13.3 The Attack Phase

After passive scanning, port scanning, enumerating, and gathering information about the target site, the attacker will be ready to attack the target system. This is the part where he or she applies the knowledge gained in the scanning phases.

13.3.1 Physical Access Attacks

If an attacker can physically sit in front of any machine connected to your network, there are a number of ways he can use that to gain access to your entire network. Their first step is simply to be able to log on to that machine. They do not need to be logged onto that network just yet, just that machine.

13.3.1.1 Bypassing the Password

One very exciting way to break into Windows computers is to simply bypass the password all together. You don’t try finding out what the password is; you just skip it. It requires about 5 minutes at the workstation with a Linux live CD.

13.3.2 Remote Access Attacks

Obviously, physical access to a workstation on the target network is not always possible. Although remote attacks are far less likely to succeed, they still have the potential to succeed. A number of possible remote attack methods exist, but this section focuses on a couple of the most common: SQL injection and cross-site scripting.

13.3.2.1 SQL Injection

SQL injection is a popular attack against web applications. A login screen requires a username and password, which must be checked against a database to see whether they are valid. All databases speak Structured Query Language (SQL). If the programmer who created the login is not careful, it might be susceptible to SQL injection. Here is how that attack works. SQL looks a lot like English. For example, to check a username and password an intruder might want to query the database and see whether any entry in the users table matches the username and password that was entered. If there is, then a match exists.

13.3.2.2 Cross-Site Scripting (XSS)

An attacker injects a client-side script into web pages viewed by other users. The term cross-site scripting originally referred to the act of loading the attacked, third-party web application from an unrelated attack site, in a manner that executes a fragment of JavaScript prepared by the attacker in the security context of the targeted domain.

Essentially, an attacker enters scripts into an area that other users interact with. When they visit that part of the site, the attacker’s script runs, rather than the intended website functionality. This can include redirecting users.