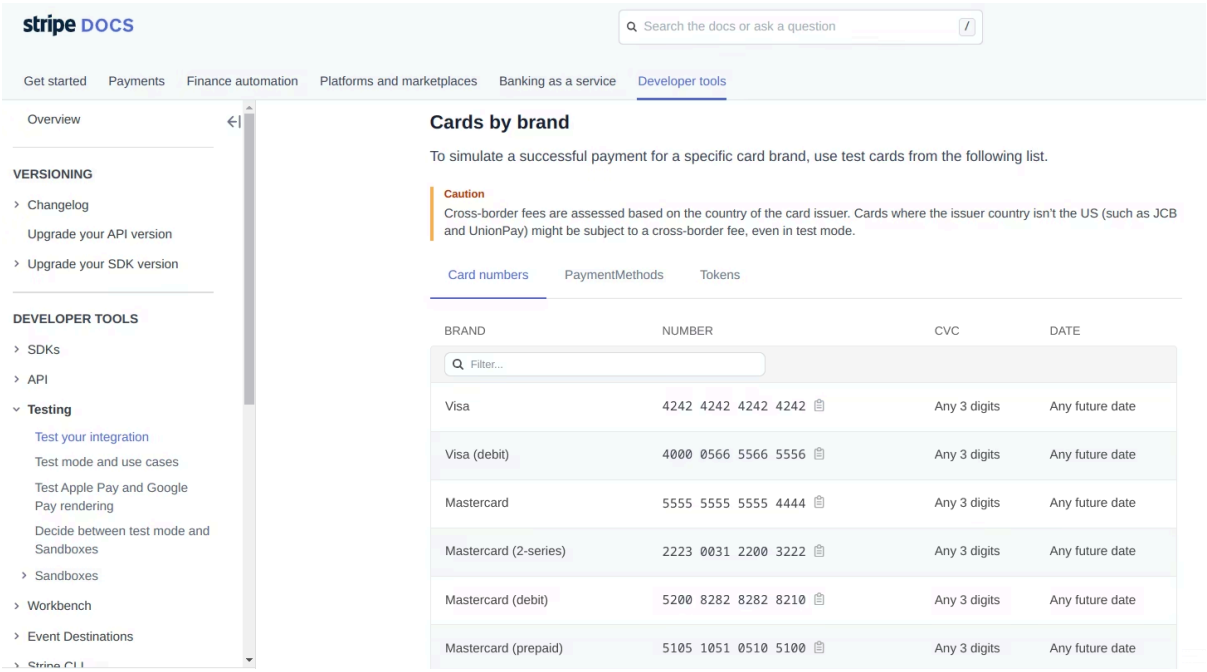
Currency confusion to manipulate price in e-commerce websites

6) Accepted test cards

Just as with all software, testing is an important part of any development phase. For checkout systems, developers often rely on test cards to ensure that their integration works without any issues. Test cards are valid card credentials and function as fake credit cards that don't incur any charges.

These cards should only be accepted in testing and development environments. However, it sometimes happens that these fake cards are still accepted in production. It's always recommended to include testing for accepted test cards.

Enumerate the third-party payment provider that the company is using through embedded forms or JavaScript files and browse through the provider's documentation for test cards. Stripe, Adyen, and 2Checkout all provide testing credentials that you can try on your target!



Stripe documentation providing test credentials

Conclusion

Online purchases keep growing year on year, and e-commerce businesses are on the rise. Unfortunately, not every e-commerce company provides the necessary security attention that their platform deserves. In this article, we went over several ways to exploit price manipulation vulnerabilities and checkout systems in e-commerce targets.

You've just learned how to hunt for vulnerabilities in e-commerce targets... Right now, it's time to put your skills to the test! Browse through our 150+ public bug bounty programs on Intigriti, and who knows, maybe your next bounty will be earned with us!

START HACKING ON INTIGRITI TODAY

REQUEST A DEMO

intigriti.com/demo

VISIT THE WEBSITE

intigriti.com

GET IN TOUCH

hello@intigriti.com