1. Why do I need to generate a new session ID for each successful login to web application?

2. How can an attacker circumvent address space randomization?

3. Find all potential vulnerabilities in this C function:

   ```c
   void store_socket_data_to_file(int fileno)
   {
     int i, len, fd, f;
     char *buf;

     fileno = fd;
     if (read(fd, &len, 8) < 0) return;
     buf = malloc(len);
     if (read(fd, buf, len) < 0) return;

     if (access(buf, "w") < 0) return;
     if (f = open(buf, "w") < 0) return;
     while (i = read(fd, buf, 1024) > 0 ) {
       write(f, buf, 1024);
     }
     close(f);
   }
   ```

4. Find all potential vulnerabilities in this PHP snippet:

   ```php
   <?php
   if (session_id() && isset($_REQUEST['newaddress'])) {
     $sql = "UPDATE users SET email='" . $_REQUEST['newaddress'] . "] . " WHERE username='" . currentUser() . "] . "'";
     mysql_query($sql) || die($sql);
     echo "Contact address has been changed to " . "] . "$_REQUEST['newaddress'];
   }
   ?>
   ```

5. Find all potential vulnerabilities in this Python function:

   ```python
def savefile(filepattern, mydata):
    filename = os.popen("ls -t " + filepattern + "] | head -n 1").read()
    if os.path.islink(path):
      raise Exception(’Not following a symlink!’)
    file_contents = open(filename).read()
    open(filename, "w").write(mydata)
   ```