

Blog Simple.



Mobile app traffic analysis — For Fun

In Security Tags mobile analysis, mobile hacking, mobile pentest, traffic analysis September 1, 2018 301 Views

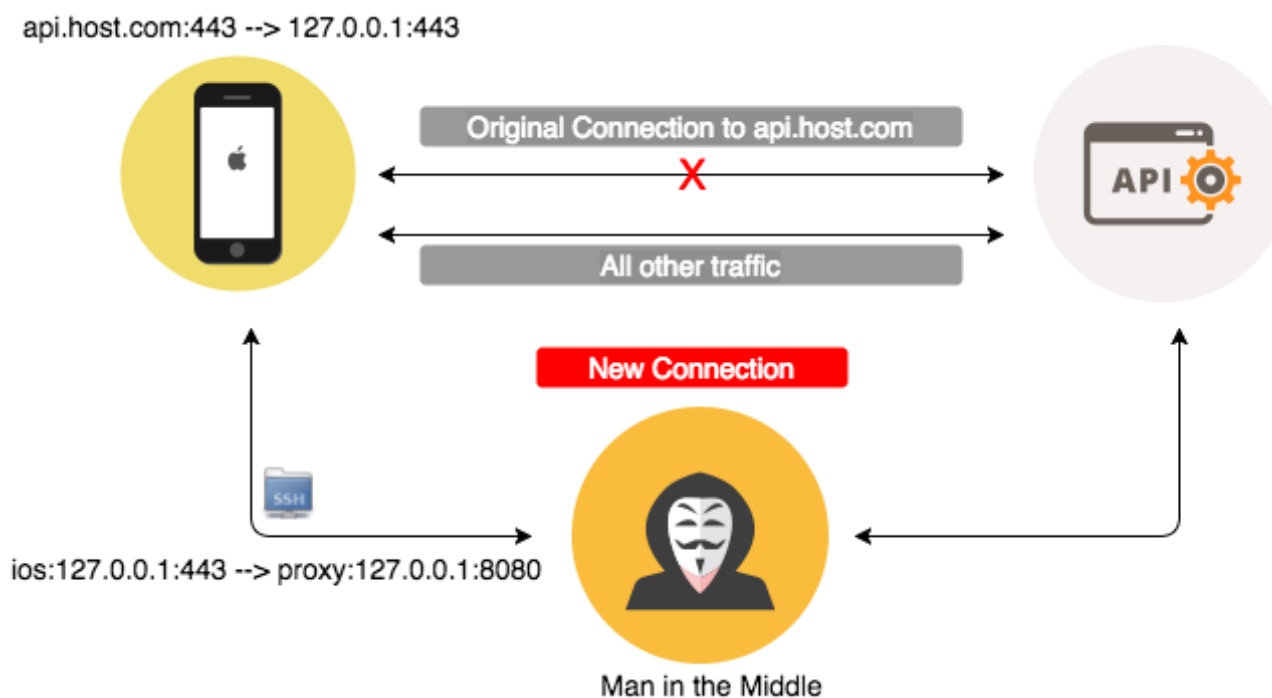


Một trong các công việc của việc pentest các ứng dụng mobile là phân tích các traffic mà ứng dụng đó gửi đi. Thường là để kiểm tra loại dữ liệu được trao đổi giữa ứng dụng di động và các endpoint nó kết nối với. Nó có thể được sử dụng để xác định các lỗi như “identify insecure communication, potential mobile app hoặc server side vulnerabilities, hoặc even insecure mobile app hoặc server-side configurations.

