Security Misconfiguration

Overview

Security misconfiguration is one of the most commonly-seen security flaws in web applications. It refers to any security issue which is not a direct result of a programming error but rather a result of a configuration error. Security misconfiguration is usually a result of using default configurations or passwords, ad hoc or insufficient configurations, open cloud storage, poorly configured HTTP headers or use of detailed error messages containing sensitive information.

Security misconfigurations can occur at any level of the web application stack; from the web server or back-end database, to the platform or network services. These misconfigurations can be exploited by an attacker in order to gain unauthorized access to the system and in some cases can result in the attacker taking complete control over the system.

In this project, we will take advantage of security misconfigurations in the web app by exploiting a default account with a default password and inadvertently expose sensitive information via detailed error messages from the server.

Default Accounts with Default Passwords

One example of a security misconfiguration is use of default accounts with their default passwords still enabled and unchanged. These default passwords can be easily guessed by an attacker resulting in unauthorized system access; in fact, there are many websites which actually list default usernames and passwords for a number of different devices and applications. Common default username and password combinations include admin/admin, admin/password, root/root, user/user and user/password. In some cases, a password is not even set. It is evident how relatively simple it is for an attacker, by simply guessing the default password, can gain unauthorized access to a system.

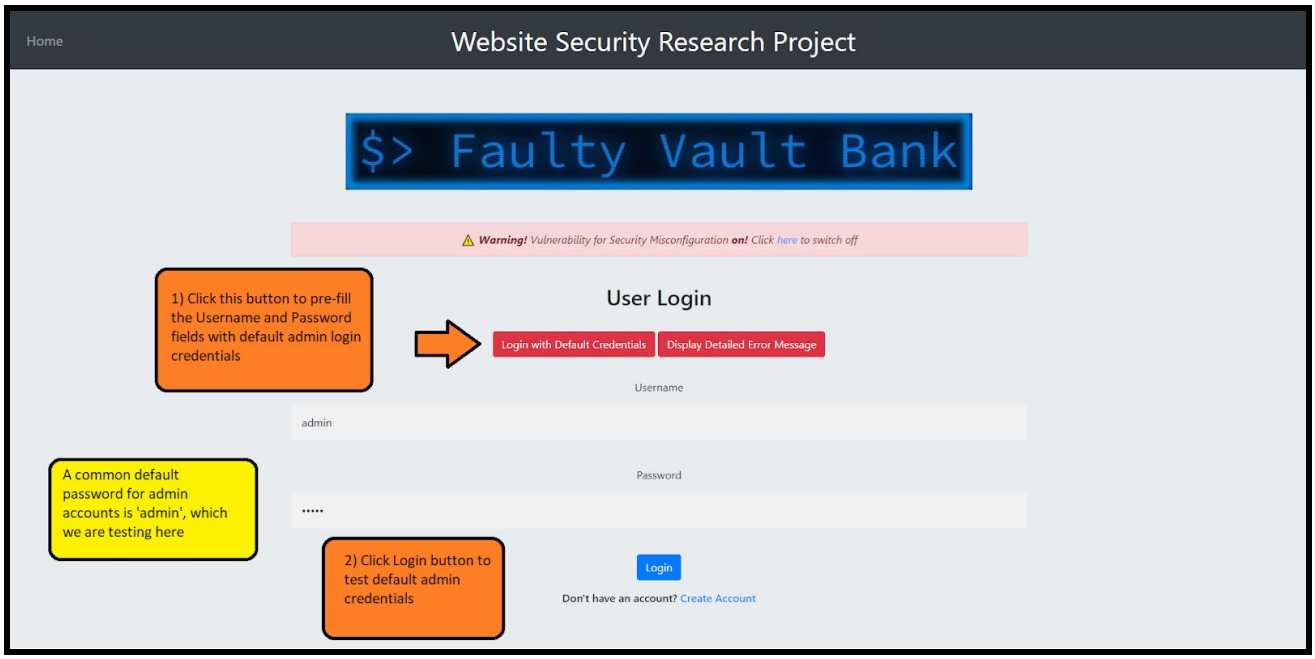
Detailed Error Messages

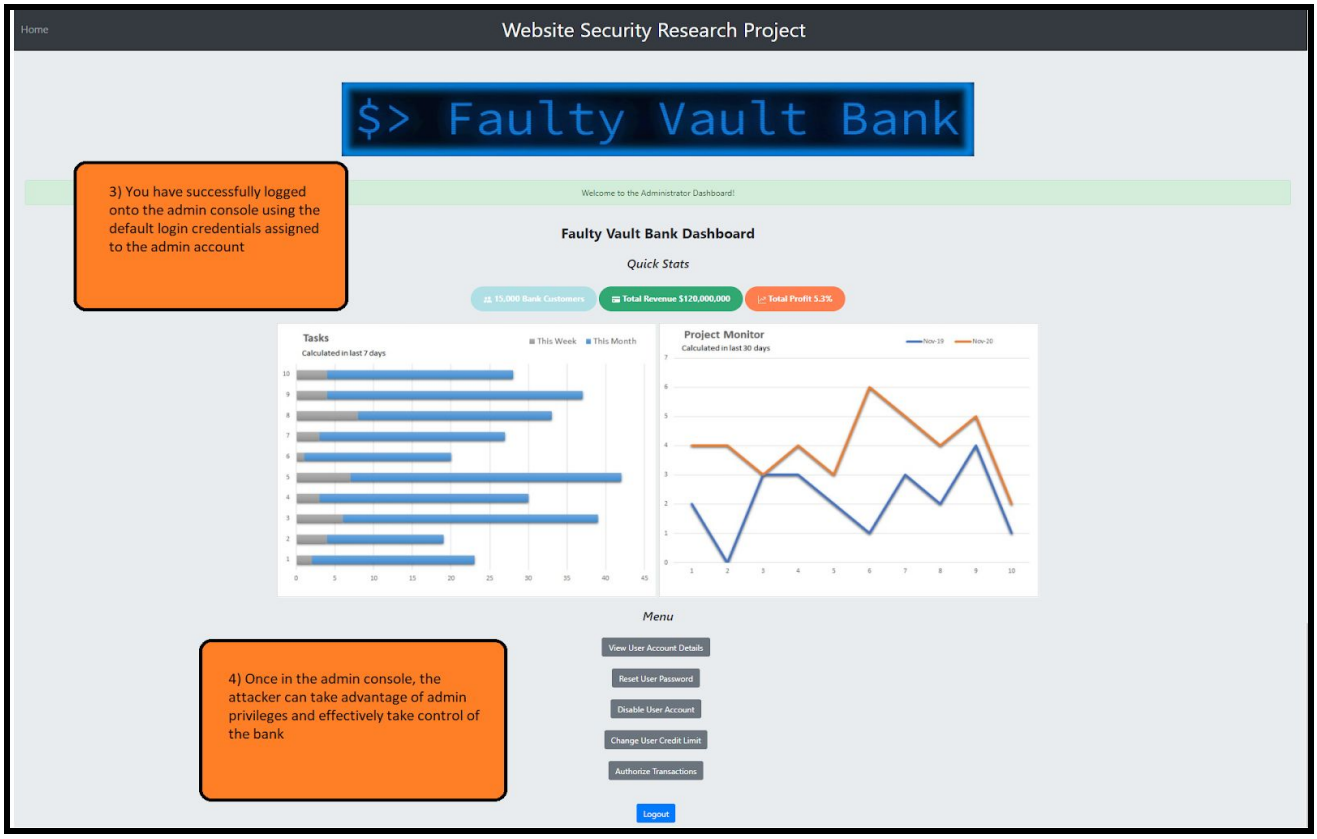
Another example of a security misconfiguration is returning a detailed error message that exposes information about the underlying server. Any information revealing software or services, versions of software, directory structure, etc… may be used to search for known vulnerabilities of the server. If the server is running a software version that hasn't been updated and has a known vulnerability, bad actors could exploit the known vulnerabilities. This is why error messages should only be returned to the user when absolutely necessary. Messages should contain premade text that informs the user of the problem. They should NEVER contain unvalidated information that could be a security risk, such as error log data.

