Business Logic

OWASP Web App Top 10

What is it?

Business Logic flaws allow attackers to manipulate the business logic of a web application to their advantage.

What causes it?

Logic flaws can be the result of coding bugs, design flaws or wrong logical assumptions made by developers during the implementation of the system.

What could happen?

Since logic flaws are application specific, impact depends on the application. Weak account validation could result in transferring more money than possible. Flaws in a checkout workflow could allow products to be ordered without paying.

How to prevent it?

Business rules should be clearly defined and checked against during the different development phases of the application: design, implementation and testing. Clear documentation and threat modelling/abuse cases and code reviews should be used.

Business Logic

Understanding the security vulnerability

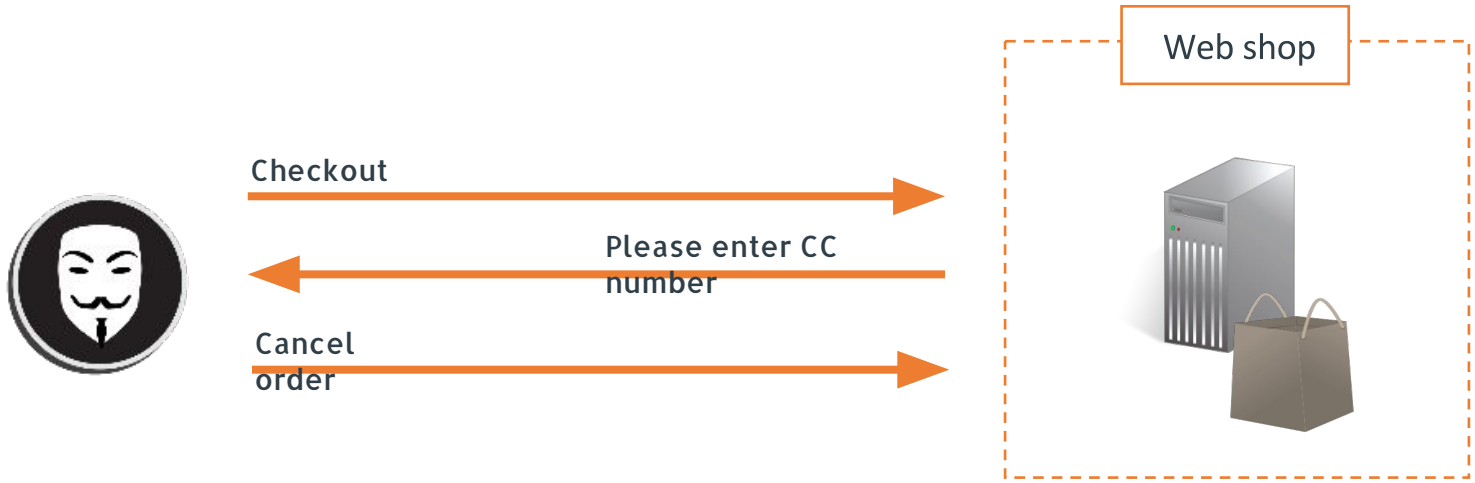
Flawed order cancellation

An attacker is connected to an e-shop where he buys a number of items.

When finished, he proceeds to the checkout page.

When presented the payment page, the attacker cancels his order.

The money is not withdrawn, but because of a logic flaw, the items are still sent to the attacker.



