به نام پروردگار هدایت کننده به راه راست

دانشگاه اصفهان

دانشکده مهندسی کامپیوتر

ترم تحصیلی ۰۲-۰۳

مستند پروژه درس شبکه های کامپیوتری - فاز اول

استاد درس : دکتر احمدرضا منتظرالقائم

طراحان : امیرعلی گلی، محمدحسین دهقانی، مهرشاد جعفری، مهدی قنبرزاده، محمدحسین رنگ‌رز

فاز اول

ان‌مپ (Nmap) یک ابزار قدرتمند مورد استفاده توسط مدیران شبکه، متخصصان امنیت، و حتی هکرها برای کاوش، بررسی و درک بهتر شبکه‌های کامپیوتری است. نام ان‌مپ مخفف "Network Mapper" است.

- ان‌مپ به کاربران کمک می‌کند تا دستگاه‌های فعال در یک شبکه را شناسایی کنند، سرویس‌ها و برنامه‌های در حال اجرا بر روی این دستگاه‌ها را شناسایی کنند و مشکلات امنیتی را مشخص کنند.
- در این پروژه، از دانشجویان می‌خواهیم برخی از قابلیت‌های این ابزار را از جمله بررسی وضعیت آنلاین بودن هاست(HOST)، بررسی پورت‌ها و سرویس‌های باز و همچنین شبیه‌سازی متدهای GET و POST در پروتکل HTTP پیاده‌سازی کنند.

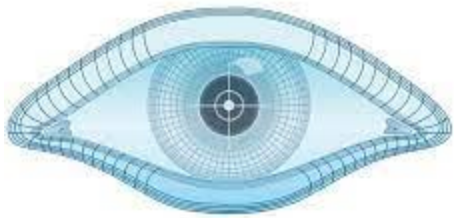
نکات قابل توجه

فاز اول قابل پیاده سازی با زبان های برنامه نویسی java, C++, Python, C# است و برای انجام پروژه شما قادر به تشکیل گروه های **دو نفره** هستید.

مهلت تحویل این فاز تا ۱۴ آذرماه ساعت ۱۲ شب می‌باشد.

تذکر: توجه داشته باشید شما قادر به استفاده از هیچ کتابخانه ای که قسمتی از پروژه ها را پیاده سازی کرده است نیستید.

فاز اول: ابزار Nmap



ان‌مپ یک ابزار بسیار قدرتمند است که توسط مدیران شبکه (Network Administrator) ، متخصصان امنیت (Security Expert) و حتی هکرها (Hacker) برای کاوش، بررسی و درک بهتر شبکه‌های کامپیوتری مورد استفاده قرار می‌گیرد.

نام این ابزار مخفف شده عبارت "Network Mapper" است. این ابزار به کاربر کمک می‌کند تا دستگاه‌هایی (Device) که در یک شبکه کامپیوتری فعال هستند را پیدا کند سرویس‌ها و برنامه‌هایی که روی آن دستگاه‌ها در حال اجرا هستند را شناسایی کند و حتی مواردی را که از لحاظ امنیتی، آسیب پذیر (Vulnerable) هستند را مشخص کند.

برای کسب اطلاعات بیشتر در مورد این ابزار می‌توانید به این لینک مراجعه کنید.

تعاریف مورد نیاز

ممکن است در حین خواندن این داکیومنت به یک سری تعاریف نیاز پیدا کنید. برای سادگی کار شما برخی از آن تعاریف آورده شده‌اند:

- **هاست (Host):** در مفهوم شبکه‌های کامپیوتری، هاست به دستگاه یا سیستمی اشاره دارد که قادر است به شبکه متصل شود و در شبکه‌ای حضور دارد. هاست می‌تواند یک کامپیوتر، سرور (Server)، روتر (Router)، گیتوی (Gateway) یا ... باشد. برای شناسایی هر هاست در شبکه یک IP منحصر به فرد به آن داده می‌شود.

- **سرویس (Service):** در تعریف شبکه، سرویس به یک نرم‌افزار یا پروتکل خاص اشاره دارد که بر روی یک هاست در شبکه اجرا می‌شود و به دیگر دستگاه‌های حاضر در شبکه خدماتی را ارائه می‌دهند.

