Homework #4

Due Time: 2022/05/01 (Sun.) 21:59
Contact TAs: `vegetable@csie.ntu.edu.tw`

Instructions and Announcements

- **NO LATE SUBMISSION OR PLAGIARISM IS ALLOWED.**
- Discussions with others are encouraged. However, you should write down your solutions **in your own words**. In addition, for **each and every** problem you have to specify the references (the URL of the web page you consulted or the people you discussed with) on the first page of your solution to that problem.
- Some problems below may not have standard solutions. We will give you the points if your answer is followed by reasonable explanations.

Submission

- Please place your answers in the same order as the problem sheet and do not repeat problem descriptions, just organize them by problem number in a tidy manner.
- Please zip all the files, including one PDF, the security folder and the ldap folder. Name the zip file "`{your_student_id}.zip`", and submit it through NTU COOL. The zip file should not contain any other files, and the directory layout should be the same as listed below:

```
{your_student_id}/
+-- {your_student_id}.pdf
+-- security/
+---- {security script}
+---- ...
+-- ldap/
+---- {ldap script 1}
+---- {ldap script 2}
```

Grading

- Security accounts for 65 points while LDAP accounts for 35 points. The final score is the sum between them.
- It's possible you don't get full credits even if you have the correct answer. You should show how you get the answers step by step and list the references.
- Tidiness score: 3 bonus points, graded by TA.
- Final score = NA score + SA score + tidiness score.

Security

說明

- 請不要用任何形式干擾其他人作答，或不是以解題為目的來攻擊本作業的各項設施。經舉發查證屬實者，將會受到非常嚴厲的懲罰。
- Security 的所有題目分數加總是 90 分，但超過 65 分會以 65 分計。
  你可以斟酌不作答某些題目。
- 對於所有標記 (*CTF*) 的題目，請至 Google 表單 上傳 flag。
  所有題目的 flag 的格式都是 `HW4{XXX}`。
- 如果你有寫了 script 或程式來進行解題，請在作業的 zip 中附上檔案，放在 security 資料夾底下，並在 report 中提及。
- 動手操作的題目都需要詳細說明自己是如何做到的。請說服批改助教「你是真的有自己想過」還有「你是真的懂」。
- 即便沒解出來也請儘量作答，可以寫下錯誤的嘗試或是網路上搜尋到的資料。批改的助教將會依照方向與距離答案多遠來給予部份分數。
- 只要不是抄襲或作弊，非常歡迎你嘗試非預期解。

1. CIA Triad & Threat Modeling (10 points)

課堂上有提到 CIA 一般用來當作資訊安全的準則，其中三個字母分別為 confidentiality, integrity 和 availability，其實也就是一個「正常的服務」所應具備的要素。

(1) (3%) 請舉出兩個現實生活中的資安事件，並說明其違反 CIA 的哪幾項，並說明原因。

為了達成 CIA，我們會透過 threat modeling 來搞清楚我們可能會面對的攻擊手法，並針對攻擊做出相應的防禦。Threat modeling 的概念並非僅適用於資訊安全，最早甚至可以追溯至西元前的戰爭時期，以下的題目請針對各題的所要提供的服務（system）和安全上的需求（security requirement），提出小於 3 項的假設（assumption）、2 種不同的 threat model （不可太過於相似，否則會斟酌扣分）以及應對措施，詳情請參考以下例題。

例題

- system: 系上網路列印服務
- security requirement: 同學們可以使用網路列印功能，在送出請求的三分鐘之內取得列印完成的印刷品

參考解答

- assumption:
  1. 電子設備的電子元件皆狀態良好
- threat model:

Threat Model	Countermeasure
有人嘗試利用網路列印頁面的網頁漏洞來攻擊服務	定期將 server 更新至最新版本
有人透過大量列印來耗盡印表機的資源（紙張或碳粉匣）	在資源剩餘量低落時，限制每個人的使用量，並通知管理員補充列印資源

題目 (2 points per problem)