Business Logic

Understanding the security vulnerability

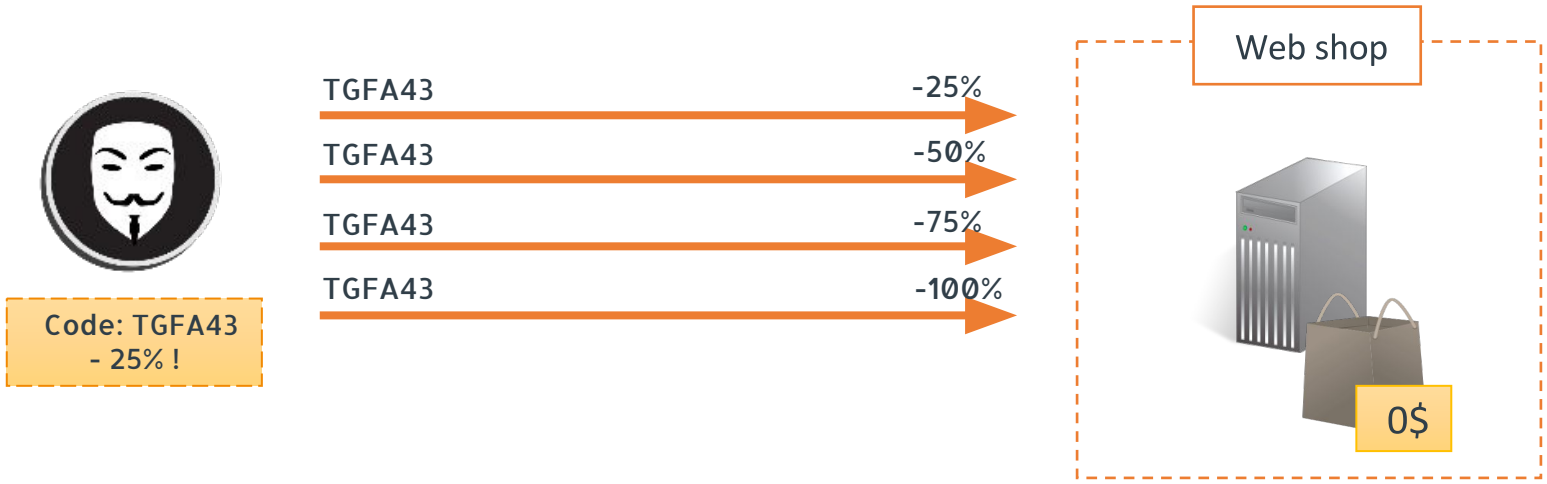
Reuse of discount coupons

An attacker is logged into an e-commerce site. He has a 25% reduction coupon.

The attacker buys products and at the payment screen, he uses the coupon.

However, because of a flaw, he is able to reuse the code multiple times, giving him 100% reduction.

He checks out and gets his order for free.



Business Logic

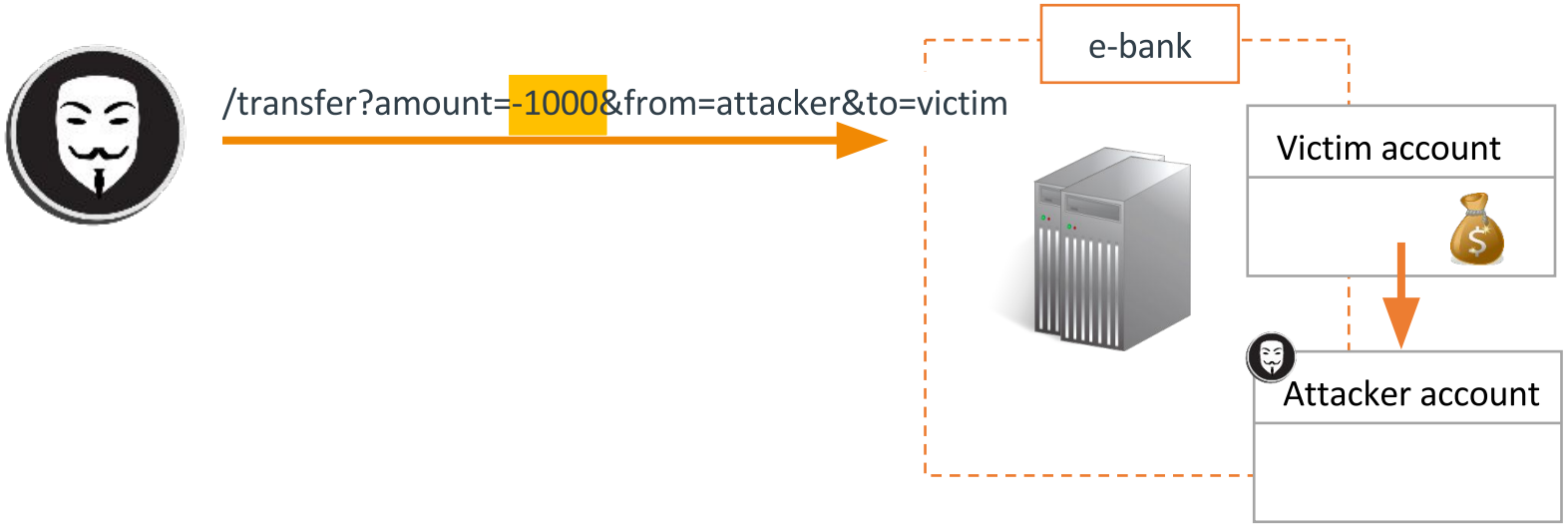
Understanding the security vulnerability

Increasing bank balance

An attacker is logged into a bank where he has a bank account.

He transfers a negative amount to the account of a victim.

The negative transfer is wrongly interpreted and the amount is transferred from the victim to the attacker's account instead.



Business Logic

Realizing the impact

The impact of business logic varies from application to application but is typically high.

A flawed checkout mechanism could lead to theft and result in reputational and financial damages.

A defective transaction mechanism could allow unlimited transfers, resulting in financial damages.

Business Logic

Preventing the mistake

Use threat modelling to help identify design flaws.

Create security tests based on abuse cases and transaction flow analysis.

Document the design of the application.

Design assumptions should be clearly stated.
Use data/transaction flows diagrams.

Make the application's design abuse resistant.

Perform security review and tests.