HACKER COMBAT COMMUNITY

Hackercombat.com

The Ultimate Web Application Penetration Testing Checklist

Web Application Penetration Testing Checklist

HACKER COMBAT COMMUNITY

1. Contact Form Testing
2. Proxy Server(s) Testing
3. Spam Email Filter Testing
4. Network Firewall Testing
5. Security Vulnerability Testing
6. Credential Encryption Testing
7. Cookie Testing
8. Testing For Open Ports
9. Application Login Page Testing
10. Error Message Testing

Web Application Penetration Testing Checklist

11 HTTP Method(s) Testing

12 Username and Password Testing

13 File Scanning

14 SQL Injection Testing

15 XSS Testing

16 Access Permission Testing

17 User Session Testing

18 Brute Force Attack Testing

19 DoS (Denial of Service) Attack Testing

20 Directory Browsing

PENTESTING

Penetration testing is the process of testing a software by trained security experts (aka penetration testers or ethical hackers) in order to find out its security vulnerabilities.

The objective of carrying out such a test is to strengthen the security vulnerabilities which the software may contain so that they don’t get easily exploited (or taken advantage of) by the hacking community.

Web Application Penetration Testing

In the case of web application penetration testing, the software being tested is a web application stored in a remote server which clients can access over the internet.

Web applications are obviously easy targets for hackers and therefore it is imperative for the developers of these web applications to frequently carry out penetration testing to ensure their web applications stay healthy – away from various security vulnerabilities and malware attacks.

Let’s take a look at some of the elements every web application penetration testing checklist should contain, in order for the penetration testing process to be really effective.

CONTACT FORM TESTING

The most preferred entry point for spammers is often a web application’s contact form. Therefore the contact form you have in your web application should be able to identify and prevent such spam attacks.

Including CAPTCHA is one of the easiest ways of preventing contact form spamming.

HACKER COMBAT COMMUNITY

PROXY SERVER(S) TESTING

Proxy servers play a huge role in scrutinizing the traffic to your web application and pointing out any malicious activity.

Therefore ensure the proxy servers within your network are functioning accurately and efficiently. Tools like Burp Proxy and OWSAP ZAP can go a long way in helping you accomplish this task.

HACKER COMBAT COMMUNITY

SPAM EMAIL FILTER TESTING

Ensure spam email filters are functioning properly.

Verify if they are successfully filtering the incoming and outgoing traffic and blocking unsolicited emails.

In other words, ensure that email security policies are being enforced properly. Because, as we all know, spam mails are the much-preferred mode of attack for hackers.

HACKER COMBAT COMMUNITY

NETWORK FIREWALL TESTING

Make sure your firewall is preventing undesirable traffic from entering into your web application.

Also, ensure the security policies configured using the firewall are being implemented properly.

A glitch in your firewall is like sending an invitation to hackers to come and hack your web application.

SECURITY VULNERABILITY TESTING

Carry out a thorough security check on various aspects associated with your web application like servers and other such network devices and make a list of the security vulnerabilities they pose. Then find and implement ways to fix them.

CREDENTIAL ENCRYPTION TESTING

Ensure all usernames and passwords are encrypted and transferred over secure “HTTPS” connection so that these credentials are not compromised by hackers through man-in-the-middle or other such attacks.

Because just as your web application needs to be secure, so is the sensitive data being submitted by your clients.

HACKER COMBAT COMMUNITY

COOKIE TESTING

Cookies store data related to user sessions. Therefore this piece of sensitive information, if it is exposed to the hackers, can result in the security of many users who visit your website or web application being compromised.

Therefore ensure your cookie data is not exposed. Or in other words, not available in readable format or as plain text.

TESTING FOR OPEN PORTS

Open ports on the web server on which your web application has been hosted also present a good opportunity for hackers to exploit your web application's security.

Therefore carry out this security check and ensure there are no open ports on your web server.

HACKER COMBAT COMMUNITY

APPLICATION LOGIN PAGE TESTING

Ensure your web application locks itself up after a specific number of unsuccessful login attempts.

This is one of the most basic elements, which, when implemented correctly can go a long way in securing your web application from hackers.

Follow us

Visit

Hackercombat.com

HACKER COMBAT COMMUNITY

ERROR MESSAGE TESTING

Ensures all your error messages are generic and do not reveal too much about the problem.

If you do so, it’s like announcing to the hacking community, “we have a problem here, you’re welcome to exploit it!” For example: “Invalid Credentials” is fine, but the message should not be specific as “invalid username or password.”

HACKER COMBAT COMMUNITY

HTTP METHOD(S) TESTING

Also review the HTTP methods used by your web application to interact with your clients.

Ensure PUT and Delete methods are not enabled, as doing so will allow hackers to easily exploit your web application.

HACKER COMBAT COMMUNITY

USERNAME AND PASSWORD TESTING

Test all the usernames/passwords that are used on your web application.

Passwords should be fairly complex and usernames should not be easily guessable.

Separate such weak usernames and passwords and alert those users to change them.

HACKER COMBAT COMMUNITY

FILE SCANNING

Ensure all files you upload to your web application or server are scanned before they are uploaded.

HACKER COMBAT COMMUNITY

SQL INJECTION TESTING

SQL injection is one of the most popular methods employed by hackers when it comes to exploiting web applications and websites.

Therefore ensure your web application is resistant to various forms of SQL injection.

HACKER COMBAT COMMUNITY

XSS TESTING

Also ensure your web application resists cross-site scripting or XSS attacks as well.

HACKER COMBAT COMMUNITY

ACCESS PERMISSION TESTING

Check the access permissions of your users and in case your web application provides role-based access, then ensure users are getting access only to those parts of the web application to which they have the right. Nothing more or less.

HACKER COMBAT
COMMUNITY

USER SESSION TESTING

This is very important. Ensure that user sessions end upon log off. Because if they don’t, that valid session can be easily hijacked by hackers – this process is known as session hijacking – for carrying out malicious activity.

BRUTE FORCE ATTACK TESTING

Using appropriate testing tools, ensure your web application stays safe against brute force attacks.

HACKER COMBAT
COMMUNITY

DOS (DENIAL OF SERVICE) ATTACK TESTING

Also ensure your web application stays safe against DoS (Denial of Service) attacks by using appropriate testing tools.

DIRECTORY BROWSING

Ensure directory browsing is disabled on the web server which hosts your web application. Because if you don’t, you’ll be giving hackers easy access to your restricted files.

Follow us

Hackercombat.com