

## Security Testing of a Jewellery Online Store

### Customer

The customer is a software development company that developed a website for online jewellery sales.

<b>Company</b>	<i>Software Development Company</i>
<b>Country</b>	<i>USA</i>
<b>Business Domain</b>	<i>E-Commerce</i>
<b>Services Used</b>	<i>Security and Functional Testing</i>
<b>Cooperation Model</b>	<i>QA Outsourcing for a Development Company</i>
<b>Duration</b>	<i>1 month</i>
<b>Efforts</b>	<i>40 man-hours</i>

### Project

The project concerned an online jewellery store. The customer had a complete application ready for use that had undergone functional testing by the in-house QA team.

There were several reasons the customer made a request for security and functional testing. The first one was frequent complaints from clients who couldn't complete the registration process, and the company's internal team was unable to reproduce the issue. The second one was the necessity of executing a full security test that required wider coverage than could be procured internally.

### Challenge

The first task of A1QA was to perform functional testing to locate registration bottlenecks and double-check the entire application for any possible omitted issues. Another important task was to run full security testing considering the specifics of the site (the checkout and payment processes, managing user profiles including administrators, confidentiality of orders data, etc.).

Within the framework of the project A1QA offered the following services:

1. Complex security testing
  - To check security of the checkout process.
  - To check security of the registration process and profile management.
  - To check validation in all modules.
2. Complex functional testing
  - To provide a list of particular cases that uncover problems with registration.
  - To extend the existing scope of functional testing to ensure that the remaining part of the functionality is defect-free.

### Solution

Our QA engineers employed a wide range of carefully selected test activities, the most important of which are described below:

1. Checking the possibility of unauthorized modification of data by hand or using special tools:
  - Changing the order amount during checkout.
  - Changing the order status during checkout.
2. Using SQL injection, XSS injection, HTML injection, and GET/POST methods of sending data, to check the following possibilities:
  - Skipping any of the checkout steps.

- Skipping any of the registration steps.
  - Breaking the confidentiality of clients personal data.
  - Breaking the confidentiality of clients private credit data.
3. Validation checking:
    - Leaving compulsory fields empty.
    - Buffer overflow.
    - Special symbols input.
    - Invalid symbols input.
  4. Checking password hashing.
  5. Checking if error handling is proper and if there is no possibility to use information in error messages for attacks.

## Technologies used

**Operating systems:** WinXP

**Browsers:** I.E 6.0, Firefox 2.0

**Security tools:** Tamper Data, Hack Bar, Web Developer Toolbar, Regular Expression Tester, AddNEdit Cookies, Modify Headers, ShowIP

**Defect tracking system:** Rational ClearQuest

## Success

- The critical problem that didn't allow users to register was localized, and the customer was provided with a set of exact cases reproducing the problem.
- The extended scope of functional testing helped reveal defects that were missing in the previous internal QA reports.
- As a result of security testing, multiple vulnerabilities of various types were found, the majority of which concerned payment and checkout steps. Having such kind of defects on the production could bring damage , including material losses.
- The security check of user permissions revealed non-trivial methods for getting access to the hidden part of functionality.
- In the end, as a result of our work, the customer received a complete list of defects.