Background

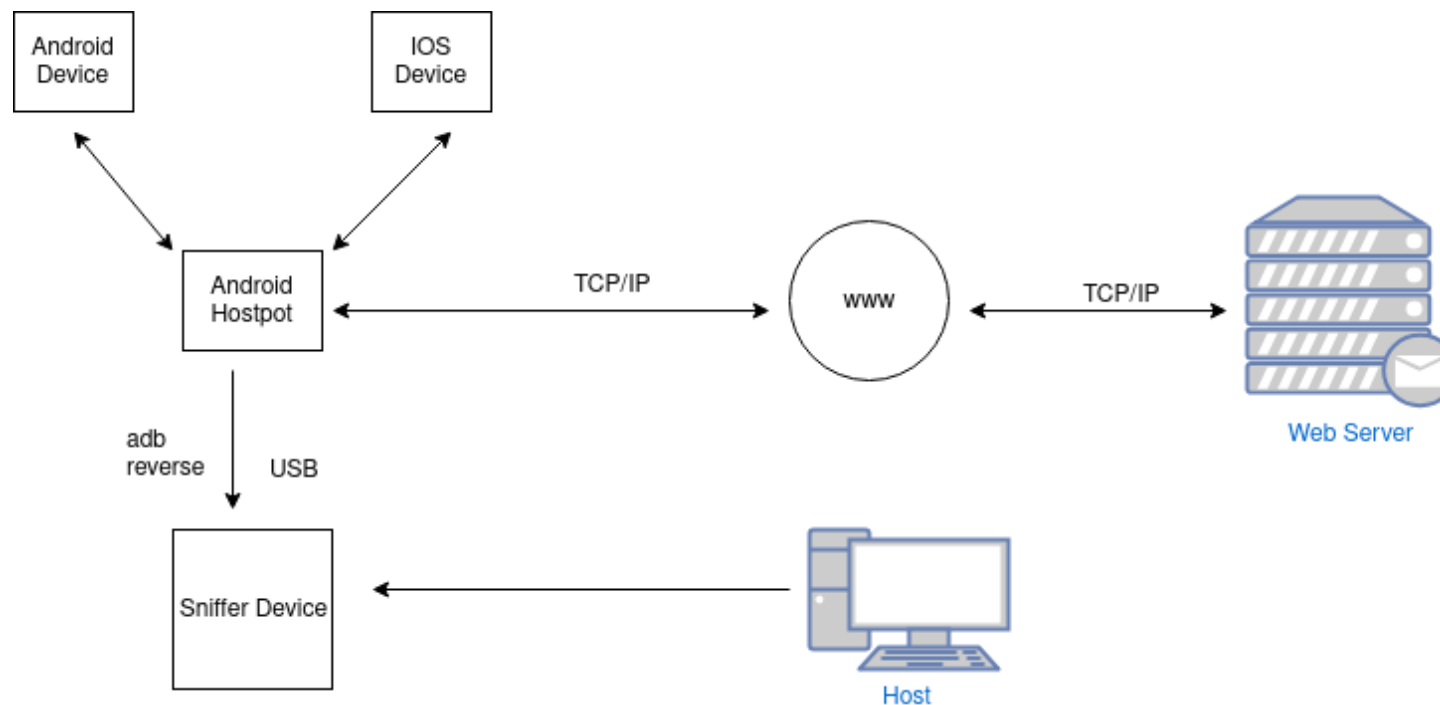
Có hai phương pháp phân tích traffic của mobile app là Passive analysis và Active analysis.

- Passive analysis: traffic sẽ được thu thập vào 1 thời điểm và sẽ phân tích vào thời điểm sau, ví dụ ta sẽ dump các traffic thành các file pcap hay pcapng, rồi phân tích chúng bằng các công cụ chuyên dụng như wireshark chẳng hạn.
- Active analysis: giống như một cuộc tấn công MiTM, ta sẽ chặn bắt các gói tin được gửi đi và sửa lại chúng trước khi request lên server, ví dụ inject các script vào web pages, packet, media hay spoofing, ..etc. Thông thường ta sẽ sử dụng các tools chuyên dụng như [Burpsuite](#), [Charles proxy](#), Bettercap, MiTMf. Riêng mình thích sử dụng [Charles proxy](#) và [Burpsuite](#).



Setup

– Bài này mình sẽ setup theo phương pháp Passive Analysis, mô hình như dưới đây:



Access Point

Mình sẽ sử dụng 1 máy Android để phát wifi, máy này cần được root để có thể thực hiện việc dump network packets từ network interfaces.

Analysis Setup

Việc tiếp theo là setup một máy ảo để thực hiện các công việc phân tích dữ liệu thu thập được, ta có thể dễ dàng setup 1 máy ảo Linux (ubuntu, centos, arch, ..etc) để cài các tools trợ giúp cho công việc phân tích. Hoặc có thể sử dụng luôn **Security Onion**, OS chuyên dụng có sẵn các tools phục vụ công việc network analysis.

