**Why Penetration Testing Must Be Part Of Your Software Development Lifecycle (SDLC)?**

With increasing popularity of web and mobile applications, organizations and businesses are now adopting them as useful tools to stay connected with their customers. But with increased reliance on modern tools, the probability of malicious attacks on organizational networks and applications has also increased considerably. Hackers use security flaws in applications to steal confidential customer information which can eventually lead to heavy losses on part of the organization. According to a recent research, three out of four websites are prone to attacks and most of the attacks are on applications, which cannot be protected by SSL or firewalls alone. It is here where the concept of penetration testing comes into picture.

To simply put it, penetration testing is a procedure that tests the security of a system or software application by making a deliberate attempt to compromise its security, thus testing how vulnerable it’s underlying network configurations and operating systems are. This helps to prepare for any possible malicious attacks or avoid potential breach of data at the hands of an outside party.

**How Penetration Testing can Help You?**

Following are some of the reasons why you must consider regular penetration testing for your organization, and particularly make it a part of your software development lifecycle (SDLC):

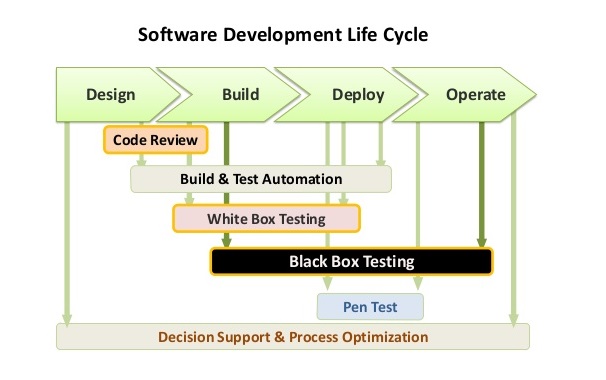
1. To outsmart automated hacking tools: Lists of vulnerabilities that can be exploited can now be easily obtained online and the use of automatic hacking tools has made it much easier for even amateur hackers to carry out successful attacks and gain access to confidential data.
2. To detect and fix vulnerabilities on time: Vulnerabilities will always exist, no matter how secure your systems are. It is therefore advisable to conduct regular pentests to detect all potential vulnerabilities and fix them before a malicious user does so. This proactive approach will help organizations avoid any threat to their confidential information that may be leaked as a result of breach of data.
3. To detect any breach of data that may have already existed: Not only will a penetration test help with the vulnerabilities, it will also help reveal any loss of private information that you may have already incurred, which, otherwise may not be revealed for a long time.
4. To help determine other weaknesses: For example, hardware and software loopholes, weaknesses in employee security awareness and system usage, etc.
5. To help practice ongoing security: It enables organizations build more confidence in terms of secure information systems. As new threats keep emerging with every passing day, regular penetration testing allows you to stay up do date and counter any conniving attacks.
6. To detect any weak practices within your organization: Internal penetration tests can help organizations find out if their policies and procedures are up to date, being understood, and followed by employees or not. It can also reveal other weaknesses such as ineffective password protection procedures, patching policies, and data encryption techniques.
7. To help measure the performance of your network and systems and take preventive/corrective actions accordingly.
8. To prepare for bad times: In case your data does get compromised in real as a result of an attack, your penetration testers and information security personnel will have gained enough experience to deal with the incident and will be better able to respond to the attack.
9. To document your security practices: Penetration testing reports provide written evidence to the management about where your organization stands in terms of its information security. Documented reports also provide you with a justification of the costs incurred on acquiring the right technology to stay more secure.
10. To conform to penetration testing requirements of PCI Data Security Standards, ISO 27001, etc.
11. To provide assurance to customers that their data is safe and protected at all times.
12. To reduce costs in the long run: Conducting regular penetration tests, especially during the software development lifecycle, reduces costs in the long run by reducing the number of vulnerabilities.

**Why make Penetration Testing a Part of SDLC?**

Making penetration testing an integral part of your software development lifecycle ensures that the end product turns out to be safe and secure for the customer. At present, the trend is to first develop a product and then conduct its security assessment at the end to check for vulnerabilities. The issues are usually fixed with a patching software, but it turns out to be much more costly than addressing the real issue.

If issues are fixed during the software development process, much of the costs can be reduced by avoiding multiple cycles of testing – patching – retesting the software at the end. Ever since the threat landscape has changed, organizations are now looking forward to providing more secure applications that are able to sustain their profitability and attractiveness for the customer. As information is getting more fragile at the hands of malicious attackers found everywhere over the internet, measures to counter such attacks also need to be improved. Malicious hackers look for all routes to enter into the network and one of these routes is the application host. Hence, the applications hosted by your organization must not be vulnerable, or else information can be easily compromised. Employing a team of penetration testers during the SDLC phase helps avoid costs that may result otherwise in case of breach of data.

Below is a generic diagram of how our penetration testing program for SDLC works.



It is important to keep in mind that penetration testing goes far beyond a set of automated tools. It is a broad approach, rather a whole process that involves the use of appropriate tools as well as human knowledge and expertise. A successful penetration tester needs to have vast experience, sharp intuitive mind, and an ability to critically analyze situations. This unique blend of abilities is necessary to allow a penetration tester to carry successful testing of vulnerabilities. This is something which automated tools alone cannot achieve.

The process of application security starts right after you begin the development process. It is, therefore, better to understand the process by dividing your SDLC into phases and taking each phase differently.

* Design – build secure design process and reviews along with formal methods like specification and modeling languages.
* Build – develop code that can be tested and used for automated review and inspection later.
* Deployment and Execution – Inspect the executed application. When conducting automated reviews and inspecting applications, use static (white box) testing and dynamic (black box) testing respectively.

Once the product is complete and ready to be launched, it is recommended to carry out a final penetration test before the product goes up for user acceptance test, so as to ensure that the test version is safe and readily accepted by consumers.