Sample Midterm Exam

1. The following code fragment exists in a program that is setuid to *root*. Its function is to read a series of lines from the standard input and append them to the file named in the variable `file`. However, for security reasons, it must not append the lines if `file` is a symbolic link to another file. The following sequence is intended to implement this functionality. Please determine if it does correctly (*i.e.*., does this code pose a security risk)? Justify your answer, of course. (Note: the label `error` is defined elsewhere to handle error conditions.)

```
/* get the attributes of the file associated with
   the name; do not follow any symbolic links      */
if (lstat(file, &buf) < 0)
      goto error;
if ((buf.st_mode&S_IFLNK) == S_IFLNK){
      fprintf(stderr, "%s is a symbolic link\n", file);
      goto error;
}
/* open the file, write to it, and close it */
if ((fp = fopen(file, "a")) == NULL)
      goto error;
while(fgets(buf, BUFSIZ, stdin) != NULL)
      fputs(buf, fp);
(void) fclose(fp);
```

2. Why is a precise statement of security requirements critical to the determination of whether a given system is secure?
3. Please describe how the vulnerabilities models are used during the Flaw Hypothesis Methodology. Be explicit: which phase of the methodology uses them, and how?
4. Classify each of the following as anomaly, misuse, or specification-based intrusion detection. Please justify your answer.
   a. The intrusion detection system detects that Matt, a secretary hired to process proposals and budgets, is running a program that connects to random ports on other systems connected to the Internet.
   b. The *su* program is writing to the file /home/bishop/gotcha. The design of *su* does not say that it may write to that file.
   c. On the MTS (Michigan Terminal System), students are repeatedly passing arguments to system calls. These arguments are pointers into parameter blocks.
5. Does the UNIX operating system enforce the principle of complete mediation for ordinary users (*i.e.*, excluding *root*)? If not, what needs to be changed to enforce that principle?
6. Into which category or categories of the Program Analysis classification do the following fall? Please justify your answer.
   a. Buffer overflow causing a return into the stack?
   b. Allowing an ordinary user to alter the password file?
   c. Simultaneous writes to a shared database?
   d. Reading a UNIX file by directly accessing the raw device and reading first the superblock, then the file's inode, and finally the file's data blocks?
7. Which of the following demonstrate violations of the principle of least privilege? Please justify your answer.
   a. The UNIX *root* account?
   b. A user whose function is to maintain and install system software. This user has access to the source files and directories, and can copy executables into system directories for other users. This user has no other special privileges.