1 Úvod

Cieľom tohto projektu bolo získať čo najviac tajomstiev ukrytých na serveroch v privátnej sieti. V nasledujúcich sekciách budú popísané spôsoby ako som dospel k jednotlivým tajomstvám.

2 Zmapovanie siete

Prvým krokom po prihlásení na `bis` server (`bis.fit.vutbr.cz`) bolo zistenie aktuálneho sieťového nastavenia pomocou príkazu `ip address`. Ďalším krokom bolo detailnejšie preskúmanie siete, ku ktorej bol `bis` server pripojený. Za týmto účelom som použil nástroj `nmap`[1]. Vo výpise (viz Listing 1) môžme vidieť `IP` adresy všetkých serverov na ktorých boli ukryté tajomstvá, ako aj služby bežiace na týchto serveroch.

```
Nmap scan report for s2 (192.168.122.5)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2 (protocol 2.0)

Nmap scan report for s5 (192.168.122.36)
PORT    STATE SERVICE VERSION
21/tcp  open  ftp     vsftpd 2.3.4
22/tcp  open  ssh     OpenSSH 8.2 (protocol 2.0)
111/tcp open  rpcbind 2-4 (RPC #100000)
Service Info: OS: Unix

Nmap scan report for s3 (192.168.122.55)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.2 (protocol 2.0)

Nmap scan report for s4 (192.168.122.211)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.2 (protocol 2.0)
80/tcp   open  http    Apache httpd 2.4.46 ((Fedora))
3306/tcp open  mysql?

Nmap scan report for s1 (192.168.122.234)
PORT    STATE SERVICE VERSION
22/tcp  open  ssh     OpenSSH 8.2 (protocol 2.0)
80/tcp  open  http    Apache httpd 2.4.46 ((Fedora))
111/tcp open  rpcbind 2-4 (RPC #100000)
888/tcp open  ypserv  1-2 (RPC #100004)
```

Listing 1: Získaný výstup použitím príkazu `nmap -sV 192.168.122.0/24 -open`. pozn.: Nerelevantné časti výstupu z pohľadu projektu boli pre väčšiu prehľadnosť vynechané.

3 Tajomstvo A

Po zmapovaní siete popísanom v predchádzajúcej sekcii som si všimol, že na každom z nájdených serverov beží služba `ssh`. Preto mojím ďalším krokom bolo preskúmanie adresára `~/.ssh` za účelom nájdenia nejakých užitočných informácií o tom, ako sa pripojiť na ďalšie z nájdených serverov.

[1] nmap (1) - Linux Man Pages

V spomínanom priečinku som našiel súbor `config`, z ktorého obsahu vyplývalo, že ako užívateľ `server1` sa môžem pripojiť k serveru `s1` a ako užívateľ `server2` sa môžem pripojiť k serveru `s2`.

Po zadaní príkazu `ssh server1@s1` som sa úspešne pripojil na server `s1` bez nutnosti zadania hesla. Nasledoval prieskum servera. Ako prvý som preskúmal domovský adresár, kde som po zadaní príkazu `ls -la` našiel podozrivý adresár s názvom `.secret`. Po otvorení tohto adresára som našiel súbor `cipher` a ešte jeden spustiteľný súbor s názvom `generate_secret_from_decrypted_cipher`.

Po otvorení spustiteľného súboru v editore `vim` som zistil, že sa jedná o `bash` skript. Ten na základe vstupného súboru (čo by podľa popisu mal byť dešifrovaný text) spočíta `sha256` hash tajomstva A. Za týmto účelom som teda chcel dešifrovať text v súbore `cipher`.