Ta sẽ phải setup 3 interface cho máy ảo để thực hiện quá trình phân tích:

- Adapter 1 – Internal network (monitor interface)
- Adapter 2 – Bridged (management interface)
- Adapter 3 – NAT (internet access)

Bây giờ ta phải thiết lập để gửi traffic tới máy ảo này để thực hiện phân tích. Để làm điều này ta sử dụng cáp USB kết nối từ máy android phát wifi với máy ảo phân tích. Để chắc chắn thiết bị sẽ gửi các gói tin tới VM, ta sẽ sử dụng kỹ thuật reverse port forward tới 1 port được chỉ định mà máy Android sẽ gửi gói tin tới, trên VM ta cũng lắng nghe trên port đã được chỉ định.

Kết nối thiết bị Android phát wifi với máy tính và sử dụng lệnh “adb reverse <local> <remote>”, lệnh này ngược với “adb forward”, nó sẽ đảo ngược các kết nối từ device tới host, local tương ứng với kết nối socket trên thiết bị Android, còn remote tương ứng kết nối trên VM. Ví dụ:

```
1 adb reverse tcp:1337 tcp:1337
```

Bên cạnh đó ta cũng còn thiết lập trên VM để lắng nghe các gói tin được gửi tới:

```
1 nc -l -v -s 127.0.0.1 -p 1337 | tshark -i -
```

Sử dụng kèm theo lệnh “**tshark**” để test xem gói tin đã được gửi tới hay chưa.

Ngoài ra có thể sử dụng “**tcpreplay**” để có thể replay lại gói tin khi cần thiết, hoặc ta có thể ném gói tin replay vào BroIDS để phân tích chúng.

```
1 nc -l -v -s 127.0.0.1 -p 1337 | tcpreplay -i eth0 -
```

Dump Packet

– Ta sẽ có 2 options để lựa chọn cho công việc này

OPTIONS 1:

Sử dụng điện thoại để dump trực tiếp packet về rồi phân tích, ta sẽ có app **Packet Capture** phục vụ cho công việc này. Ứng dụng này tạo 1 điểm VPN để thu thập toàn bộ dữ liệu đi qua nó, ta có thể dump cho 1 hoặc nhiều ứng dụng mà ta muốn.

Cái này sẽ có 1 vài hạn chế nhất định, mình sẽ hướng dẫn cách fix vào các bài sau.

OPTIONS 2:

Phương thức này sẽ có nhiều ưu điểm hơn so với options 1, ta sẽ dump packet tại access point, các ứng dụng vẫn sẽ request tới các endpoint và hoạt động bình thường, ta sẽ chỉ quan tâm các gói tin đi qua access point, phương thức này sẽ khắc phục được các hạn chế mà options 1 mắc phải nhưng mà ta sẽ vướng phải issue là toàn bộ traffic hoạt động của device sẽ đi qua access point chứ không riêng rẽ 1 app nào.

Để khắc phục điều này ta sẽ phải mất 1 chút thời gian để phân tích tìm ra các endpoint và tập trung khai thác mà app request tới.

Dưới đây là hướng dẫn khi sử dụng 1 máy android làm Wifi Hostspot.

- Access shell sử dụng **android debug bridge(adb)** và tìm network interface được sử dụng cho Wifi Hostspot (sử dụng command “**netcfg**” hoặc “**ifconfig**”, interfaces thường sẽ là **ap0** hoặc **wlan0**)
- Cài đặt **tcpdump** (**/data/local/tmp**) và **busybox** lên máy
 - adb push tcpdump /data/local/tmp
 - adb shell “chmod 777 /data/local/tmp/tcpdump
 - adb push busybox-armvxxx /data/local/tmp

- adb shell “chmod 777 /data/local/tmp/busybox-armvxx
- Sau khi thực hiện các bước trên thì ta đã có thể thực hiện công việc dump packets.