- **پورت (Port):** در شبکه‌های کامپیوتری، پورت به یک عدد از 0 تا 65535 اشاره دارد که برای تعیین و شناسایی خدمات و برنامه‌ها مورد استفاده قرار می‌گیرد. هر پورت متناظر با یک خدمت یا برنامه خاص در یک هاست است و به آن امکان ارتباط و تبادل داده با سایر هاست‌های موجود در شبکه را می‌دهد.
  - **پورت باز (Open Port):** اگر در یک هاست پورتی در وضعیت باز قرار داشته باشد یعنی آن دستگاه به درخواست‌های ورودی به این پورت پاسخ می‌دهد و ارتباط با آن دستگاه از طریق آن پورت امکان پذیر است.
  - **پورت بسته (Close Port):** در نقطه مقابل پورت باز قرار دارد و اگر در دستگاهی، پورتی در این حالت قرار داشته باشد به آن معناست که هاست موردنظر به درخواست‌های ورودی به این پورت پاسخ نخواهد داد و ارتباط با آن دستگاه از طریق پورت ذکرشده امکان پذیر نخواهد بود.

هدف پروژه

در این پروژه قصد داریم تا دانشجویان پس از آشنایی با تعدادی از قابلیت‌های نرم‌افزار ان‌مپ ، به پیاده‌سازی برخی از قابلیت‌های ساده این ابزار قدرتمند بپردازند.

پیشنهاد:

توصیه می‌شود برای آشنایی بیشتر با این نرم‌افزار، برنامه را دانلود کرده و پس از نصب، تعدادی قابلیت‌های ساده آن را امتحان کنید. همچنین برای مشاهده نحوه کار این ابزار می‌توانید از این لینک به صورت آنلاین، برخی از قابلیت‌های آن را امتحان کرده و نتیجه را مشاهده کنید.

در ادامه تصاویری از محیط ابزار و همچنین وبسایت معرفی شده قرار داده شده است.

گزارش پورت ها

```
pentester@TryHackMe$ sudo nmap -sV 10.10.76.34

Starting Nmap 7.60 ( https://nmap.org ) at 2021-09-10 05:03 BST
Nmap scan report for 10.10.76.34
Host is up (0.0040s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE VERSION
22/tcp  open  ssh     OpenSSH 6.7p1 Debian 5+deb8u8 (protocol 2.0)
25/tcp  open  smtp    Postfix smtpd
80/tcp  open  http    nginx 1.6.2
110/tcp open  pop3    Dovecot pop3d
111/tcp open  rpcbind 2-4 (RPC #100000)
MAC Address: 02:A0:E7:B5:B6:C5 (Unknown)
Service Info: Host:  debra2.thm.local; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

در تصویر بالا کاربر پس از دادن IP هدف خود به ابزار ان‌مپ و استفاده از دستور sV- نتایج اسکن را که شامل شماره پورت، وضعیت هر پورت، سرویسی که روی آن پورت در حال اجراست و همچنین ورژن آن سرویس را به عنوان گزارش دریافت کرده است.

گزارش کامل

```
root@kali: /home/geek
File  Actions  Edit  View  Help

┌──(root💀 kali)-[/home/geek]
└─# nmap -A 192.168.2.107
Starting Nmap 7.91 ( https://nmap.org ) at 2021-02-09 15:20 EST
Nmap scan report for 192.168.2.107
Host is up (0.0011s latency).
Not shown: 977 closed ports
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to 192.168.2.104
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      vsFTPd 2.3.4 - secure, fast, stable
|_End of status
```

در این تصویر کاربر با دادن آپشن A- به ابزار درخواست گزارش کاملی از اسکن هدف را دارد.

مثال

در این مثال از وبسایت گوگل اسکن سریع گرفته شده است.

قابلیت‌های مدنظر جهت پیاده سازی

برنامه پیاده‌سازی شده توسط شما باید بتواند پس از دریافت آدرس IP هدف و یک بازه از پورت‌هایی که قصد بررسی آن‌ها را داریم عملیات‌های زیر را انجام دهد:

1. **بررسی وضعیت آنلاین بودن یا نبودن یک Host**
2. **بررسی محدوده‌ای از پورت‌های یک Host و گزارش پورت‌هایی که در حالت Open قرار دارند و همچنین سرویس‌هایی که روی آن پورت‌ها در حالت اجرا قرار دارند.**
3. **شبیه‌سازی متدهای GET و POST پروتکل HTTP**

تمامی قابلیت‌های خواسته شده به وسیله Socket Programming قابل پیاده سازی هستند. در ادامه به بررسی هرکدام از موارد گفته شده می‌پردازیم.

1. **بررسی وضعیت آنلاین بودن یا نبودن یک Host**

برای پیاده‌سازی این قابلیت برنامه باید تلاش کند یک ارتباط با هاست خواسته شده برقرار کند. در صورتی که این ارتباط با موفقیت برقرار شد می‌توان دریافت

که هاست موردنظر آنلاین است و در غیر این صورت هاست آفلاین شناخته شده و نتیجه گزارش داده خواهد شد.

2. **بررسی پورت‌ها**

برای پیاده‌سازی این قابلیت برنامه باید پس از دریافت IP یک هاست و یک رنج از پورت‌های مدنظر جهت اسکن شدن، تک تک پورت‌ها را مورد بررسی قرار دهد و در صورتی که پورت در وضعیت باز قرار داشت؛ شماره آن پورت و سرویسی که روی آن پورت درحال اجرا است را برگرداند.

