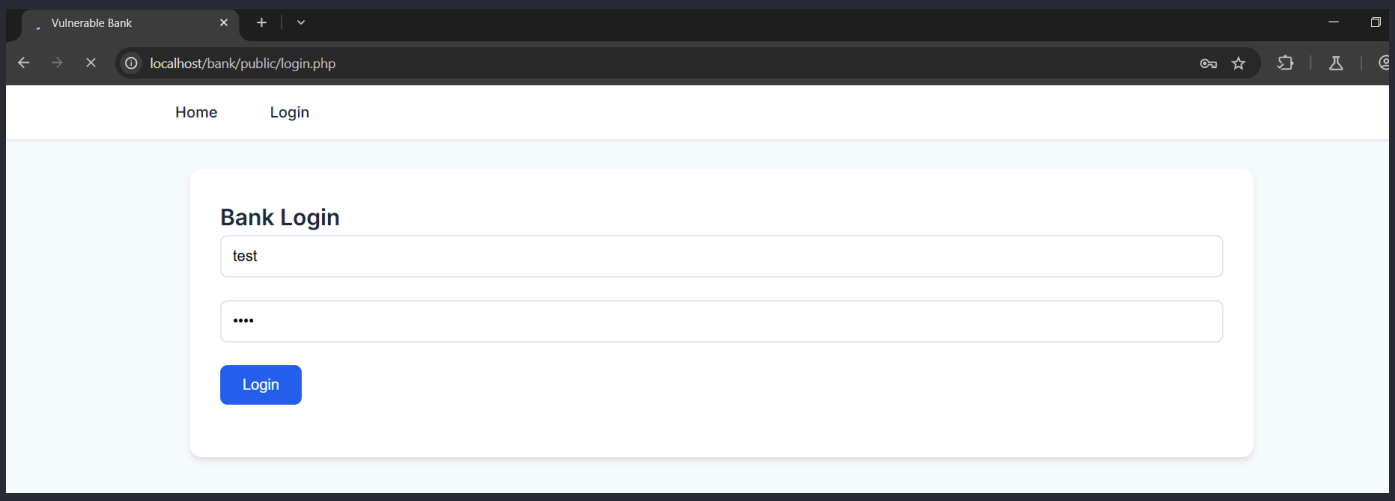
Brute Force Attack Report

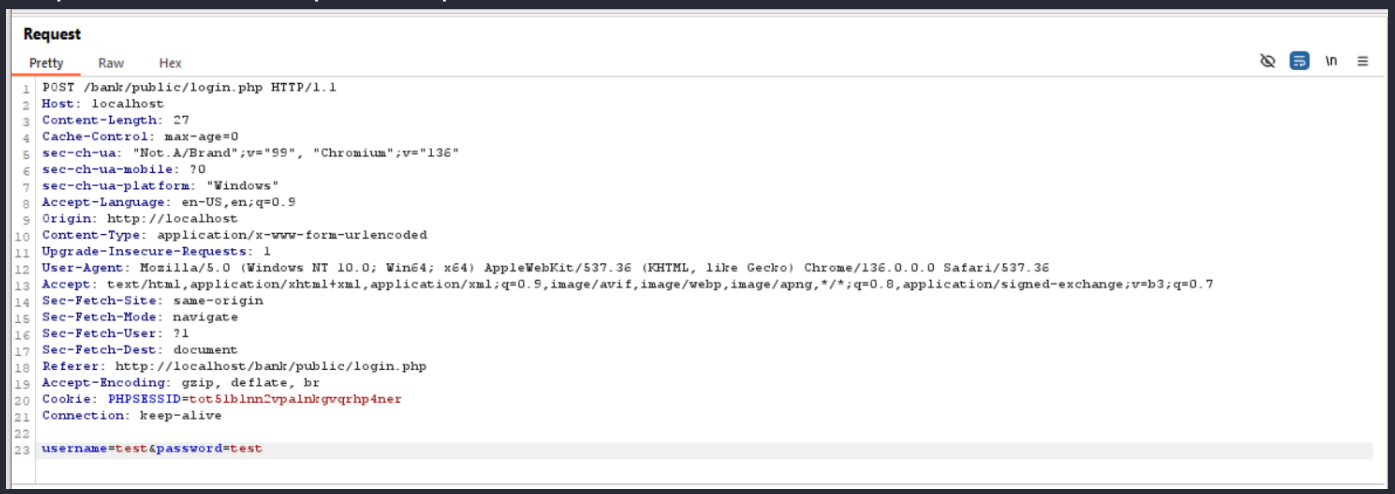
Part 1: Performing the Brute Force Attack