(2)
- system: 個人筆電
- security requirement: 沒有被擁有者允許的人不能使用

(3)
- system: 簡訊實聯制
- security requirement: 任何人皆以自己的真實身份進行實聯制掃描並傳送簡訊

(4)
- system: Nasa 線上期末考
- security requirement: 考試期間，各組不得以任何方式與非同組的人類進行交流

2. Web Security (23 points)

`OWASP Juice Shop`: https://github.com/juice-shop/juice-shop

`OWASP Juice Shop` 是一個相當**不安全**的網頁服務，一般用於資安相關的訓練或競賽。其中包含了 OWASP Top Ten 以及其他現實生活中的資安漏洞。在本題中，請參考上方連結自行架設一個 `OWASP` 伺服器（建議用 Heroku 架設以方便保存進度），並完成以下要求。

Note. 你可以在 `/#/score-board` 找到 Scoreboard

(1) (15%) 拿到 15 分後附上 Scoreboard 截圖（不需附上過程）

(2) (3%) 從 OWASP Top Ten 2021 中選擇三個介紹並簡單舉例（e.g. Injection 就是利用某某某來進行攻擊，像是某某某就可能被如此如此進行 injection）

(3) (5%) 簡單介紹 CSRF 原理

3. Linux Q&A (12 points)

在 2022 年 4 月 17 日的德田館，Luisa 發現 Terrance 正在研究即將在四天後釋出的 Ubuntu 22.04，他好奇的湊過去想看看有沒有什麼有趣的新功能，結果新功能沒見到幾個，卻意外發現 Terrance 竟然對 Linux 系統一無所知！連指令也只會用最最基本的 "rm -rf /"！身為 Nasa 團隊成員，Luisa 認為每個人都應該對 Linux 系統有最基本的認識，因此準備了好幾道題目來測驗 Terrance，並讓他在十四天後交出答案。在努力了幾天後，Terrance 還是有幾道題不太會，只好放在 NASA HW4 裡，讓大家來幫他解答。

(1) (3%) 請問在現行 Linux 架構下，使用者密碼一般會被存在哪個檔案裡？

另外，在仔細研究這個檔案之後，你會發現只有 `root` 有權限進行讀寫，那麼一般使用者又是如何使用 `passwd` 來達到更改密碼的效果呢？

(2) (3%) 如果你有一台暴露在網際網路上的 server，就會發現每次 ssh 上去時，shell 顯示自從你上次登入以來有很多 login failure。請以 Ubuntu 為例（版本 $\geq$ 14.04），找到這些登入嘗試的 log 被放在哪個檔案，並說明那個檔案裡存了哪些資訊。

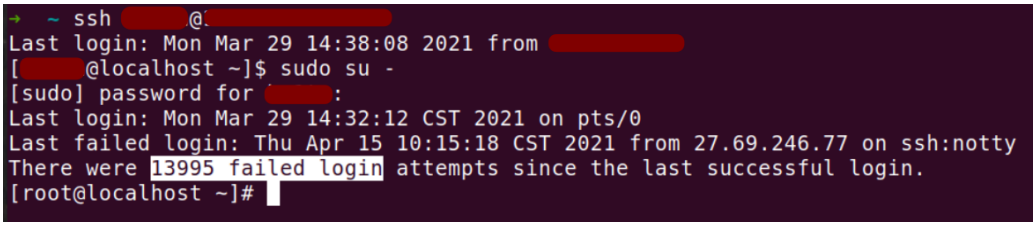

Figure 1: 很多人來敲門

(3) (3%) 在一台 Linux 電腦上，存在著非常多我們從來就不知道的使用者，不信的話連上工作站執行 `cat /etc/passwd` 就可以看到了。例如說 http 這個使用者，就是用來處理跟網頁伺服器有關的工作；systemd-network 這個使用者，就是用來處理跟電腦網路有關的工作。請說明為什麼這些工作需要額外創建專門的使用者來處理，並舉出如果全部都用 root 使用者來執行的話會有什麼安全問題。

(4) (3%) 在期中考題中，有一題關於 Docker Escape 的題目 (DinD)，Docker 的實作仰賴許多 Linux kernel 的特性，像是 Linux namespaces 和 cgroups 等等，請簡單介紹何謂 Linux namespaces 並舉出兩種不同形式的 Linux namespaces 進行進一步說明。

4. Cryptography (24 points)

Note. requirements.txt 包含本大題可能用到的 python 套件，可以用以下指令安裝：

```
$ pip install -r requirements.txt
```

