**Command insertion**

**Other Names: command injection**

**Description 1**

Command injection is an attack in which the goal is execution of arbitrary commands on the host operating system via a vulnerable application. Command injection attacks are possible when an application passes unsafe user supplied data (forms, cookies, HTTP headers etc.) to a system shell. In this attack, the attacker-supplied operating system commands are usually executed with the privileges of the vulnerable application. Command injection attacks are possible largely due to insufficient input validation.

This attack differs from Code Injection, in that code injection allows the attacker to add his own code that is then executed by the application. In Code Injection, the attacker extends the default functionality of the application without the necessity of executing system commands. [1]

**Description 2**

command injection occurswhen input data is interpreted as an operating system command. **The impact of this attack is that** an **attacker can execute arbitrary commands on the system**. [3]

**Example 1**

The following code is a wrapper around the UNIX command *cat* which prints the contents of a file to standard output. It is also injectable:

**#include <stdio.h>**

**#include <unistd.h>**

**int main(int argc, char \*\*argv) {**

**char cat[] = "cat ";**

**char \*command;**

**size\_t commandLength;**

**commandLength = strlen(cat) + strlen(argv[1]) + 1;**

**command = (char \*) malloc(commandLength);**

**strncpy(command, cat, commandLength);**

**strncat(command, argv[1], (commandLength - strlen(cat)) );**

**system(command);**

**return (0);**

**}**

Used normally, the output is simply the contents of the file requested:

**$ ./catWrapper Story.txt**

**When last we left our heroes...**

However, if we add a semicolon and another command to the end of this line, the command is executed by catWrapper with no complaint:

$ ./catWrapper "Story.txt; ls"

When last we left our heroes...

Story.txt doubFree.c nullpointer.c

unstosig.c www\* a.out\*

format.c strlen.c useFree\*

catWrapper\* misnull.c strlength.c useFree.c

commandinjection.c nodefault.c trunc.c writeWhatWhere.c

If catWrapper had been set to have a higher privilege level than the standard user, arbitrary commands could be executed with that higher privilege.

**Example 2**

please watch video in [2]

**Reference**

[1] <https://www.owasp.org/index.php/Command_Injection>

[2] <https://www.youtube.com/watch?v=aFUZLA0f2KE>

[3] <https://www.netsparker.com/web-vulnerability-scanner/vulnerability-security-checks-index/command-injection/>