EECE 412, Fall 2009

Quiz #4

This quiz consists of 6 pages. Please check that you have a complete copy. You may use both sides of each sheet if needed.

Your Family name: __________________

Your Given name: __________________

Your student ID: __________________

#	Points	Out of
1		6
2		3
3		12
TOTAL		21

Name of your left neighbor: ________________

Name of your right neighbor: ________________

ATTENTION: When necessary, make reasonable assumptions and state them clearly in your solutions.

1. Consider the following example code in C.

```
void foo (int a, char* s) {
        char buffer[10];
        strcpy(buffer, s);
}

void main( int argc, char* argv[ ] ) {
        foo(1, argv[1]);
}
```

If everything goes fine when function foo is called, then the memory layout during execution of foo is shown in the following figure, where the thick black arrow shows how the program counter would change on the return from foo to main.

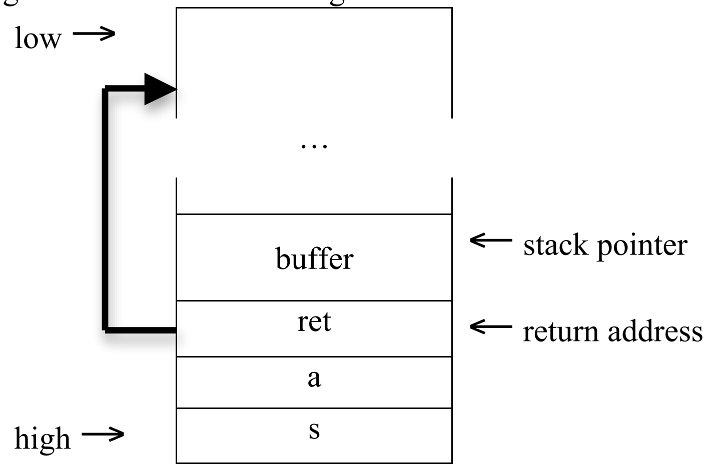


Figure 1.

Now, suppose a buffer overflow has occurred in foo, which resulted in the following memory layout. Such an overflow would generally crash the program.

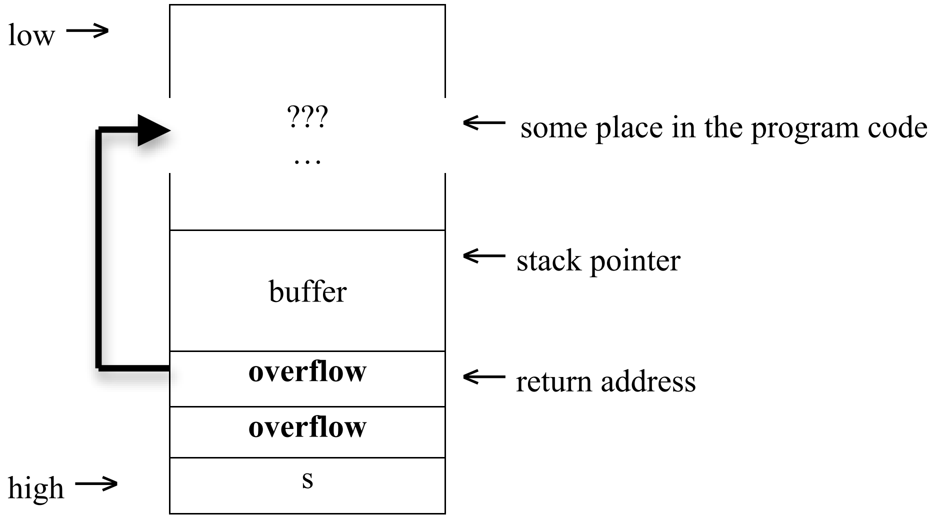


Figure 2.

Now, imaging that at another time, when the program was executed again, another buffer overflow (this time more malicious) occurred, which resulted in the memory layout shown in the following Figure 3.

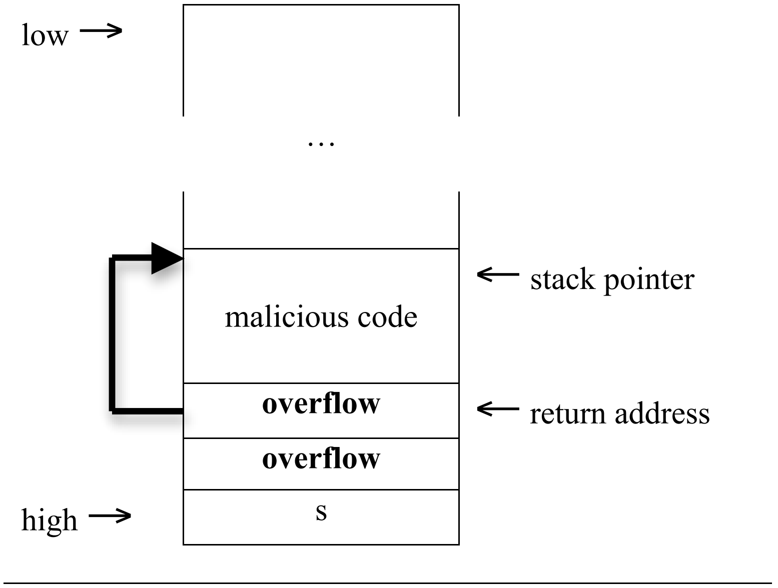


Figure 3.

