What's the Difference Between a Vulnerability Scan and a Pentest?

When tasked with securing an application there is a choice between an automated vulnerability scan and a pentest. When starting out, companies will often focus on automated testing/scanning procedures rather than manual in-depth application reviews.

Although these types of test are worthwhile exercises and should form part of a 'defence in depth' approach to security, manual application testing should be considered for critical applications.

Automated Scanning

There are a variety of tools available for automated vulnerability scanning — ranging from the simplest port scanners through network vulnerability scanners and then onto application security scanners and database security scanners. For companies, a great advantage of automated scanners is that they can be run repeatedly and provide metrics for progress in resolving vulnerabilities. The earlier an issue is addressed, the easier and cheaper it is to fix.

However automated scanners have issues. The simplest ones will give basic results, which will only provide protection from the least-skilled attackers. As scanners get more advanced they scan against known attack signatures. These signatures must constantly be kept up to date or they will not provide protection from the latest attacks.

There is also the issue of false positives, where valid application behaviour can be reported as a vulnerability. More advanced scanners have to provide a mechanism to prevent the same valid behaviour being reported as a vulnerability every time the scan is run and clogging up the results.

Automated scanners also have issues when new web technologies are introduced. There is generally a lag before the scanner is updated to handle new developments.

Manual Testing

Manual testing is, as you would imagine, a far more in-depth process and capable of identifying issues that would not be found by automated scanning. In terms of high risk issues, several types of input validation issue (e.g. code injection, SQL injection, and XSS injection) can be affected by application specific flows and procedures, which prevent them being detected by the standard signature based tests used in automated scanners. Additionally, very few automated scanners can deal correctly with file upload related issues (such as the ability to upload executable files) but a pentest can.

Automated scanners also tend to have difficulty dealing with permission based issues: horizontal privilege vulnerabilities (where one user can access another's data using techniques such as 'parameter tampering') and vertical privilege issues (e.g. where a normal user can access administrative functionality through, for instance, a 'forced browsing' vulnerability).

A web application assessment is primarily a manual pentest technique with consultants replicating techniques that external malicious parties would use to 'hack' a site. The only difference being that in most cases the security consultant is time-limited in their approach where an attacker would not be.

Testing for a Client

Prior to any pentest, consultants will engage in a scoping exercise with the client to understand the application from both a technical point of view and its businesses purpose.

The scope of the test needs to be carefully considered. A test may cover an entire application, or it may focus only on a particularly sensitive section, such as the payment process. Many applications include an administrative interface that is not accessible to users; a decision needs to be taken on whether to include this. Infrastructure testing may only cover systems that are visible to the Internet, or it may include on-site work to cover internal systems.

Whilst perimeter or network penetration tests may focus on application without credentials, the real benefit of application testing is gained through authenticated testing.

A standard application test will require two accounts at each access level within the application e.g two admin accounts, two manager accounts, two standard user accounts. This allows the tester to map all functionality within the application and then check whether a low privileged user can access admin or manager functionality. If the application contains 10+ access levels it is advisable to consolidate these to just three, therefore providing access to the highest privileged, the lowest and one in between.

The tester will access the web site in a similar manner to a user, but making use of an intercepting proxy tool which allows full control of HTTP messages.

The first stage of the pentest assessment is to discover all the pages in the application and gain familiarity with its workings. After this is done, a wide variety of potential vulnerabilities are tested for.

While conducting the thorough manual test, the tester will usually enable the Burp Suite active scanner. This automatically attempts the more repetitive tests; e.g. checking for simple cross-site scripting and poor configuration. The scanner is blocked from accessing certain pages; e.g. the login page, to avoid test accounts becoming locked-out.

Directory and file brute force tools will be used to look for pages that exist on the web server but are not linked to from other pages. This includes checking for backup versions of files discovered during testing.

Where vulnerabilities are identified, a working exploit will usually be developed, unless to do so would go against client instructions or place the tested application at undue risk.

Generally, it is key that access is provided in an application assessment so that the secure parts of the application can be tested, rather than spending time working out how to defeat the access control mechanism.

Due to the complexity of the test, a tester will require prerequisites that are often not required for an automated scan. These will include information relating to the application such as:

- Providing external network IP address ranges and URLs to be tested
- Provision of login credentials for suitable testing accounts. A minimum of two per application role
- Provision of suitable test data, allowing the test accounts to access all parts of the application
- Where defensive tools are in use it may be required to add the testers source IP address to the exclusion list for the relevant phase of testing

The Best of Both Worlds

Manual and automated testing can be deployed together to provide ‘defence in depth’. Integrating automated testing into the development and deployment life-cycle provides confidence that simple vulnerabilities have not been introduced (or re-introduced by a code regression) and that a configuration error has not opened a simple attack vector.
Then manual testing can be effectively used to identify the subtle flaws that automated scanners are bad at finding, without having to report on simple configuration issues.

Secarma is able to advise and educate on the effective use of vulnerability scanners and perform pentesting, from simple quick scans to targeted application and database reviews.

For more information about our pentest services, vulnerability scanning, or other cybersecurity services, contact a member of our dedicated team.