Najprv som ale musel prísť na to o akú šifru sa jedná. Na to som použil online nástroj BOXENTRIQ[2], ktorý tvrdil, že by mohlo ísť o Stĺpcovú transpozičnú šifru. Rovnaký nástroj taktiež disponoval aj funkciou[3] na lámanie tejto šifry. Po zadaní šifrovaného textu nám ako jedno z možných riešení navrhol kľúč `abcde`, ktorého použitím došiel k dešifrovanému textu: `PANAMA BOTSWANA CANADA NICARAGUA ARGENTINA GUYANA`[4]. Odhadnutý kľúč teda vyzerá byť správny keďže výstupom sú plnovýznamové slová (názvy krajín). Po spustení daného skriptu nad pôvodným dešifrovaným textom (bez medzier) `./generate_secret_from_decrypted_cipher PANAMABOTSWANACANADANICARAGUAARGENTINAGUYANA` sme tak dostali tajomstvo A.

4 Tajomstvo B

Keďže z výstupu nástroja `nmap` vieme, že na serveri `s1` beží služba `http`. To ma zaviedlo do adresára `/var/www/html`, kde som narazil na problém, že som nemal práva na zobrazenie obsahu. Na stiahnutie webového obsahu som teda použil nástroj `curl` a po zadaní príkazu `curl http://s1` na `bis` serveri som dostal html kód webovej stránky. Vrátený kód som zanalyzoval a zistil som, že ide o formulár používajúci `POST` metódu a majúci jednu položku `url`. Skúsil som teda vytvoriť `POST` požiadavku `curl -d url=vutbr.cz http://s1` a ako odpoveď som dostal pole odpovedí `DNS` protokolu (ip adresu a hostname mail serveru). Z analýzy teda vyzerá, že táto stránka vykonáva DNS lookup. Zároveň som si zo syntaxe vrátenej odpovede všimol, že vrátené pole má formát používaný v jazyku `PHP`. Overil som si to príkazom `curl -i http://s1` a dozvedel som sa, že serverová časť je naozaj implementovaná v jazyku `PHP/7.4.10`. Zo zistených informácií vyplýva, že by šlo využiť tzv. command injection attack. Po niekoľkých neúspešných pokusoch som vytvoril `POST` požiadavku vyzerajúcu nasledovne `curl -d url=";ls"http://s1`. Z vrátenej odpovede je vidieť, že v adresári `/var/www/html` sa nachádzajú súbory `index.php` a `secret.txt`. Na výpis obsahu súboru `secret.txt` som použil `curl -d url=";cat secret.txt"http://s1` a tým som získal tajomstvo B.

5 Tajomstvo C

Na základe informácií nájdených v adresári `~/.ssh` na `bis` serveri som bol schopný sa pripojiť na server `s2` zadaním príkazu `ssh server2@s2`. Po úspešnom pripojení na server nasledovalo prehľadanie zaujímavých súborov. Pri tomto hľadaní som došiel až do adresára `/var/spool/`. Podľa špecifikácie na `linuxfoundation` tento adresár obsahuje dáta, ktoré čakajú na určitý spôsob spracovania, či už nejakým programom, užívateľom alebo administrátorom a po spracovaní by mali byť (alebo často sú) odstránené. To vo mne vzbudilo záujem a v adresári `mail` som našiel neprázdny súbor s názvom `joe` (mail inbox užívateľa `joe`). K tomuto súboru, ale mal výhradné práve iba užívateľ s rovnakým menom. Keďže sme na Linuxe (Fedora 32) skúsil som príkaz na zmenu užívateľa (`su`) a na moje prekvapenie bez vyžiadania hesla som zrazu bol prihlásený ako užívateľ `joe`. Po preskúmaní obsahu spomínaného súboru som narazil na tajomstvo C.

[2] BOXENTRIQ - Cipher Identifier and Analyzer
[3] BOXENTRIQ - Columnar Transposition Cipher Tool
[4] Medzery boli do dešifrovaného textu doplnené ručne pre lepšiu čitateľnosť.

