Account Enumeration Vulnerability: `bet365.ee`

Gregor Eesmaa
gregor.eesmaa@ut.ee
University of Tartu

May 31, 2025

1 Introduction

Account enumeration is a security vulnerability enabling attackers to determine if specific user accounts exist on a service. The vulnerability usually lies in the account registration functionality of a service, where an error message is returned, indicating that a user with the specified account identifier is already registered. However, an online service can also leak this information in other, more subtle ways, which are often overlooked by software developers. For example, even without a direct message, small visual differences in responses, or slight variations in how the server behaves (like the exact data returned) for existing versus non-existing accounts, can still reveal if an account is registered.

On 2025-05-11, we reassessed `bet365.ee` and found that **the service is still vulnerable to account enumeration**.

If the account identifier of an online service is personal data (e.g. email address, personal code etc), then the fact, whether it is associated to an account, is also considered personal data. Any disclosure of personal data to third parties without a legal basis constitutes a data breach [1].

We advise you to investigate the potential data breach, and notify the supervisory authority and the affected data subjects, if necessary. Detailed guidelines for mitigating this type of flaw are available in [2].

2 Vulnerabilities Found

We tested the login form, password reset form, account registration form and email change form of `bet365.ee`. We identified security issues in all of these functionalities. The vulnerabilities found are described in more detail in subsections below.

2.1 Login Form

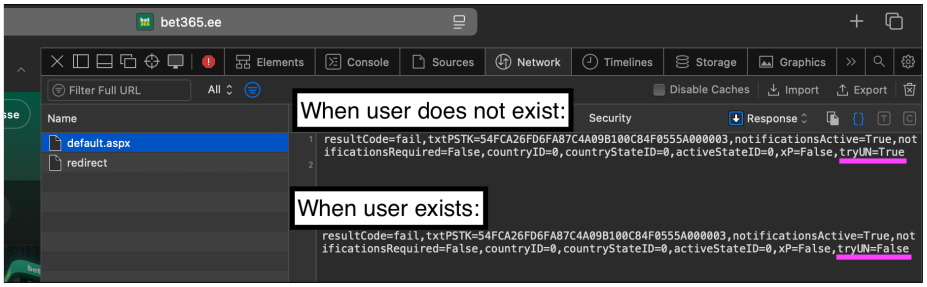

Figure 1: The vulnerability in the login form

The login form is susceptible to account enumeration attacks. This is because when a user with the provided email does not exist, there are subtle differences in the response received from the server compared to when the password is incorrect (see Figure 1). Additionally, the form appears to lack anti-bot measures such as CAPTCHA, enabling attackers to easily automate these attacks [3].

It is also crucial to eliminate any side-channels that an attacker could exploit to differentiate between account existence and non-existence. For example, the response should not be faster for an existing account than for an email with which an account does not exist.

To mitigate the flaw, the response must be uniform for both registered and unregistered email addresses. This uniformity must apply to the message displayed to the user as well as the underlying HTTP response details (like status codes, headers, and body content).

For example, the indistinguishable user-facing message could be: “Email or password is incorrect”. [2]