Attack Procedures

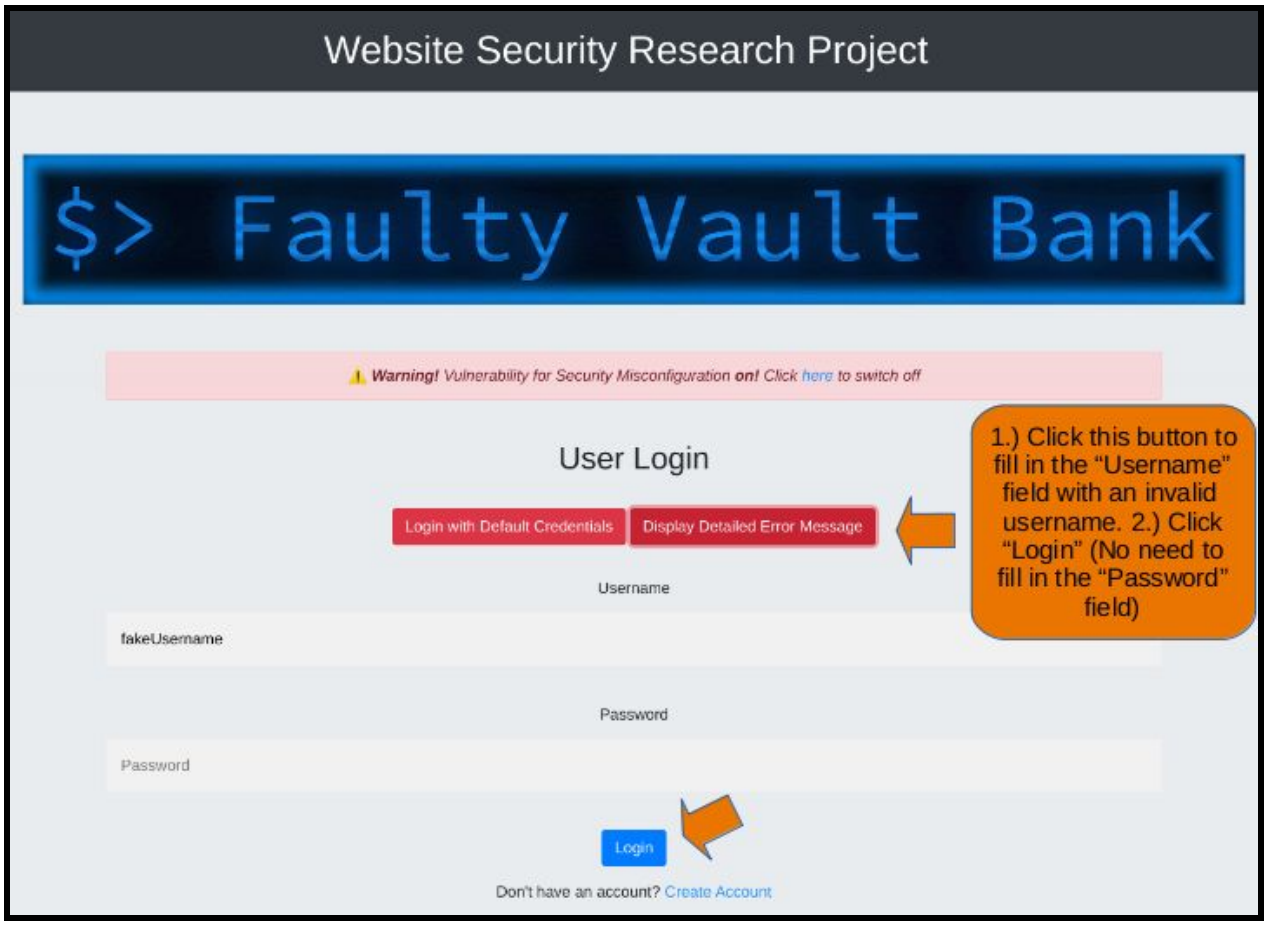
The screenshot below demonstrates the first step for all the Security Misconfiguration attack examples in this write-up. All remaining steps for each Security Misconfiguration attack can be found in their labeled sections.

