

الفصل الثامن عشر

ثغرة ال Session Hijacking

المؤلف

د.م/ أحمد هاشم الفقي

استشاري أمن المعلومات و التحول الرقمي

Ahmed.Hashem.ElFiky@outlook.com

ثغرة ال Session Hijacking

- اقتحام الجلسات هي عملية السيطرة على جلسة المستخدم Session الذي يقوم بإستخدام النظام. عملية إقتحام الجلسة تلزم ان يقوم المخترق بالتقاط رقم الجلسة Session ID او توليد إجباري لها Brute Force او إعادة توليد للرقم Reverse Engineering, قد يبدو المفهوم صعب لذا سأسترسل بشرحه أكثر.
- من المعروف ان هناك نوعين من الجلسات ,الجلسات الدائمة Persistent وهي التي يتم من احلها تعريف ملفات الارتباط اي Cookies وحفظها في جهاز المستخدم لكي يتعرف عليه النظام عند عودته في اي وقت مرة أخرى.



ثغرة ال Session Hijacking (تكملة...)

- النوع الثاني هو الجلسات الغير الدائمة non-Persistent وهي التي تنتهي بمجرد إغلاق المستخدم للمتصفح, في كلا النوعين يتم تعريف رقم جلسة Session ID للمستخدم, رقم الجلسة هذا يستخدم لمعرفة متغيرات المستخدم الذي يرسلها او يستقبلها خلال جلسته على نظام , هذا الرقم ينشئ عادة بشكل إفتراضي من لغة البرمجة التي تستخدمها من خلال رقم أي بي IP المستخدم وقت الجلسة يدمج معها بعض المتغيرات الأخرى.
- بعض المبرمجين يكتفي بتوليد هذا الرقم بشكل إفتراضي دون ان يسعى لتشفيره او إضافة المزيد من العوامل عليه لجعل عملية التوليد الاجبارية او إعادة التوليد له تكون صعبة, وهنا تكمن المشكلة حيث يقوم المخترق بمحاولة توليد رقم الجلسة بمعرفة بعض المعطيات اللحظية و يرسلها عن الطريق HTTP Request الى نظام الذي يقرأ رقم الجلسة ويقارنه برقم الجلسة الموجود لديه في ذاكرة , فإذا تطابق, فهذا يعني من وجهة نظر النظام ان المخترق هو المستخدم الحقيقي.

ثغرة ال Session Hijacking (تكمله...)

- ويمنحه بذلك حق الوصول لمنطقة المستخدم الخاصة اي حسابه البنكي على سبيل المثال , الجدير بالذكر ان هجمات ال XSS يمكن ان تستخدم للإستلاء على الجلسات وذلك عن طريق تمرير كود جافا سكريبت للنظام يقوم بقراءة رقم الجلسة المستخدم وإرسال هذا الرقم للمخترق...
- الحماية من هذه الثغرة
- حاول تشفير رقم الجلسة وتعقيدها قدر المستطاع.
- إستخدام ال SSL لتشفير البيانات الحساسة المرسله والمستقبله من و الى نظامك.
- برمجيا قم بإنهاء اي جلسته يمضي عليها وقت كافي تقدر بأن المستخدم خلالها قد انتهى فعلا من عمله خلالها او انه قد ترك شاشة النظام مفتوحة ولم يعد يستخدمها.
- حصن نظامك ضد هجمات ال XSS