Capturing the Login Request

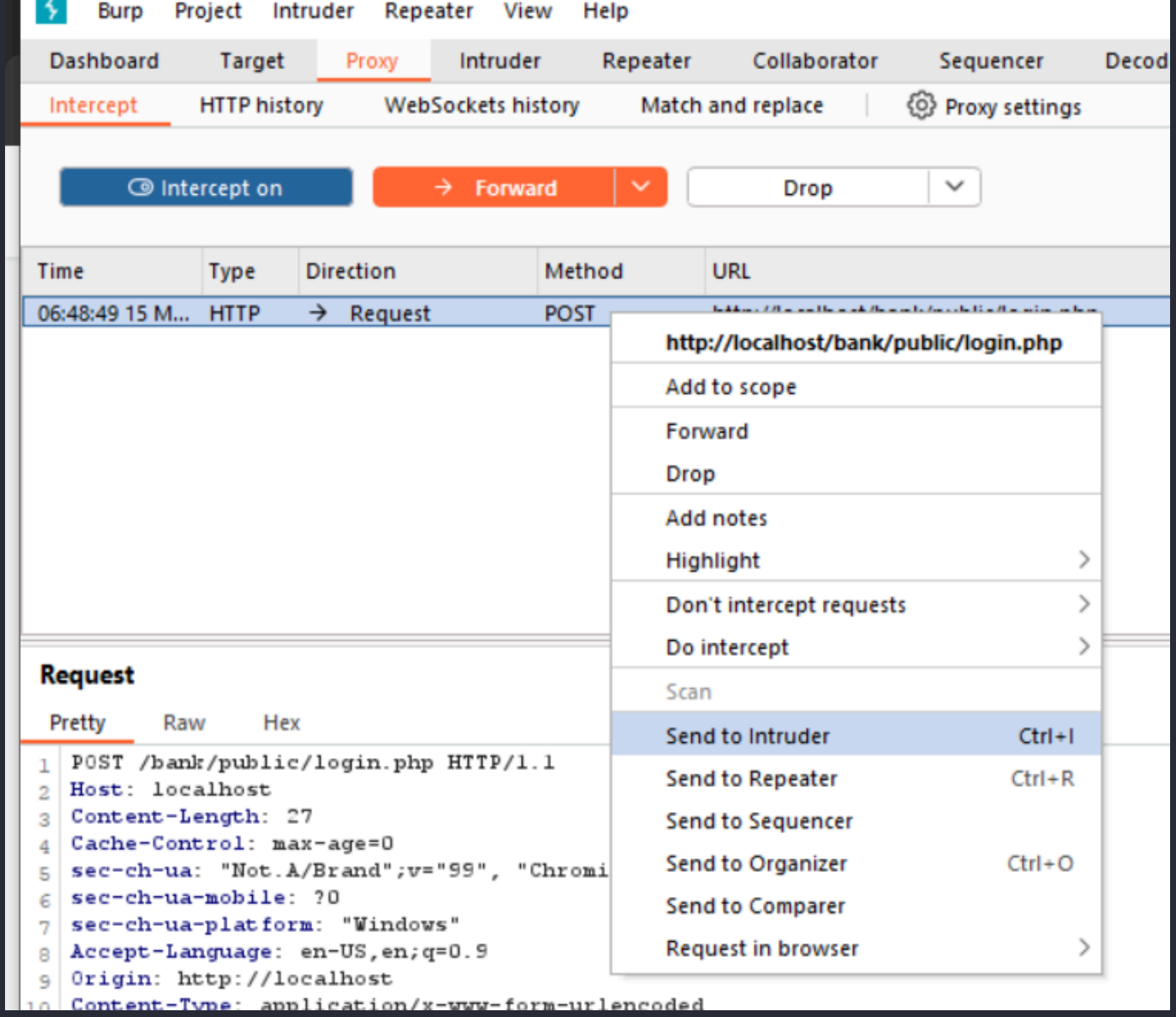
1. Navigate to the Vulnerable Bank login page: `http://localhost/bank/public/login.php`
2. Enter test credentials
   - Username: `test`
   - Password: `test`



