Powered Web Application Pentesting: E-Commerce

Chintan Gurjar

Buy now testing

Tamper product ID to purchase other high valued product with low prize

Tamper product data in order to increase the number of product with the same prize

Gift/Voucher/Coupon Code testing

Tamper gift/voucher count in the request (if any) to increase/decrease the number of vouchers/gifts to be used.

Tamper gift/voucher value to increase/decrease the value of voucher in terms of money. (e.g. $100 is given as a voucher, tamper value to increase, decrease money).

Reuse gift/voucher by using old gift values in parameter tampering.

Check the uniqueness of gift/voucher parameter and try guessing other gift/voucher code.

Use parameter pollution technique to add same voucher twice by adding same parameter name and value again with & in the BurpSuite request.

Forceful use of coupon that are not applicable for specific products.

Bypassing coupon terms and conditions.

Bypass coupon's validity.

Cart/Basket testing

- Tamper user id to delete products from other user's cart.
- Tamper cart id to add/delete products from other user's cart.
- Identify cart id/user id for cart feature to view the added items from other user's account.

Shipping address testing

Tamper BurpSuite request to change other user's shipping address to yours.

Try stored-XSS by adding XSS vector on shipping address.

Use parameter pollution technique to add two shipping address instead of one trying to manipulate application to send same item on two shipping address.

Payment gateway testing

Price manipulation at client side with zero or –ve values

Manipulation of third-party checksum

Price manipulation before transaction is completed

Place order testing

Tamper payment options parameter to change the payment method. E.g. Consider some items cannot be ordered for cash on delivery but tampering request parameters from debit/credit/PayPal/net banking option to cash on delivery may allow you to place order for that item.

Tamper the amount value for payment manipulation in each main and sub requests and responses.

Check if CVV is going in cleartext or not.

Check if credit/debit card details are masked or not.

Check if application itself process your card details and then perform transaction or it calls any third-party payment processing company to perform transaction.

Order tracking testing

Track other user's order by guessing tracking order number

Brute force tracking number prefix or suffix to track mass orders for other users

Wish list page testing

Check if a user A can add/remote products in Wishlist of other user B’s account.

Check if a user A can add products into user B’s cart from his/her (user A’s) Wishlist section.

Post product purchase testing

1

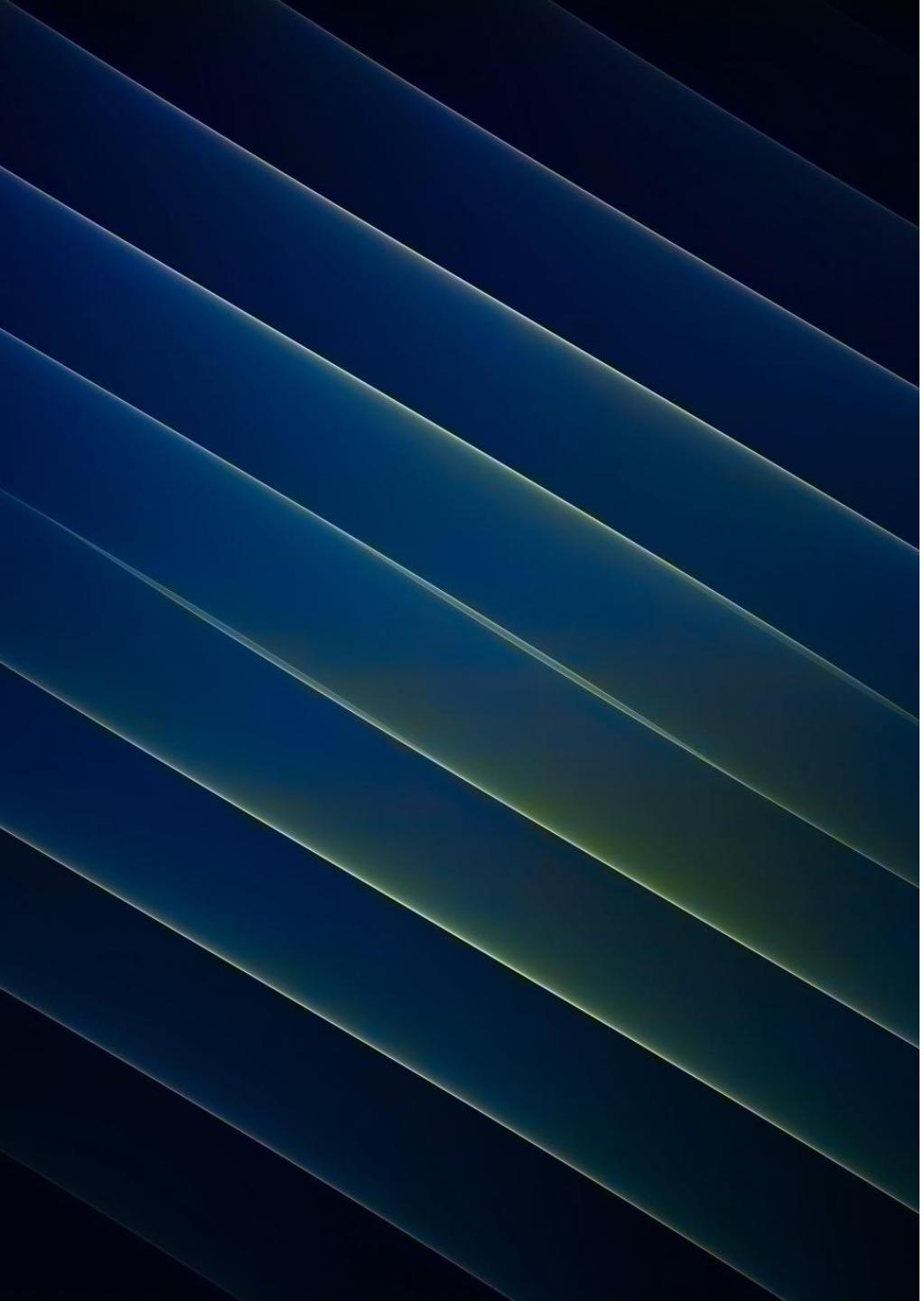
Check if user A can cancel orders for user B's purchase.

2

Check if user A can view/check orders already placed by user B.

3

Check if user A can modify shipping address of placed order by user B.



Order management testing

- Check if user can request for cash back/refund even while the order is being placed or already placed.
- Client-side validation bypass for ordering maximum number of products.
- Check whether a user book/reserve products in future using fake information.

Out of band testing

Can user order product which is out of stock?

Thank you

Say hello!

LinkedIn - https://www.linkedin.com/in/chintangurjar/

Blog – https://infosecninja.blogspot.com/

Twitter - @iamthefrogy