| 1 void CWE121\_Stack\_Based\_Buffer\_Overflow\_\_src\_char\_declare\_cpy\_01\_bad() 2 { 3 char \* data; 4 char dataBuffer[100]; 5 data = dataBuffer; 6 /\* FLAW: Initialize data as a large buffer that is larger than the small buffer used in the sink \*/ 7 memset(data, 'A', 100-1); /\* fill with 'A's \*/ 8 data[100-1] = '\0'; /\* null terminate \*/ 9 { 10 char dest[50] = ""; 11 /\* POTENTIAL FLAW: Possible buffer overflow if data is larger than dest \*/ 12 strcpy(dest, data); 13 printLine(data); 14 } 15 } |
| --- |

From the above code, we can see a stack buffer overflow caused by the statement “strcpy(dest, data)“ at line 12. Specifically, the buffer “dest” has a size of 50 char elements. However, the statement tries to copy the string carried in the buffer named “data” to “dest”. As initialized by the statements of “memset(data, ‘A’, 100-1)” at line 7 and “data[100-1] = ‘\0’” at line 8, the string in “data” will have a size of 99 char elements (note: string in C program is basically a sequence of bytes that ends at a NULL byte). Thus, the strcpy will overflow “dest” and thus, a stack overflow happens.