(1) (7%) (*CTF*) 當年 Qpoi 第一次聽到 caesar cipher 和 block cipher 時，便設計了一個加密算法，並將它稱為 caesar block cipher。本題你將會拿到用該算法加密的 flag 和用來加密的原始碼，請從中還原出原本的 flag。

p.s. 此題的 flag 在 `HW4{}` 中的字串皆為小寫英文字母，且具有意義

- 本題需要用到的檔案在 這裡，其中有
  - `encrypt.py`: 將 flag 加密
  - `output`: 執行 python encrypt.py 的輸出

(2) (8%) (*CTF*) 老師上課有簡單提到 Diffie-Hellman，是一個簡單卻又強大用來製造 shared key 的方法。在密碼學中，有時就算方法本身沒有嚴重的問題，若實作出現漏洞也可能使系統出現漏洞。

而在本題的情境裡，Alice 和 Bob 利用 Diffie-Hellman 建立 shared key 後，Alice 利用這個 shared key 將祕密用 one-time pad 傳給 Bob。你身為 Eve 的好朋友，決定想辦法窺探這個祕密。

- 本題需要用到的檔案在 這裡，其中有
  - `encrypt.py, utils.py`: 用來建立 shared key 的原始碼
  - `hack.py`: Eve 先幫你寫好的一些 code
  - `output`: 執行 `python encrypt.py` 的輸出
- Hints
  - 了解一下 Diffie-Hellman 的運作
  - 仔細看一下 Alice 和 Bob 的實作

(3) 除了系統的設計架構，有時造成漏洞的可能是使用者本身，像是過於簡單的密碼或將密碼記在公開的記事本？

連結 展示了針對 Mac 電腦的攻擊手法，請利用相同的原理來嘗試解出以下 Ubuntu 的密碼

再次提醒，中華民國刑法 妨害電腦使用罪：

- 第 358 條
  無故輸入他人帳號密碼、破解使用電腦之保護措施或利用電腦系統之漏洞，而入侵他人之電腦或其相關設備者，處三年以下有期徒刑、拘役或科或併科三十萬元以下罰金。
- 第 359 條
  無故取得、刪除或變更他人電腦或其相關設備之電磁紀錄，致生損害於公眾或他人者，處五年以下有期徒刑、拘役或科或併科六十萬元以下罰金。

請大家帶著嚴肅與戒慎恐懼的心來完成這一小題。

本題請用 Willy's Ubuntu 進行測試（或在 這裡 下載），嘗試登入 Willy 的帳號，並回答

- (7%) Willy 的密碼
- (2%) (*CTF*) 桌面上的 flag

注意事項：

- 提供解法時，解法不可包含直接分析所提供的 ova 檔，但可以透過此方法獲取靈感

Hint:

- 或許有用？
- 仔細看看上方關於 Mac 攻擊手法的影片

5. WiFi Hacking (21 poitns)

(1) (7%) 德田館 217 實驗室是個神秘的領域，裡面住著強大的生靈，一般人只要經過，就會被裡面傳出的聲音嚇得不輕。那些聲音究竟是什麼，身為修習 NASA 的勇者，希望你能前往此處去調查一番。

- 請在德田館 **217 實驗室外面的走廊**進行解題。
- 本題的目標是破解 "Mysterious Room" 這個 WiFi 的密碼，並連上這個 WiFi。
- 當發現訊號不好時，可以盡量靠近 217 實驗室門邊
- WiFi 密碼的格式似乎是某人的手機號碼⋯？
- 如果一點頭緒也沒有，可以嘗試搜尋 "WPA2 PSK cracking" 或類似的關鍵字⋯

(2) (7%) (*CTF*) 現在，你似乎漸漸有點頭緒了，你聽出一些像是 LDAP、ADA、DSA 等的艱難字詞混雜在那些怪異的聲響之中，但不知為何，這些聲音中似乎混雜著一些雜訊 ⋯⋯

- 請在至少完成第一小題之後再解這題。
- 本題的目標是找出哪個可疑的 IP 位址連上了 "Mysterious Room"，並破解這個 IP 位址正在進行的 TCP 連線內容。
- 如果一點頭緒也沒有的話，可以嘗試搜尋 "decrypt 802.11 traffic" 或類似的關鍵字⋯