Discuss three different options for mitigating or avoiding the above two types of buffer overflow attacks. Explain which option will work and will not work for which of the above buffer overflow attacks.

1. No execute (NX) bit would result in the program crash for the attack show in Fig. 3 but not the one in Fig. 2.

2. Canary would crash the program in the case of both attacks.

3. The use of "safe" language, e.g., Java or C#, would prevent the attacker from overflowing the buffer and overriding the return address, which could result in an application-level exception.

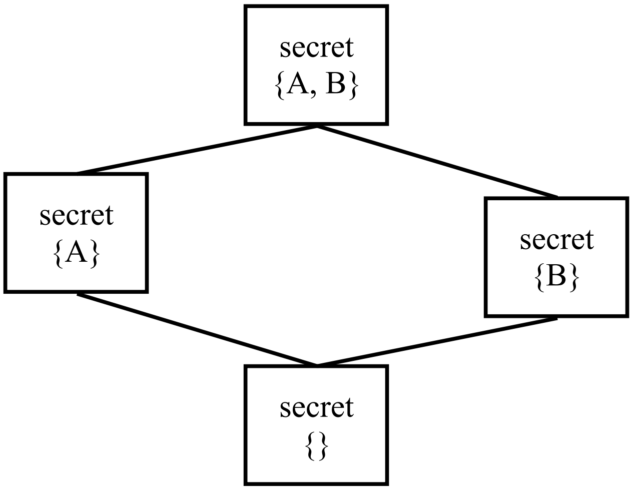
2. Explain the difference between how SQL injection attacks and cross-site scripting attacks work.

An SQL injection attack is an exploit that injects malicious user inputs into dynamically constructed SQL statements; while cross-site scripting attacks injects malicious user inputs into dynamically generated HTTP responses.

SQL injection attacks can target both server-side servers and client-side victims while cross-site scripting attacks can only attack client-side victims that are using a web browser.

3. Access control. For the following two policies, determine whether they are equivalent (i.e., users have same permissions in both policies). Explain your answer.

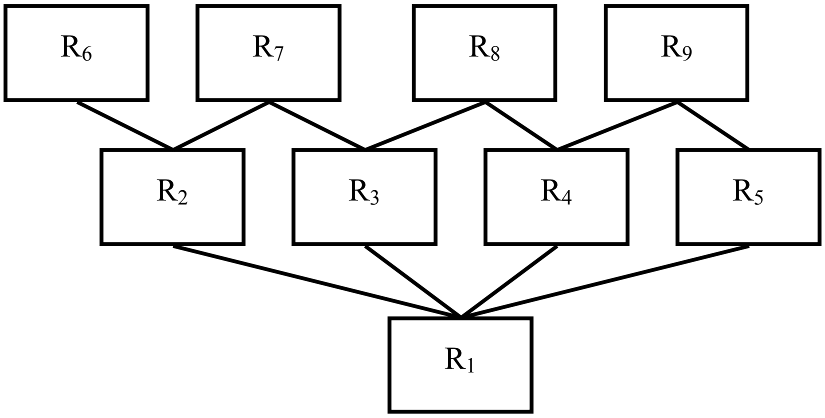
a. BLP policy.



Label	**Object Classification**	**Subject Clearance**
secret {A, B}	O_3	U_4
secret {B}	O_4	U_3
secret {A}	O_2	U_2
secret { }	O_1	U_1

b. Hierarchical RBAC policy

Role hierarchy (RH):



Permission-to-role assignment (PA):

permssn role	O_1	O_2	O_3	O_4
R_1	read		append	
R_2				append
R_3				read
R_4		read		
R_5		append		
R_6	append			
R_7				
R_8			append	
R_9				

User-to-role assignment (UA):

user role	U_1	U_2	U_3	U_4
R_1				
R_2				
R_3				
R_4				
R_5				
R_6	X			
R_7				X
R_8			X	
R_9		X		

The above two policies are equivalent: ____Yes, ____No

Because …

Access matrices for the above policies are almost the same. The difference is underlined. U_1 can append to O_2 in the BLP policy but not in the RBAC one.

object user	O_1	O_2	O_3	O_4
U_1	read, append	append	append	append
U_2	read	read, append	append	---
U_3	read	read	read, append	read
U_4	read	---	append	read, append