6 Tajomstvo D

Toto tajomstvo sa nachádzalo na serveri `s2` a teda postup pripojenia na server je rovnaký ako v prípade tajomstva C. Hneď v domovskom adresári som našiel spustiteľný súbor s názvom `secret_app`. Po spustení sa zobrazila hláška `"Weclome in secret application!!"` a následne aplikácia čakala na zadanie vstupu. Najprv som si myslel, že po zadaní nejakého konkrétneho vstupu ("hesla") mi aplikácia vráti tajomstvo. No po vyskúšaní vstupov ako: `key`, `bis`, `secret`, `tajemstvi`,... som dospel k záveru, že tadiaľ cesta nevedie. Po niekoľkých neúspešným pokusoch som sa skúsil pozrieť na obsah tohto binárneho súboru pomocou editora `vim` a v ňom sa ukrývalo tajomstvo D.

7 Tajomstvo E

Podobne ako na serveri `s1` som preskúmal obsah domovského adresára užívateľa `server2` a v adresári `~/.ssh` som opäť narazil na súbor `config`. Tentokrát ale súbor obsahoval informácie, že na server `s3` sa môžem pripojiť ako užívateľ `joe` a na server `s4` ako užívateľ `server`.

Na základe získaných informácii som sa teda chcel pripojiť na server `s3` ako užívateľ `joe`. No po zadaní príkazu `ssh joe@s3` bolo odo mňa vyžadované heslo. Na internete som si našiel zoznam najčastejších hesiel[5] a vyskúšal som ich, až pokým nenastal úspech. Nakoniec som sa pripojil na server `s3` s heslom `password1` (19. najpoužívanejšie heslo podľa nájdeného zoznamu). Hneď po pripojení som preskúmal domovský adresár užívateľa `joe` a našiel som súbor `secret.txt` v ktorom bolo tajomstvo E.

8 Tajomstvo F

Pri ďalšom prehľadávaní servera `s3` som v koreňovom adresári okrem bežných priečinkov ako `bin, dev, etc, var, ...` našiel aj adresár `database_backup`, v ktorom bol súbor `2020_dump`. Po otvorení tohto súboru som na prvom riadku našiel text `"GDBM dump file created by GDBM version 1.18.1."`. Z manuálových stránok utility `gdbm_dump`[6], pomocou ktorej sú súbory tohto typu vytvárané, som sa dozvedel, že tento súbor môžem dať ako vstup utilite `gdbm_load`[7], ktorá dokáže vytvoriť pôvodný databázový súbor. Po zadaní `gdbm_load 2020_dump ~/secret_db.gdbm` a následnom preskúmaní súboru `secret_db.gdbm` pomocou interaktívneho nástroja `gdbmtool`, som našiel tajomstvo F.

9 Tajomstvo G

Na základe informácií nájdených na serveri `s2` som sa pripojil na server `s4` ako užívateľ `server`. V domovskom adresári som našiel adresár `libgcd`. Po zobrazení obsahu tohto adresára príkazom `ls -la`, som si všimol, že sa jedná o `git` repozitár. Zadaním príkazu `git status` som zistil, že lokálna vetva je o jeden commit pred vetvou `origin/master`. Následne som pomocou príkazu `git log --stat` zistil, že jediným súborom, ktorý bol posledným commitom zmenení bol už neexistujúci súbor `CMakeList.txt`. Správou tohto commitu bola veta `"Super secret commit message"` a preto som sa chcel pozrieť na konkrétne zmeny vykonané v súbore `CMakeList.txt`. To som spravil zadaním príkazu `git show HEAD` a našiel som tajomstvo G.

[5] List of the most common passwords
[6] gdbm_dump (1) - Linux Man Pages
[7] gdbm_load (1) - Linux Man Pages