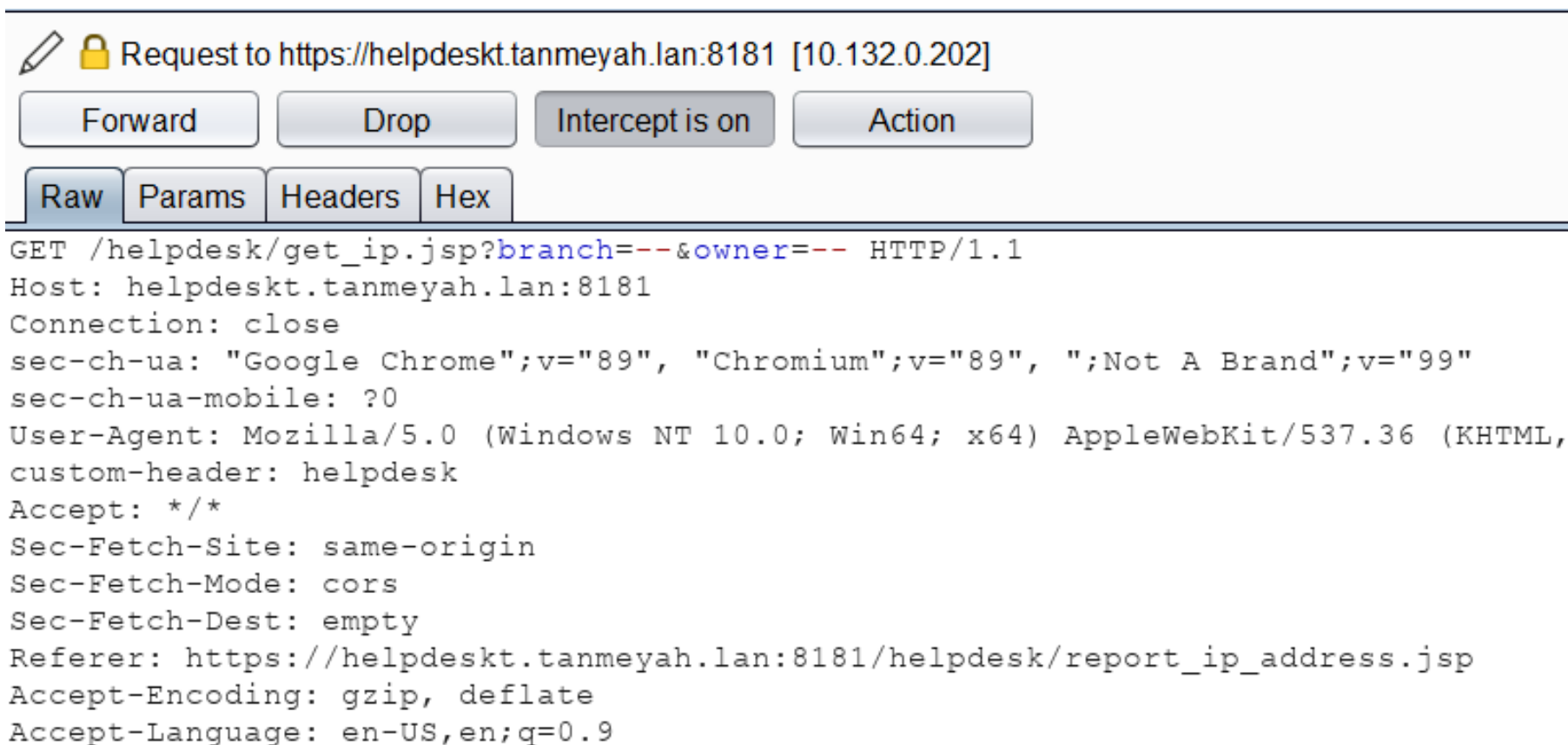
إداة BurpSuite Sequencer لتنبؤ ال Session ID

  Request to <https://helpdeskt.tanmeyah.lan:8181> [10.132.0.202]

GET /helpdesk/get_ip.jsp?branch=--&owner=-- HTTP/1.1
Host: helpdeskt.tanmeyah.lan:8181
Connection: close
sec-ch-ua: "Google Chrome";v="89", "Chromium";v="89", ";Not A Brand";v="99"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, custom-header: helpdesk
Accept: */*
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://helpdeskt.tanmeyah.lan:8181/helpdesk/report_ip_address.jsp
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: JSESSIONID=4772a365178cd5f874eb660bb528;

إداة BurpSuite Sequencer لتنبؤ ال Session ID

- نقوم بحذف ال Cookie من ال request ثم ارسال request الى Sequencer



إداة BurpSuite Sequencer للتنبؤ ال Session ID

The screenshot displays the Burp Suite Sequencer interface. At the top, a request to `https://helpdeskt.tanmeyah.lan:8181 [10.132.0.202]` is shown. Below this are buttons for `Forward`, `Drop`, `Intercept is on`, and `Action`. A tabbed interface below these buttons shows `Raw`, `Params`, `Headers`, and `Hex`. The `Raw` tab is active, displaying the raw HTTP request:

```
GET /helpdesk/get_ip.jsp?branch=--&owner=-- HTTP/1.1
Host: helpdeskt.tanmeyah.lan:8181
Connection: close
sec-ch-ua: "Google Chrome";
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/109.0.0.0 Safari/537.36
custom-header: helpdesk
Accept: */*
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://helpdeskt.tanmeyah.lan:8181/helpdesk/address.jsp
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
```