2.2 Password Reset Form

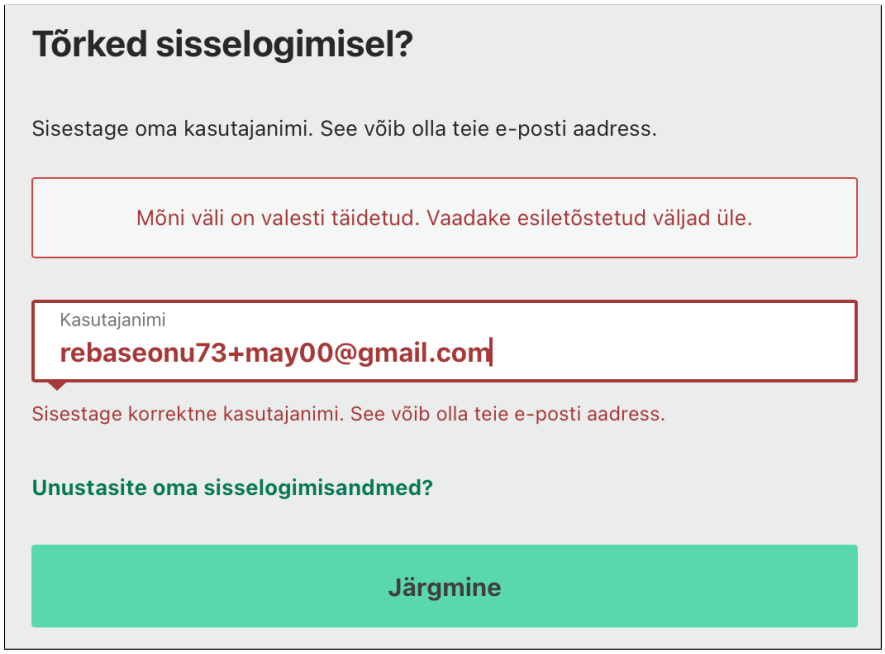

Figure 2: The vulnerability in the password reset form

The password reset form is also susceptible to account enumeration attacks. This is because when a password reset is requested for an email address that is not registered with the service, the form shows an error message (see Figure 2). Additionally, the form appears to lack anti-bot measures such as CAPTCHA, enabling attackers to easily automate these attacks [3].

No confirmation email is sent to the provided email address after this form is submitted. Additionally, validation of the email address is done in a separate request before the full form could even be submitted. These shortcomings allow the attacker to verify registered email addresses without triggering a confirmation email, thereby ensuring that the email owner remains unaware of the potential attack.

It is also crucial to eliminate any side-channels that an attacker could exploit to differentiate between account existence and non-existence. For example, the response should not be faster for an existing account than for an email with which an account does not exist.

To mitigate the flaw, the response must be uniform for both registered and unregistered email addresses. This uniformity must apply to the message displayed to the user as well as the underlying HTTP response details (like status codes, headers, and body content).

For example, the indistinguishable user-facing message could be: "A password reset link has been sent if an account with this email exists". [2]

2.3 Account Registration Form

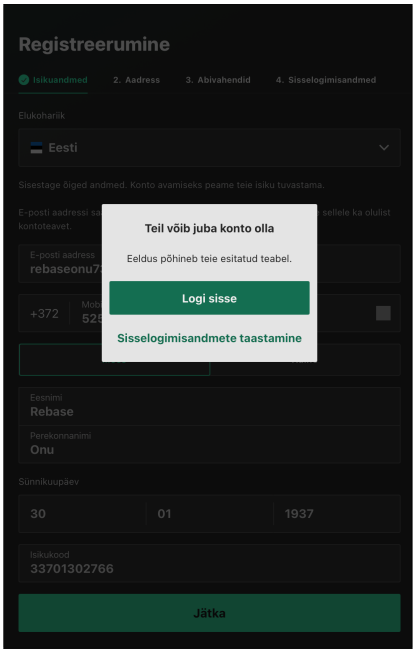

Figure 3: The vulnerability in the account registration form

The account registration form is also susceptible to account enumeration attacks. This is because when the provided email address is already taken, the form shows an error message (see Figure 3). Additionally, the form appears to lack anti-bot measures such as CAPTCHA, enabling attackers to easily automate these attacks [3].

No confirmation email is sent to the provided email address after this form is submitted. Additionally, validation of the email address is done in a separate request before the full form could even be submitted. These shortcomings allow the attacker to verify unregistered email addresses without triggering a confirmation email, thereby ensuring that the email owner remains unaware of the potential attack.

It is also crucial to eliminate any side-channels that an attacker could exploit to differentiate between account existence and non-existence. For example, the response should not be faster for an existing account than for an email with which an account does not exist.

To mitigate the flaw, the response must be uniform for both registered and unregistered email addresses. This uniformity must apply to the message displayed to the user as well as the underlying HTTP response details (like status codes, headers, and body content).

