CSE 345/545: Foundations to Computer Security

Homework Assignment II (**100** Points)

Due: `2359hrs 12 October 2017`

Plagiarism policies will be strictly enforced.

A large number of security vulnerabilities in a software system are the results of design and programming errors, some of which are covered in this lab assignment.
Required files if any are available: `github.com/mfrw/fcs`

Part I [`10 points`]

IIIT-Delhi has a campus-wide 802.1X wireless network.

1. Explain the authentication technique used by IIIT-D, if any.
2. Is the IIIT-Ds wireless service susceptible to packet sniffing? Provide justification for your answer.
3. Describe what you would consider to be a typical Internet usage session. What could a potential attacker learn by sniffing this traffic?

Part II [`15 points`]

In this problem, you will use JavaScript to exploit Cross-Site Scripting (XSS) vulnerabilities. Open the part2.html using a web browser, and answer the following questions:

1. You can instruct web browsers to run JavaScript by typing JavaScript URIs in the address bar. For example, try to type the following URI into your browsers address bar.

   ```
   javascript:alert('hello world');
   ```

   Describe what happens. Explain how the browser will interpret and process the JavaScript URI.

2. Execute the JavaScript snippet present int the file `tc.js`, in the browser:

   Describe what happens. Check whether the source code of this webpage has been changed. Explain how can an attacker change the contents of webpage without modifying its source code?

3. Web sites containing user-created content (UCC) are at great risk of having Cross-Site Scripting (XSS) vulnerabilities. The given web page contains a Discussion Board. Assume that the Discussion Board is dynamically updated by users. Click the link `check out this cool link` in the Discussion Board and describe what happens. Modify the given source code to add a link in the Discussion Board that replaces the logo image with
   `https://www.wpblog.com/wp-content/uploads/2017/08/wordpress-site-is-hacked.jpg`

4. Using the Submit button, you can write anything on the Discussion Board. Add a link on the Discussion Board that replaces the logo image in your web browser with
   `https://www.wpblog.com/wp-content/uploads/2017/08/wordpress-site-is-hacked.jpg`.
   What did you add on the Discussion Board?

5. Click the link `Go to IIIT-D homepage` to the IIIT-Ds homepage. Click the link `Hello, see this page` in the Discussion Board. Then, click the link `Go to IIIT-D homepage` again. Describe what happens. Check whether the source code of this webpage has been changed. How can an attacker trick other users to visit a fake website without modifying the source code?

6. There is a link `Check your email` in the webpage obtained by using the above HTML code. Change the HTML code to add a link in the Discussion Board to make the link `Check your email` to lead to `https://www.reddit.com/r/hacking/`. How can an attacker steal other users email login information using this vulnerability?

7. Besides changing the content of a webpage, an attacker can also inject malicious code and execute it in your web browser using XSS vulnerabilities. This is very dangerous because the malicious code will have the same privilege as your web browser. In this part, you will simulate Session Hijacking attack which is one of the most common attack patterns using XSS vulnerabilities.

   Set your Session ID as 12345 by typing the following JavaScript URI into your browsers address bar.

   ```
   javascript:void(document.cookie='Session ID=12345');
   ```

   Check whether the Session ID is set correctly by typing the following JavaScript URI into your browsers address bar.

   javascript:alert(document.cookie);

   Add the link given in `db_link.htm` to the Discussion Board using the `Submit` button.

   Click the new link and describe what happens. How can an attacker steal your Session ID using this vulnerability?

8. How can you prevent the XSS vulnerabilities? Modify the code to prevent it.

9. Create a Self signed Certificate using OpenSSL and host the code using a any webserver of your choice (SSL). You can get creative and try without using a webserver.
   Think Shakespeare !

10. Would Using `https` do any good for the security of the site in the context discussed sofar.

Part III [25 points]

Use any web security application testing tool that serves as a proxy for intercepting browser web requests and web server replies.

1. With the tool turned on, go to your favorite Web sites and interact with them, such as logging and downloading files. Explain the kinds of information that can be captured.

2. List three vulnerabilities/possible attacks that can occur by exploiting the data obtained.

3. If you were a developer, what could you do to prevent misuse of this data? Provide justification on why your approach would prevent the misuse of the data.

One such tool you can use is WebScarab available at :
`http://sourceforge.net/projects/owasp/files/WebScarab/`.
Some tutorial links are here:
`https://www.owasp.org/index.php/WebScarab_Getting_Started`
`http://yehg.net/lab/pr0js/training/webscarab.php`

Part IV [45 points]

Given the speed constraints of crypto, most of the implementations are either assembly or c. Write `C` programs, that implement: Only C programs are acceptable, no python, c++, java or any other language.

1. DES algorithm. Name the source code file as `des.c` and the binary as `des`

   ```
   # For encryption
   mfrw@kp $: ./des -e -i plain.txt -o enc.txt -k key.txt
   # For decryption
   mfrw@kp $: ./des -d -i enc.txt -o plain.txt -k key.txt
   # Flags
   -e => encrypt
   -d => decrypt
   -i => input file
   -o => output file
   -k => key file
   ```

   Ensure that the command line switches are implemented correctly as the submission would be checked via a **dumb script**.

2. Use the `OpenSSL` C library to create a naive chat application, which sends and receives encrypted data. Be creative in design so no instructions here.
   Verify by using `Wireshark`.

3. The security domain is dynamic and one has to adapt and improvise with tight deadlines to meet. Most of the times the technology is entirely new and a problem creeps up. Isolating the problem and finding a viable solution by scraping the documentation and best practice guidelines is needed for quick response. In this spirit modify the source file `fileserver.go`

to serve TLS (transport layer security). The current program is able to act as a simple web server for serving http. Modify the source code so that it uses TLS (https). Use the documentation and generate appropriate files with minimum code intrusion.

4. Download the `ToR` source code. Don't download the ToR Browser Bundle, build only ToR from source. Run the binary on the command line, and configure your browser to use it. The configuration should be such that your computer should accept connections from any computer on IIITD-LAN. Bonus Points for using bridges and even more bonus points for using any transport method like obs4 etc.

5. Create your own key file, public/private keys as deemed appropriate. Encrypt the `large_file.txt` using OpenSSL tool:

   - With a stream Ciphers RC4, ChaCha algorithm
   - With symmetric Ciphers DES, 3DES, AES
   - With RSA

   Time all the encryption algorithms. Compare the encryption times for the three classes of encryption algorithms. Can you reason sanely why some are blisteringly fast and others terribly slow ?

Part V [`5 points`]

This is the reading section which should be informative. All the below given reads are very light in nature. If in anycase you would want more involved ones, approach the TA's or drop a mail.

- Read the article by Aleph One, `Smashing the Stack for fun and profit`. Write a 5 line description of what you understood and also a proof of concept binary and shellcode.
- Read this interview by Voice & Data of our former faculty about crytpo, the nature of security breaches and its impact. Write a short 2-5 line jist of the points you felt interesting. `https://goo.gl/MBiVWg`

Submission

Submit a zip file containing

- A pdf file with brief steps and screen shots completely documenting the process, as you went. Note that while taking screen-shots your username is visible in the screen shot.
- The Source code of programs you write.

Submission guidelines (points will be deducted if not followed): Please post it onto Backpack by the deadline. Do not send it by email! No email submissions will be entertained.