Sample Solution and Comments on Quiz #4

1. (8 points) Which of the following are effective (but not necessarily most feasible) ways of reducing the number of security vulnerabilities in software systems?

(Correct items are in bold)

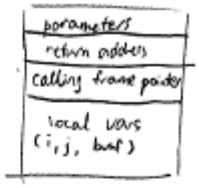
1. **Build security into development process**
2. Tighten the perimeter controls, e.g., firewalls
3. **Improve software development processes in order to reduce the number of all (i.e., not only security ones) defects by order of magnitude**
4. **Practice principles of designing secure systems**
5. Purchase insurance against security vulnerabilities
6. Make sure outsiders do not know the details of the software design and implementation
7. Outsource software development to a an offshore software vendor
8. **Employ static analysis and other tools that can flag potential vulnerabilities so that the developers can review and correct the suspects**

2. (15 points) Explain how buffer overflow attacks work and what can be done to avoid, prevent, and detect them.

Suppose we have the following code:

```
void foo (char* str) {
    int i, j;
    char buf[200];
    strcpy(buf, str);    // copy str into buf
    printf("Hello, %s", buf);
}
```

stack structure when foo() is called.



Now suppose we call the function foo by passing in a string longer than 200 chars. Since the string is longer than the size of the buffer 'buf' (200 chars), the content on the buffer overflows and overwrites other data (see diagram to the left).

If the provided string is designed such that the return address is overwritten to point to some other location (such as malicious code included in the buffer content), the buffer overflow attack succeeds, and the system may enter an unauthorized state.

The main way to avoid and detect such attacks is to sanitize and validate inputs. Do not assume the user would use the function properly.

Also, avoid using functions (such as strcpy()) that are susceptible to buffer overflow attacks. There are safer alternatives (such as strcpyn()) that prevents overflows from occurring. In the case of parsing functions such as scanf(), it is better to implement your own safe parsing function.

Contributed by Henry Ng.

3. (6 points) For the following examples, identify which principle of designing secure systems is followed or not. Explain your answer:

1. On Berkley-based versions of the UNIX operating systems, user are not allowed to change from their accounts to the root account, which has administrative privileges, unless two conditions are met:
   - The user should know the root password.
   - The user account is in the wheel group.

   Meeting either condition is not sufficient for acquire root access. Meeting both conditions is required.

Satisfies principles of separation of duty.
Access / privilege is not granted ~~by~~ based on single condition; instead multiple conditions are required to ~~more~~ fine-grained security.
In this case, knowing both root password and the account is in wheel group. is more secure than just having to know the root password in order to login as root.

Contributed by Alice Ho Yu Au-Yeung.

2. Administrative accounts in the UNIX and Windows operating system allow users to perform any modifications to the OS and its resources. Thus even those applications that perform only system backup can still delete or modify any files if they are run under admin accounts.

Violates principle of least privilege.
Doing system backup doesn't need admin privilege, thus it is not using the least set of privilege necessary to do the job.

Contributed by Alice Ho Yu Au-Yeung.

4. **(10 points) List the main failure causes for enterprise security initiatives in today organizations.**

Sample answer:

1. **Lack of demonstrated ROI**
2. **Poor definition of success**
3. **No real business alignment**
4. **No long-term strategy to decrease the level of overall security risk and exposure**
5. **No framework within which to design and deploy solutions for new problems**
6. **Technically led, IT-based security projects**
7. **Low prioritization of security as compared to business initiatives**
8. **Lack of appreciation for the importance of security in today's enterprise**
9. **Immaturity of technology solutions**

Comments:

1. Most people did question 1 and question 2 well.
2. Lots of people misunderstood question 4 and answered something else.