A context menu is open over the request, listing various actions:

- Scan [Pro version only]
- Send to Intruder Ctrl+I
- Send to Repeater Ctrl+R
- Send to Sequencer**
- Send to Comparer
- Send to Decoder
- Request in browser
- Engagement tools [Pro version only]
- Change request method
- Change body encoding
- Copy URL

إداة BurpSuite Sequencer لتتبع ال Session ID

Live capture

Manual load

Analysis options

?

Select Live Capture Request

Send requests here from other tools to configure a live capture. Select the request to use, configure the other options.

Remove

Clear

#	Host	Request
4	https://helpdesk.tanmeyah.lan:8181	GET /helpdesk/get_ip.jsp?branch=--&owner=

Start live capture

?

Token Location Within Response

Select the location in the response where the token appears.

☒ Cookie:

JSESSIONID=4b990d9424a4fbcaa169 ...

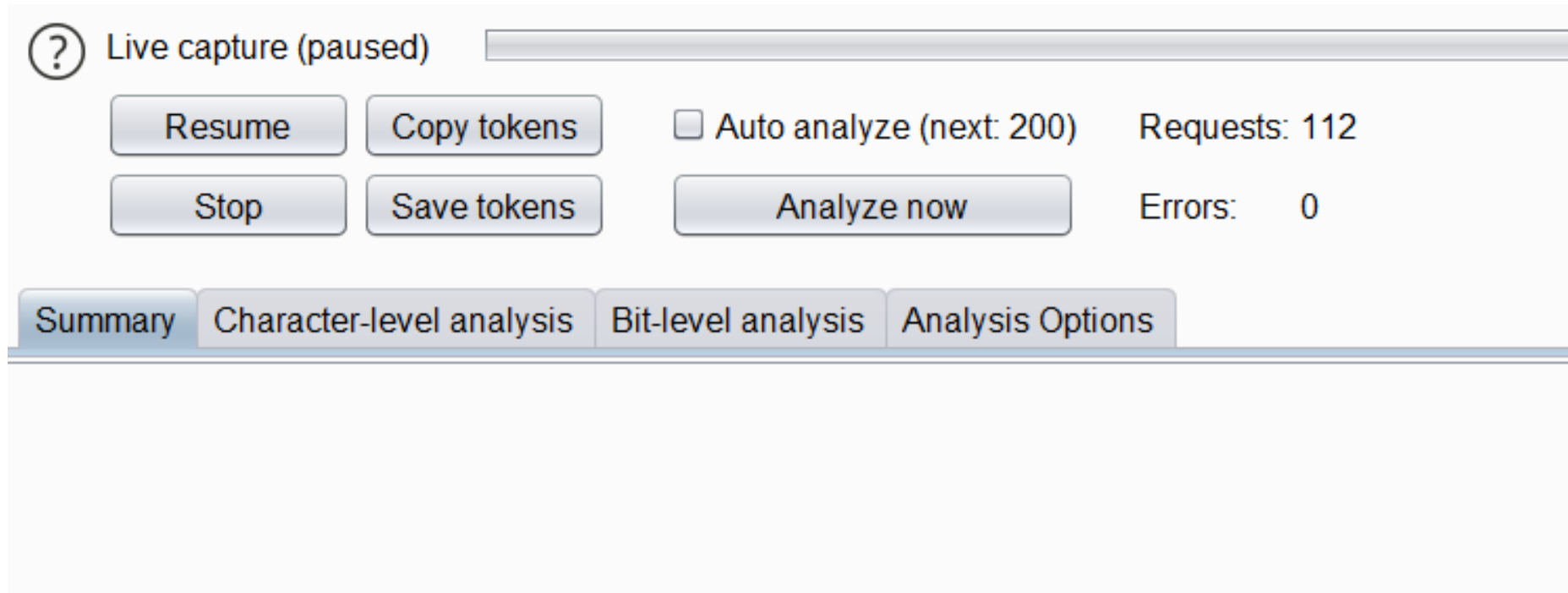
☐ Form field:

☐ Custom location:

Configure

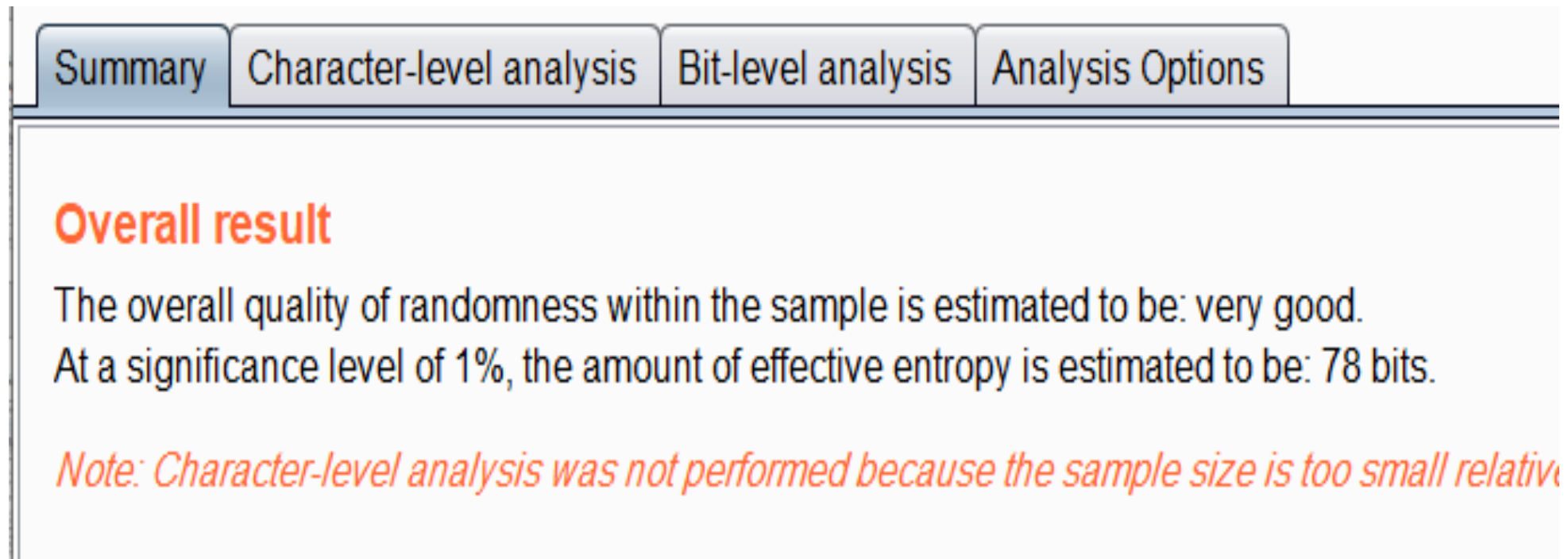
إداة BurpSuite Sequencer لتنبؤ ال Session ID

- ثم الضغط على زر ال Live Capture للتلقاط ال Cookies حيث بعد ما يلتقط اكثر من 100 قيمة لل Cookie نقوم بالضغط على زر Analyze now لمعرفة إذا كان بإمكاننا تنبؤ قيمة ال Cookie او لا



إداة BurpSuite Sequencer لتنبؤ ال Session ID

- حيث فى مثالنا هذا قيمة ال Cookie صعب ان تنبؤ لان Randomness = Very Good



The screenshot shows the Burp Suite Sequencer interface with the 'Summary' tab selected. The 'Overall result' section states that the overall quality of randomness is 'very good' and that at a 1% significance level, the effective entropy is estimated to be 78 bits. A note indicates that character-level analysis was not performed due to a small sample size.

Summary Character-level analysis Bit-level analysis Analysis Options

Overall result

The overall quality of randomness within the sample is estimated to be: very good.
At a significance level of 1%, the amount of effective entropy is estimated to be: 78 bits.

Note: Character-level analysis was not performed because the sample size is too small relative

تم بحمد الله انتهاء الفصل الثامن عشر