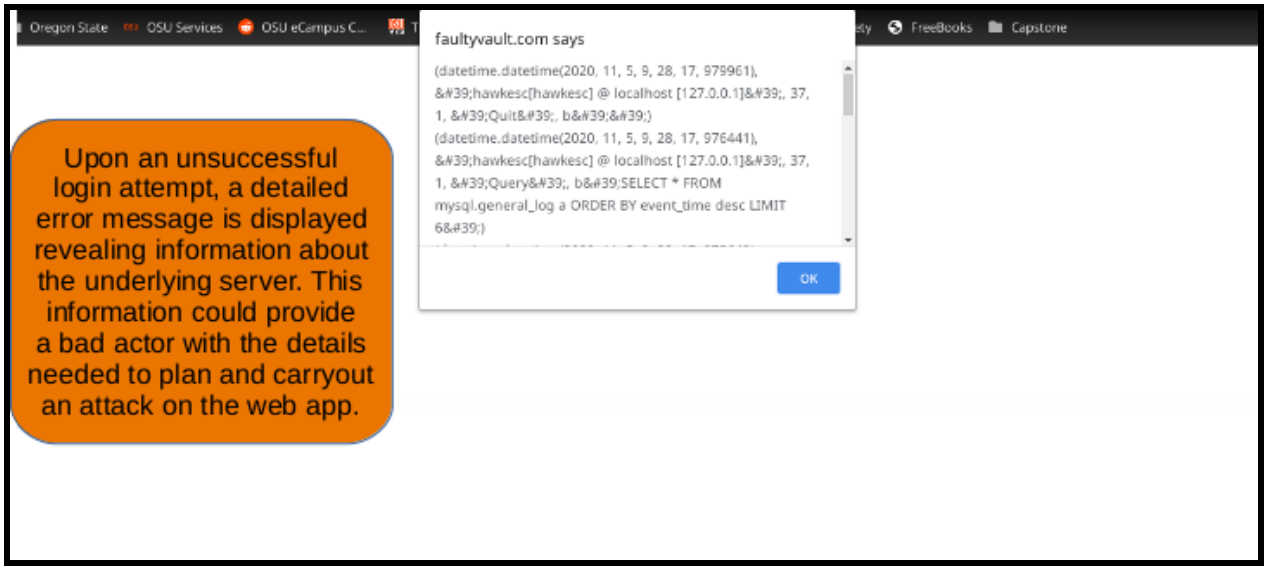
Logging On With Default Credentials





Returning Detailed Error Messages

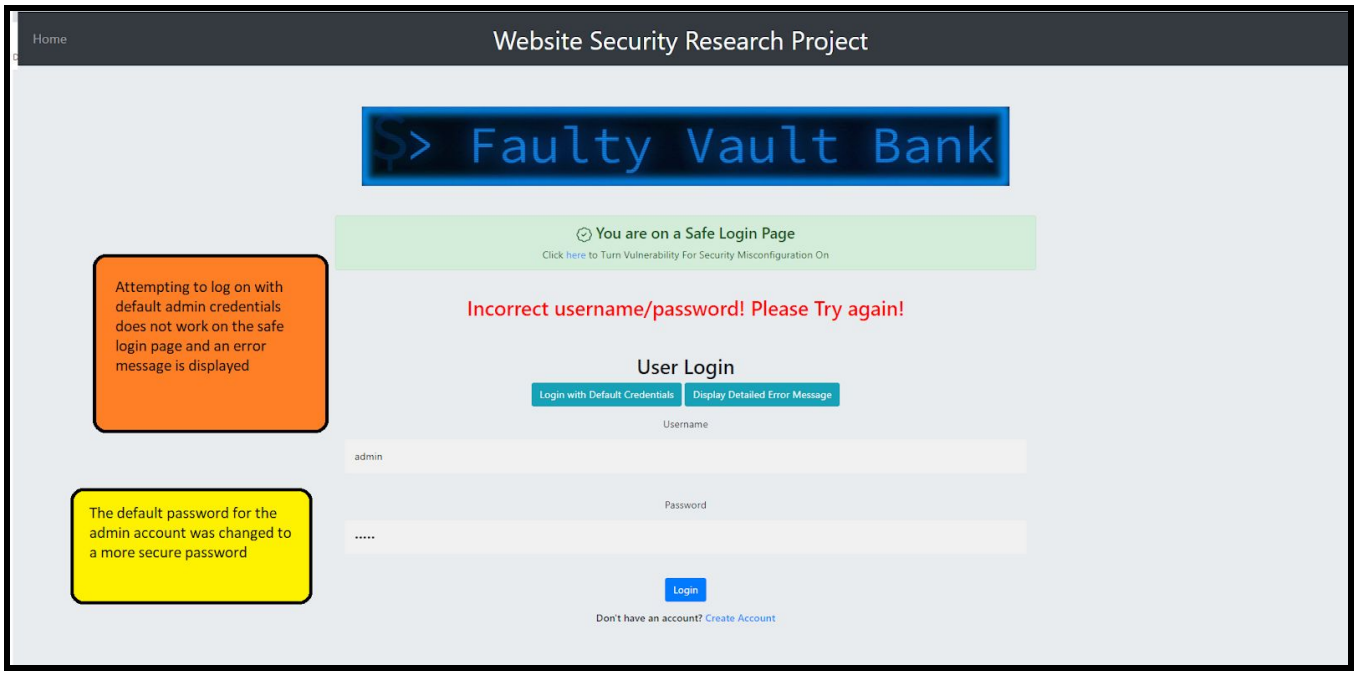




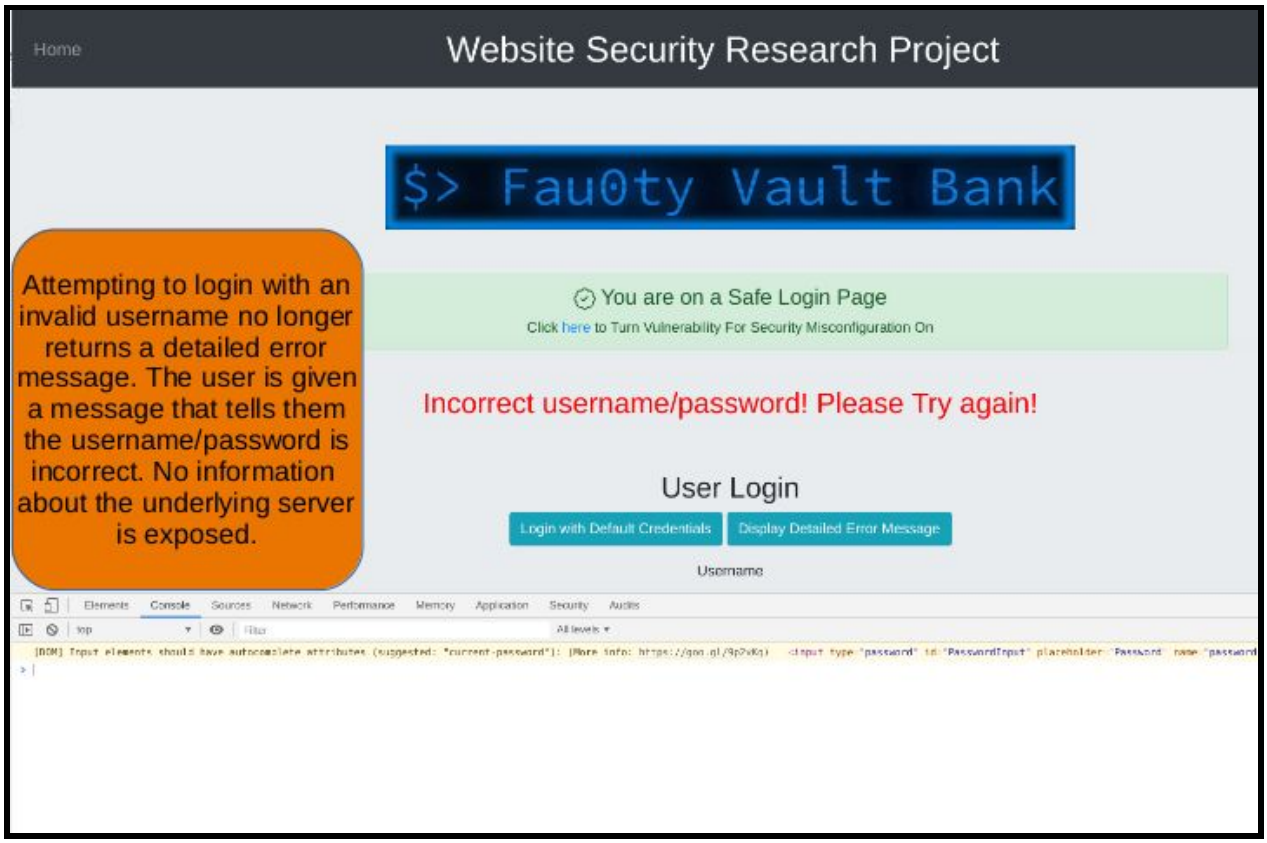
Website Hardening

Every piece of software that you install on your system requires manual security configuration, including the web server, application server and database server. To prevent unauthorized access, change every default password to something secure and not easily guessable by a would-be attacker. Furthermore, uninstall or disable any unused or unnecessary features or services from any new piece of software you install on your system. A minimal platform without any unnecessary components or features is much more secure and is less likely to have as many configuration issues.

Attempt at Logging On With Default Credentials After Website Hardening:



Attempting To Login With Invalid Username After Website Hardening:



Resources

https://owasp.org/www-project-top-ten/2017/A6_2017-Security_Misconfiguration

https://www.acunetix.com/blog/web-security-zone/security-misconfigurations/

https://datarecovery.com/rd/default-passwords/

https://www.tutorialspoint.com/security_testing/testing_security_misconfiguration.htm