(3) (7%) (*CTF*) 原來，剛才的雜訊是惡魔想要強行闖入這片神聖領域時所產生的噪音，為了維護這片淨土，你現在必須立刻阻止惡魔！

- 請在至少完成第一小題之後再解這題。
- 本題的目標是讓上一題的可疑 IP 位址無法穩定連上 “Mysterious Room” 連續 40 秒。
- 若要查看當前狀態，請在 217 實驗室外面連上 “battle-field” 這個 WiFi，並透過 “battle-field” 連上 這個網址 來確認解題進度和獲取 flag。
  - 上述網址只有在連上 “battle-field” 之後才能查看。
  - “battle-field” 的 WiFi 密碼和 “Mysterious Room” 的密碼相同。
- 如果一點頭緒也沒有的話，可以嘗試搜尋 “wifi deauthentication” 或類似的關鍵字⋯

LDAP

我們在系上是使用 LDAP (Lightweight Directory Access Protocol，輕型目錄存取協定)[1]來管理眾多的使用者帳號。大家日常使用的 217 工作站、CSIE Wi-Fi、以及 CSIE Mail 等服務皆需透過 LDAP 來驗證以及獲取使用者的相關資訊。在這份作業中，你需要設定兩台虛擬機來模擬系上工作站的架構：一台 LDAP Server 以及一台工作站，兩台之間請透過橋接網路介面卡連接。

一些共通的規定

- 請詳細說明每一題你按照順序做了什麼，以及列出你使用的參考資料。我們也鼓勵你寫出你遇到的問題（例如明明已經調了 xxx 設定，但是還是沒有達成 ooo 的效果），以及查了什麼資料找到如何解決該問題。
- 所有小題中都請使用 TLS 的方式與 LDAP 連線。

設定 (23%)

1. 請安裝一台 CentOS 7 的虛擬機器，並在上面架設 OpenLDAP[2]。LDAP 的設定請按照下列的要求：(3%)
   - olcSuffix: dc=nasa,dc=csie,dc=ntu
   - olcRootDN: cn=nasa,dc=nasa,dc=csie,dc=ntu
   - 設置 dc=nasa,dc=csie,dc=ntu 的節點，並在下面設置 root (nasa) 以及 people, group 兩個 organizationalUnit
   - LDAP root password: nasa2022

   請提供安裝步驟以及使用 `ldapsearch` 查詢所有 `dc=nasa,dc=csie,dc=ntu` 子樹（subtree）下資訊的結果。若有需要，你可以參考第八次 Lab 投影片中的步驟（該 Lab 安排在此份作業釋出後的下一週）
2. 請安裝一台 Arch Linux 的機器作為工作站，並在上面安裝 SSSD (System Security Services Daemon，系統安全服務背景服務程式)，安裝完後請連上前一小題的 LDAP Server。(8%)
3. 在 LDAP 中創造兩個使用者，其中一個可以在工作站上使用 `sudo` 指令成為 `root`，而另一個不能。(6%)
4. 修改以下權限 (6%)：
   (a) 使用者不可以修改其他使用者的資料（例如修改別人的密碼）。
   (b) 使用者不可以修改自己的家目錄路徑，使用者編號，群組編號。
   (c) 只有輕型目錄存取協定客戶端的網際協定位址可以連上輕型目錄存取協定伺服器。

腳本 (12%)

1. 請使用 Python 3.6.8（CentOS 7 上使用 `yum` 安裝會得到的版本）或是 Shell Script 寫出一個可以新增使用者的腳本，且這個腳本要可以在 LDAP Server 跑起來。這個腳本輸入的第一行是使用者名稱，第二行是該使用者的密碼。新增的使用者的家目錄位於 `/home/<user name>`。提示：你可能會想要遍歷所有使用者來找到一個沒有被用過的使用者編號（uid）。(6%)

[1] https://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol
[2] https://www.openldap.org/

2. 請使用 Python 3.10.4 （Arch Linux 上使用 `pacman` 安裝會得到的版本）或是 Shell Script 寫出讓使用者可以修改他們的名字（`givenName`）的腳本。這個腳本會跑在 SSSD 的客戶端。這個腳本輸入的第一行是該使用者自己的密碼，第二行是欲換的新名字。(6%)

Notes

- 腳本寫完後請記得將其放入 ldap 資料夾底下，並一同繳交至 NTU COOL 上。
- 若您有在自己的腳本中使用任何額外的函式庫，請在 PDF 中註明其安裝方式並簡易說明該函式庫之功能。