CSE 345/545: Foundations to Computer Security

Assignment III (100 Points)

Due: 2359hrs 10 November 2018

Plagiarism policies will be strictly enforced

Part I [20 points]

The PGP (Pretty Good Privacy) is the defacto standard for signing, encrypting, and decrypting texts, e-mails, files, directories, and whole disk partitions and to increase the security of e-mail communications. Phil Zimmermann developed PGP in 1991. We usually use the GNU version of the PGP called GPG (GNU Privacy Guard). Go through RFC 4880 for more information.

1) [10 points] Create your public/private key pair (4096 bit).
2) [5 points] Encrypt a file with the public key. Make sure you sign it too with your own private key. Use the ASCII armored option when you encrypt.
3) [5 points] Can you decrypt the contents of the file you just created? Why or Why not? Give Reasons.

Part II [30 points]

Cryptographic hash functions have a variety of uses. One of the typical use cases is the checksum of a file. Integrity of a file is ensured if the checksum is same.

1) [10 points] Calculate the md5, sha1, sha2 (all 4 variants), sha3 of a sample text of your choosing. Time them on some large files of your choice to get a flavour for their relative speed.
2) [20 points] Priyabrata wants to read some of the stories present on the website (http://www.textfiles.com/stories/). He's mainly interested in the stories with filenames- **100west.txt**, **13chil.txt**, **14.lws**, **16.lws** and **17.lws**. Unfortunately, he's unable to access this website as the site was recently blocked by his ISP. On further searching, he comes across another source (https://drive.google.com/open?id=16bStDDMXlFAx59_kt0cFxwnoVurmj86D) which contains all the stories he wants to read. Still, he is a bit suspicious that all or some of the

stories in the new link might have been tampered with by a third party but he cannot identify how many and which ones.

a) Your task is to help Priyab identify which of the five stories were modified by the third party and how many of them were modified. Also explain your methodology for identifying the tampered stories.
b) Is there any possibility that the file was modified but still couldn't be detected i.e. the checksum of both is still same even after modification? Why / Why not?
c) Which security property of **cryptographic hash functions** might get violated in b) and which hash functions in Part II-1 may face this issue (with respect to computational resources available at present)?

Note – Question 2c) needs to be answered with respect to the Security properties of Cryptographic hash functions which differentiates them from normal hash functions

Part III [30 points]

Password are a very important token for security. All kinds of replacements have been attempted to replace the password like biometric, otp based etc, but none has the simplicity, ease of implementation and usability of a password. Choosing a secure and weired/non-guessable password is very important.

1) [10 points] Write C (only C) program that implements a password functionality

```
# Registering a user
arun@kp $: ./passwd -r
Enter Username: arun
Enter Password:                 # note that nothing should get printed - like unix passwords
# Authentication
arun@kp $: ./passwd -a
Enter Username: abcd
[error] User not registered
# Ok second Try
arun@kp $: ./passwd -a
```

Enter Username: arun
Enter Password: ******* # bonus for printing stars
[success] Authenticated

Check Part 3 directory of Resources for standard files.

2) [5 points] Write a separate C program and assuming you don't know the password of some registered user, launch a brute force attack. Follow the coding conventions of passwd.
3) [5 points] Open up your /etc/passwd file. If you are on Windows, use the lab (linux) machines. Study its permissions. Where is the password for your Username? Can you explain the rationale?
4) [5 points] Modify the passwd program with minimal intrusion so that bruteforcing becomes harder. Name it passwd2.
5) [5 points] Download John The Ripper, compile and install it. Grab an unshadowed version of you /etc/password file and crack the password for your username.

Part IV [20 points]

There are a lot of tools in the domain of security. here we try to scratch the surface of some.

1) [5 points] Nix based systems have an embedded industry standard firewall called as packet-filter. It uses the net-filter framework inside the Linux Kernel and BPF (Berkely Packet Filter) subsystem in BSD based systems. The linux variant of packet-filer is iptables. A replacement of iptables called nftables is also under heavy development. Use iptables to:

   • Disable Echo-Reply (pong) to your machine. Your machine should not reply to ping from any other machine (act dead/not available). You should be able to ping other devices though.
   • Host a webpage on your machine. Use iptables to only allow your own mobile-phone to access the web-page and block all others.

2) [5 points] nmap (Network Mapper) is an important tool for discovering devices on your network. It has excellent documentation. Use nmap to:

• Find all computers in the B519 lab subnet which provide access to ssh.

• Do an OS finger-printing of the hostel wifi. Provide stats of type of OS found (eg 70%Windows, 20%Linux ...) [please prove these stats wrong]

3) [10 points]Setup a VPN server on a random lab machine. Connect your mobile-phone to it. Host a web-page on your computer. Access the webpage through the VPN. Verify the IP address that shows up in the log of the webserver. This may be helpful: https://github.com/Nyr/openvpn-install

Part V [20 points]

In this section the task is to analyze network packet traces (commonly called a "pcap"). Extract relevant details using the Wireshark network analyzer.

1) [10 points] Examine log.pcap file (check Resources) and concisely answer the question below. Each response should require utmost 2-3 sentences.
   a) Multiple devices are connected to the local network. What are their MAC and IP addresses?
   b) What type of network does this appear (e.g., a large corporation, an ISP backbone etc)? Support your claim with some evidence.
   c) One of the clients connects to an FTP server during the trace. What is the DNS hostname of the server? Do you think FTP is a safe protocol to transfer confidential data? Name a replacement that could be used instead of FTP which is not clear-text.
   d) One of the clients make HTTPS connections to sites other than Facebook. What is domain name of the site the client is connecting to? Is there any way the HTTPS server can protect against leak of information which was discovered in the previous question? During TLS handshake, the client provides a list of supported cipher suites, are any of these worrisome from a security point of view?
   e) One of the clients makes some requests to Facebook. Even though logins are processed over HTTPS, what is insecure about the way the browser is authenticated to fb? How would an attacker leverage the above info to impersonate the user?

2) [10 points] Write a python program that analyzes a pcap file in order to detect possible SYN scan. Use dpkt to make your life easier. The output of the program should be the malicious IP's doing the scans. Your Task is to find at least one malicious IP.

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.