```
PS C:\Users\     \Desktop> python nmap.py 1.1.1.1 80 81
1.1.1.1 is online
open port detected: 1.1.1.1     -- Port: 80     -- Service: http
```

نمونه ای از ورودی و خروجی مدنظر برای قابلیت‌های شماره 1 و 2 را مشاهده می‌کنید.

3. **شبیه‌سازی متدهای GET و POST**

POST و GET از متدهای درخواست پروتکل HTTP (HTTP Request Methods) هستند. GET برای فراخوانی داده مورد استفاده قرار می‌گیرد و متد پست برای ثبت کردن یک مقدار جدید. برای پیاده سازی این قابلیت، یک فایل server.py در اختیار شما قرار خواهد گرفت. این فایل یک سرور را شبیه‌سازی می‌کند که اطلاعات تعدادی از کاربران را نگهداری می‌کند. این اطلاعات در تصویر زیر قابل مشاهده هستند.

```
users = {
    'user1': {'name': 'Alice', 'age': 30},
    'user2': {'name': 'Bob', 'age': 25},
    'user3': {'name': 'Charlie', 'age': 35},
}
```

شما باید در برنامه پیاده‌سازی شده خودتان قابلیتی را به وجود بیاورید که ابزار بتواند با متد GET اطلاعات کاربر خواسته شده را که با ID آن کاربر (ستون اول که شامل مقادیر user1, user2, user3 می‌باشد ID کاربران را مشخص می‌کند) داده می‌شود پیدا کرده و مقادیر آن را گزارش دهد. فرمت قابل قبول برای برنامه سرور به شرح زیر است:

GET user_id

که شما با وارد کردن ID کاربر مدنظر می‌توانید اطلاعات آن را مشاهده کنید. به عنوان مثال به تصویر زیر دقت کنید.

```
Enter 'GET user_id' or 'POST user_name user_age' to simulate a request: GET user1
Response from the server:
HTTP/1.1 200 OK
Content-Type: application/json

{'name': 'Alice', 'age': 30}
Enter 'GET user_id' or 'POST user_name user_age' to simulate a request: |
```

همچنین ابزار باید این قابلیت را داشته باشد که بتواند با استفاده از متد POST و دریافت نام و سن کاربر، اطلاعات آن کاربر را به مجموعه اطلاعات کاربرها اضافه کند. فرمت قابل قبول برای برنامه سرور به شرح زیر است:

POST user_name user_age

دستور POST پس از ساخت هر کاربر جدید یک ID منحصربه فرد برای او می‌سازد که به فرمت زیر است:

user + {1 + شماره آخرین یوزر ساخته شده}

به عنوان مثال ID اولین یوزر ساخته شده برابر خواهد بود با user4.

نکته: لازم به ذکر است که در هر دو دستور مقادیر باید با کاراکتر space از هم جدا شده باشند.

به عنوان مثالی برای دستور POST به تصویر زیر دقت کنید:

```
Enter 'GET user_id' or 'POST user_name user_age' to simulate a request: POST Arthur 43
Response from the server:
HTTP/1.1 200 OK

User data updated
Enter 'GET user_id' or 'POST user_name user_age' to simulate a request: GET user4
Response from the server:
HTTP/1.1 200 OK
Content-Type: application/json

{'name': 'Arthur', 'age': 43}
Enter 'GET user_id' or 'POST user_name user_age' to simulate a request: |
```

همانطور که مشاهده می‌کنید سرور پس از دریافت اطلاعات کاربر جدید آن اطلاعات را تحت ID جدید user4 ذخیره کرده که از طریق آن نیز قابل دسترسی و فراخوانی هستند.