3. Click the "Login" button .
4. Burp Suite will intercept the request.

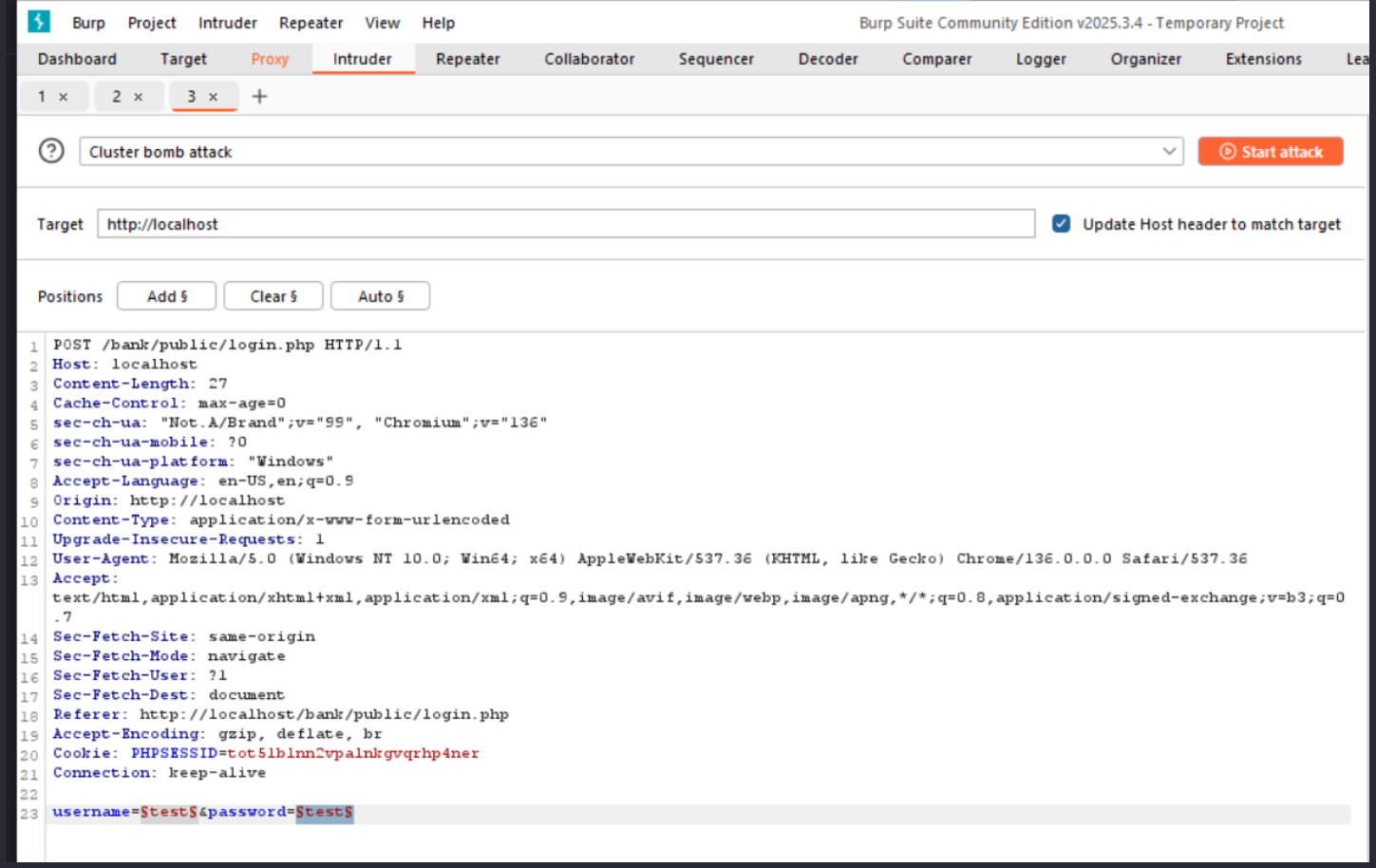


5. Right-click on the intercepted request and select "Send to Intruder"



Setting Up the Intruder Attack

- Add placeholders for request parameters .
- Select Cluster Bomb Attack .



Loading Wordlists

- Specifying payload for username .
- Specifying payload for password .

Payloads

Payload position: 1 - test

Payload type: Simple list

Payload count: 7

Request count: 35

Payload configuration

This payload type lets you configure a simple list of strings that are used as payloads.

Paste

Load...

