**Injection Flaws**

**Description 1**

Injection flaws allow attackers to relay malicious code through a web application to another system. These attacks include calls to the operating system via system calls, the use of external programs via shell commands, as well as calls to backend databases via SQL (i.e., SQL injection). Whole scripts written in perl, python, and other languages can be injected into poorly designed web applications and executed. Any time a web application uses an interpreter of any type there is a danger of an injection attack. The **consequences** can also run the entire range of severity, **from trivial to complete system compromise** or destruction. In any case, the use of external calls is quite widespread, so the likelihood of a web application having a command injection flaw should be considered high. [1]

**Description 2**

Injection occurs when user-supplied data is sent to an interpreter as part of a command or query. Attackers trick the interpreter into executing unintended commands via supplying specially crafted data. **Injection flaws allow attackers to create, read, update, or delete any arbitrary data available to the application.** In the worst case scenario, these flaws allow an attacker to **completely compromise the application** and the underlying systems, even bypassing deeply nested firewalled environments. [2]

**Description 3**

Injection flaws are a class of security vulnerability that allows a user to "break out" of the web application context. If your web application takes user input and inserts that user input into a back-end database, shell command, or operating system call, your application may be susceptible to an injection flaw. A user exploits this by breaking out of the intended "context" and appends additional and often unintended functionality. **By allowing injection flaws in your application you are allowing an attacker to create, read, update, or delete any arbitrary data available to the application**. [3]

**Examples:** LDAP injection, XML Injection, XPath Injection, OS command injection, and HTML injection.[3]

**Reference**

**[1]** [**http://www.upenn.edu/computing/security/swat/SWAT\_Top\_Ten\_A6.php**](http://www.upenn.edu/computing/security/swat/SWAT_Top_Ten_A6.php)

**[2]** [**https://www.owasp.org/index.php/Top\_10\_2007-Injection\_Flaws**](https://www.owasp.org/index.php/Top_10_2007-Injection_Flaws)

**[3] http://bretthard.in/post/injection-flaws**