- adb shell
- cd /data/local/tmp/
- su
- ./tcpdump -i wlan0 -s0 -w - | ./busybox-armvxx nc 127.0.0.1 1337

Nếu install busybox từ playstore thì sử dụng command:

- ./tcpdump -i wlan0 -s0 -w - | nc 127.0.0.1 1337

Nếu không xảy ra lỗi thì trên máy ta sẽ thu được toàn bộ packet mà ứng dụng và máy đang hoạt động.



Aishee



Previous Post:

Web Application Penetration Testing Course

Next Post:

Wakanda: 1 — VulnHub



Related Posts:

Mirai botnet Tut 2: Bruteforce and DDoS Attack

[FUN] Bypass XSS Detection WAF

Machine learning for Web Application Firewall (WAF)

Leave a reply:

Your email address will not be published.

Post Comment

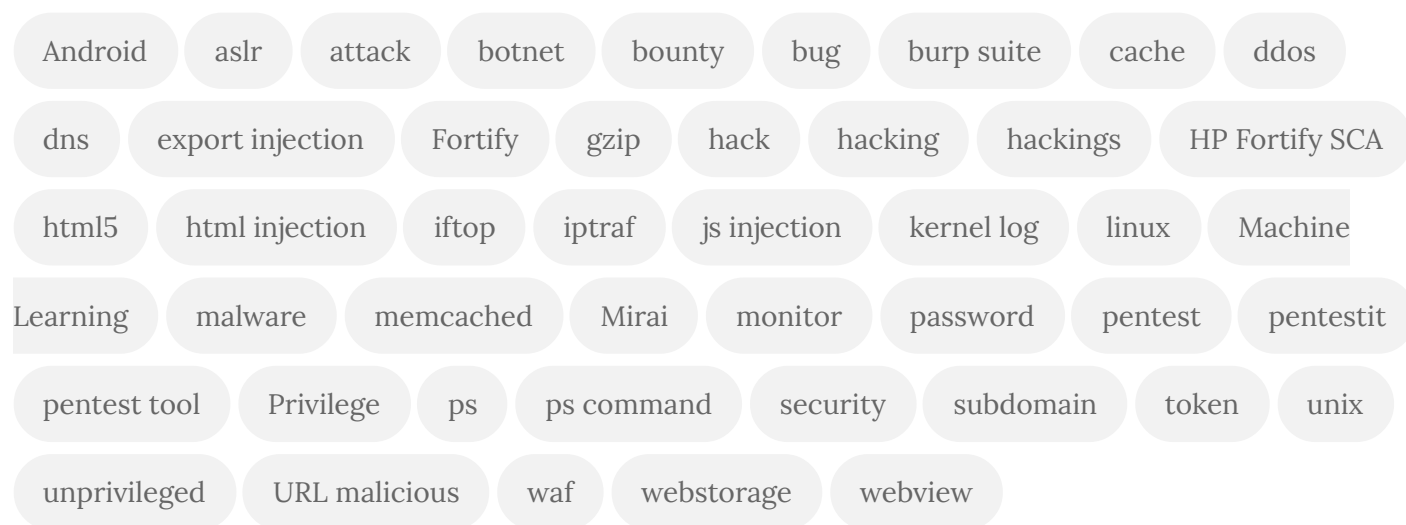
This site uses Akismet to reduce spam. [Learn how your comment data is processed.](#)

About Me



My name is Nguyen Anh Tai. I am an independent security researcher, bug hunter and leader a security team. Security Researcher at CMC INFOSEC. I developed the every system for fun :D. My aim is to become an expert in security and xxx!

Tags



Tweets

 **Aishee** RT @cyb3rops: Oh my gosh, this is the noisiest RAT that I've ever seen. Drops .js, .ps1, .bat, .exe, .exe as .jpg, ssh service, TeamViewer,...


about 18 days ago



 **Aishee** RT @axi0mX: EPIC JAILBREAK: Introducing checkm8 (read "checkmate"), a permanent unpatchable bootrom exploit for hundreds of millions of iOS...

about 19 days ago



 **Aishee** 7% through "The Simulation Hypothesis: An MIT..." by Rizwan Virk. Try the book for free:
<https://t.co/iaDzmZri9Y> <https://t.co/oq5dvJoUyr>

about 4 months ago



Make by Aishee - A blog simple for social