Remove

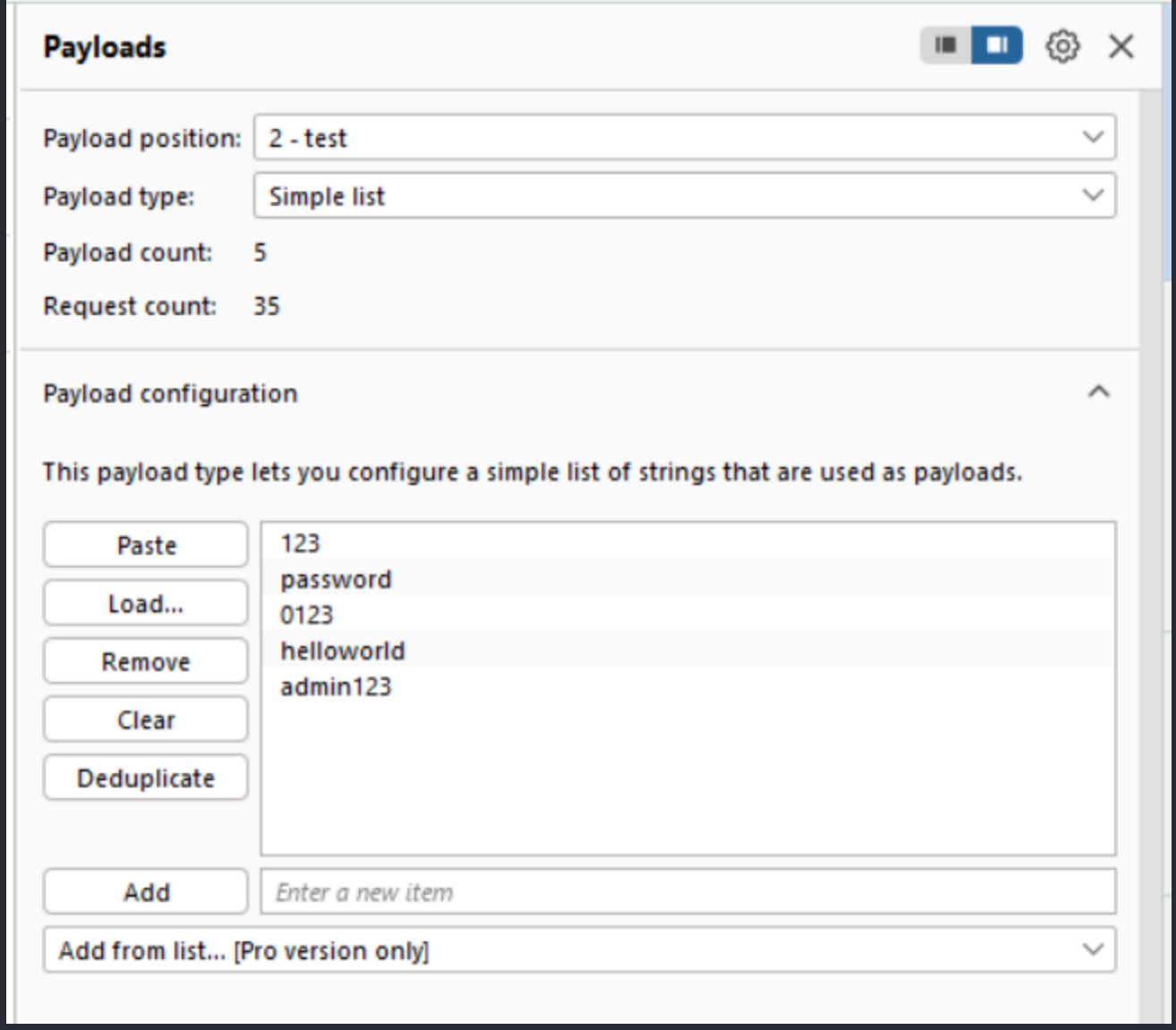
Clear

Deduplicate

admin123
user1
user2
user3
user4
admin

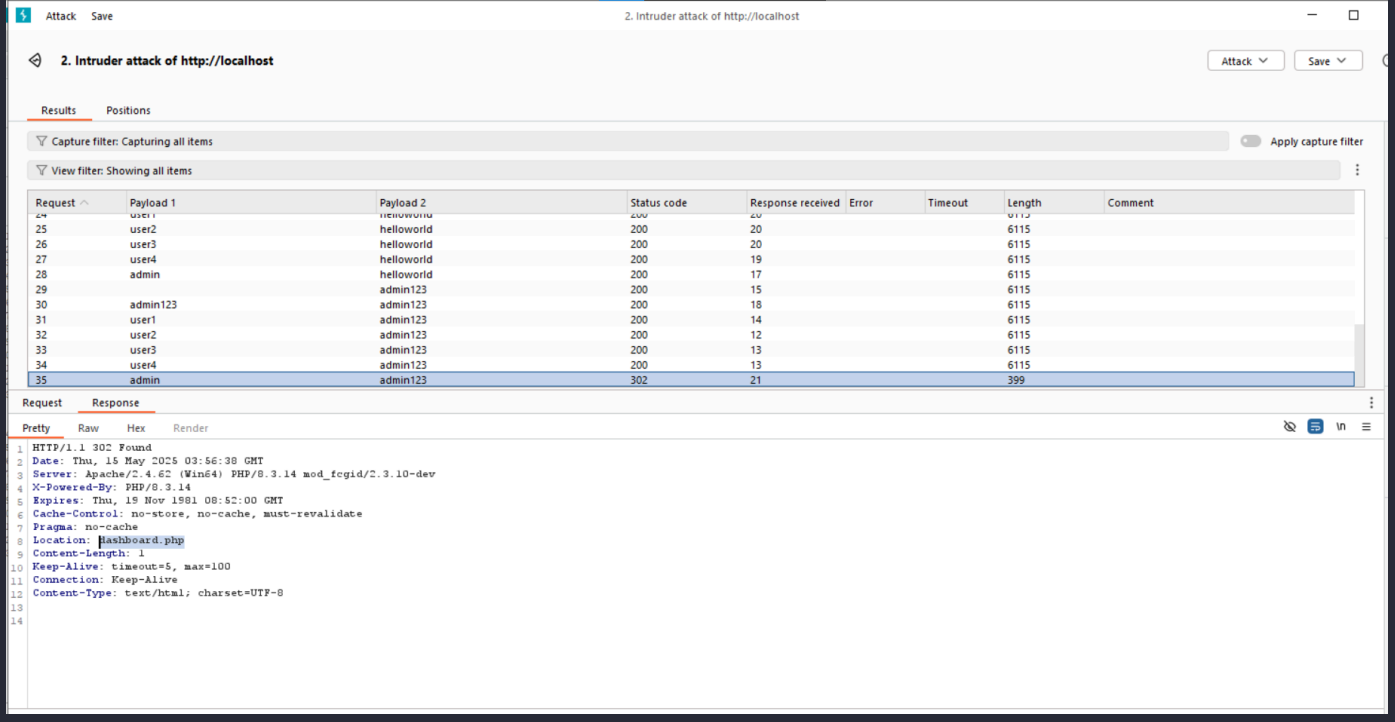
Add — *Enter a new item*

Add from list... [Pro version only]



Running the Attack

- Successful login attempt will have a different status code .
- Redirection to `dashboard.php` page .



Part 2: Implementing Security Measures

1. Failed Login Attempts Tracking

- Create a database table to log all login attempts, including successful and failed ones:

```
// Create table if it doesn't exist
$create_table = "CREATE TABLE login_attempts (
    id INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(50) NOT NULL,
    ip_address VARCHAR(45) NOT NULL,
    success TINYINT(1) DEFAULT 0,
    attempt_time DATETIME NOT NULL
)";
// Track failed attempts
function checkLoginAttempts($conn, $username, $ip) {
    // Check if there are too many failed login attempts
    $lockout_time = 15 * 60; // 15 minutes lockout
    $max_attempts = 3; // Max 3 failed attempts
```

```
    $query = "SELECT COUNT(*) as attempts FROM login_attempts
              WHERE (ip_address = '$ip' OR username = '$username')
              AND success = 0
              AND attempt_time > DATE_SUB(NOW(), INTERVAL $lockout_time SECOND)";
    $result = mysqli_query($conn, $query);
    $data = mysqli_fetch_assoc($result);
    return $data['attempts'] >= $max_attempts;
}

function logLoginAttempt($conn, $username, $ip, $success) {
    $success_val = $success ? 1 : 0;
    $query = "INSERT INTO login_attempts (username, ip_address, success, attempt_time)
              VALUES ('$username', '$ip', $success_val, NOW())";
    mysqli_query($conn, $query);
}
```

2. Account Lockout Mechanism

- temporary account lockout after a specific number of failed login attempts:
-

```
// Account Lockout after multiple failed attempts
if (checkLoginAttempts($conn, $username, $ip)) {
    $error = "Too many failed login attempts. Please try again later.";
} else {
    // Process login attempt
    // ...
}
```

Part 3: Verifying the Fixes

- Failed Attempt Even Though username and password are right .
- no redirection for `dashboard.php` .
- `login_attemps` table in database .