For example, the indistinguishable user-facing message could be: “We have sent further instructions to the provided email address”. Send an email in both cases, but differentiate the content based on account existence. For example, for new registration, provide means for account activation, and for existing accounts, provide means for account recovery. [2]

2.4 Email Change Form

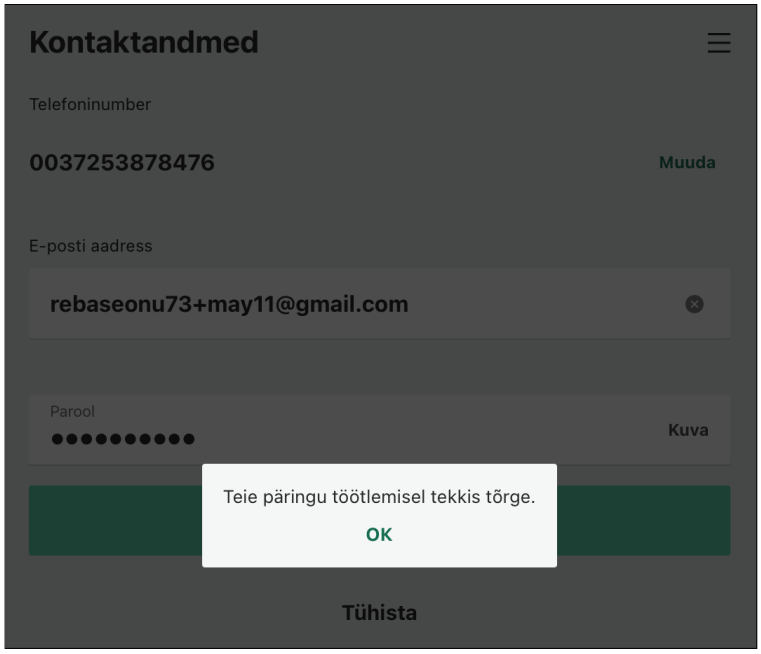

Figure 4: The vulnerability in the email change form

The email change form is also susceptible to account enumeration attacks. This is because when the provided email address is already taken, the form shows an error message (see Figure 4). Additionally, the form appears to lack anti-bot measures such as CAPTCHA, enabling attackers to easily automate these attacks [3].

Moreover, no confirmation email is sent to the provided email address after this form is submitted. This allows the attacker to verify unregistered email addresses, ensuring that the email owner remains unaware of the potential attack.

It is also crucial to eliminate any side-channels that an attacker could exploit to differentiate between account existence and non-existence. For example, the response should not be faster for an existing account than for an email with which an account does not exist.

To mitigate the flaw, the response must be uniform for both registered and unregistered email addresses. This uniformity must apply to the message displayed to the user as well as the underlying HTTP response details (like status codes, headers, and body content).

For example, the indistinguishable user-facing message could be: “We have sent further instructions to the provided new email address”. Send an email in both cases, but differentiate the content based on account existence. For example, if the email is unused, provide means for confirming the new email, but if the email is used, provide means for account recovery.

About This vulnerability report is part of an ongoing study on user enumeration vulnerabilities in Estonian online services. The study is conducted by the University of Tartu master's student Gregor Eesmaa (supervised by Arnis Paršovs - arnis.parsovs@ut.ee). The findings of this study will be published in a master's thesis scheduled for defence in August 2025.

References

[1] European Union. *General Data Protection Regulation (GDPR): Regulation (EU) 2016/679*. Official Journal of the European Union, L 119/1. 2016. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32016R0679.

[2] OWASP. *Authentication Cheat Sheet - Authentication and Error Messages*. Accessed: 2025-01-26. URL: https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html#authentication-and-error-messages.

[3] OWASP. *Authentication Cheat Sheet - Protect Against Automated Attacks*. Accessed: 2025-01-26. URL: https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html#protect-against-automated-attacks.