10 Tajomstvo H

Pri odhaľovaní tohto tajomstva som postupoval podobne ako pri tajomstve B. Na serveri `s4` beží služba `http` podobne ako na serveri `s1`. Znova ma to zaviedlo do adresára `/var/www/html`. Narozdiel od predchádzajúceho kroku, som sa dokázal pozrieť na obsah tohto adresára no k súboru `index.php`, ktorý sa tam nachádzal, som už práva nemal. Zase som použil nástroj `curl` a z výsledku som zistil, že ide o aplikáciu, ktorá na základe zadaného mena a hesla vráti nejaké informácie o užívateľovi. Z výpisu dostupných služieb na serveri `s4` vidíme, že tam beží aj služba `mysql`. Z analýzy stránky zase vyplýva, že informácie o užívateľoch, ktoré vracia sú s vysokou pravdepodobnosťou uložené v nejakej databáze. To ma viedlo k myšlienke použiť SQL injection attack. Na vytvorenie požiadavky som sa inšpiroval riešením dostupným na portáli `w3schools`[8]. Po poslaní `POST` požiadavky v tvare `curl -d 'name=" or ""=" &password=" or ""="' http://s4`, som dostal odpoveď obsahujúcu informácie o všetkých užívateľoch a rovnako aj tajomstvo H. Trik spočíva v tom, že `or "" = ""` po dosadení do SQL zabezpečí, že podmienka bude vždy pravdivá, keďže prázdny reťazec sa rovná prázdnemu reťazcu a teda ako výsledok takejto požiadavky dostaneme celú tabuľku.

11 Tajomstvo I

Zo zmapovanie siete som vedel, že na serveri `s5` beží `ftp` služba. Z `bis` serveru som sa teda pokúsil prihlásiť zadaním príkazu `ftp s5`. Zistil som, že na serveri beží `vsFTPd` vo verzii `2.3.4`, na ktorú je možné použiť smajlíkový útok[9]. Ako užívateľské meno som teda použil reťazec ukončený smajlíkom `aa:)` a heslo som nezadal žiadne (pri tomto útoku na tom nezáleží). Po tomto prihlásení sa mi vypísala správa `"220 Opened port 57432, take a look ;)"` a tak som sa z ďalšieho terminálu zadaním príkazu `nc s5 57432` pripojil na tento otvorený port, kde som našiel tajomstvo I.

12 Tajomstvo J

Z výpisu nástroja `nmap` som videl, že na serveri `s5` beží služba `rpcbind`. Keďže som sa s touto službou nikdy predtým nestretol, pustil som sa do hľadania. Zistil som, že `rpcbind` utilita mapuje RPC služby na porty, na ktorých následne tieto služby načúvajú. Následne som pomocou príkazu `rpcinfo -p s5` zistil, že na serveri `s5` je na porte 613 služba `ypbind`. Z výpisu nástroja `nmap` je vidieť, že na serveri `s1` beží `ypserv`. Tieto informácie naznačujú, že v tejto sieti funguje protokol NIS[10] typu klient-server, kde klientom je server `s5` a serverom je server `s1`. Z manuálových stránok `ypserv`[11] som zistil, že databázové súbory sa nachádzajú v adresári `/var/yp`. Pri prehľadávaní tohto adresára som narazil na `Makefile`, ktorým sa vytvárajú NIS databázy. Preskúmal som teda tento súbor a všimol som si, že informácie o užívateľoch, ktoré sa distribuujú NIS klientom sú uložené v domovskom adresári serveru `s1`. Po zadaní príkazu `ls -la` som si všimol, že ako užívateľ `server1` môžem zapisovať do súboru `shadow` obsahujúceho šifrované heslá užívateľov. Vytvoril som teda nové heslo pre užívateľa `bis_user` pomocou príkazu `mkpasswd -m SHA-512 <heslo>` a pôvodné heslo v súbore `shadow` som nahradil mnou vytvoreným heslom. Poslednou vecou, ktorú bolo potrebné spraviť pre úspešné pripojenie na server `s5` bola distribúcia zmenených údajov na server `s5`. Z dokumentácie som zistil, že server distribuuje informácie pri inicializácii a tak som použil príkaz `sh /usr/lib64/yp/ypinit -m`. Následne som sa prihlásil na server `s5` ako užívateľ `bis_user` s novým heslom `<heslo>` a v súbore `~/.secret/secret.txt` som našiel tajomstvo J.

[8] SQL Injection
[9] FTP Anonymous Login Issue and Smiley Face Attack
[10] Network Infromation Service je protokol pre distribúciu systémových konfiguračných dat, ako užívateľské mená medzi počítačmi v počítačovej sieti.
[11] ypserv(8) - Linux man page