Web Application Security

Use Burp to analyse the randomness of Session-IDs in DVWA

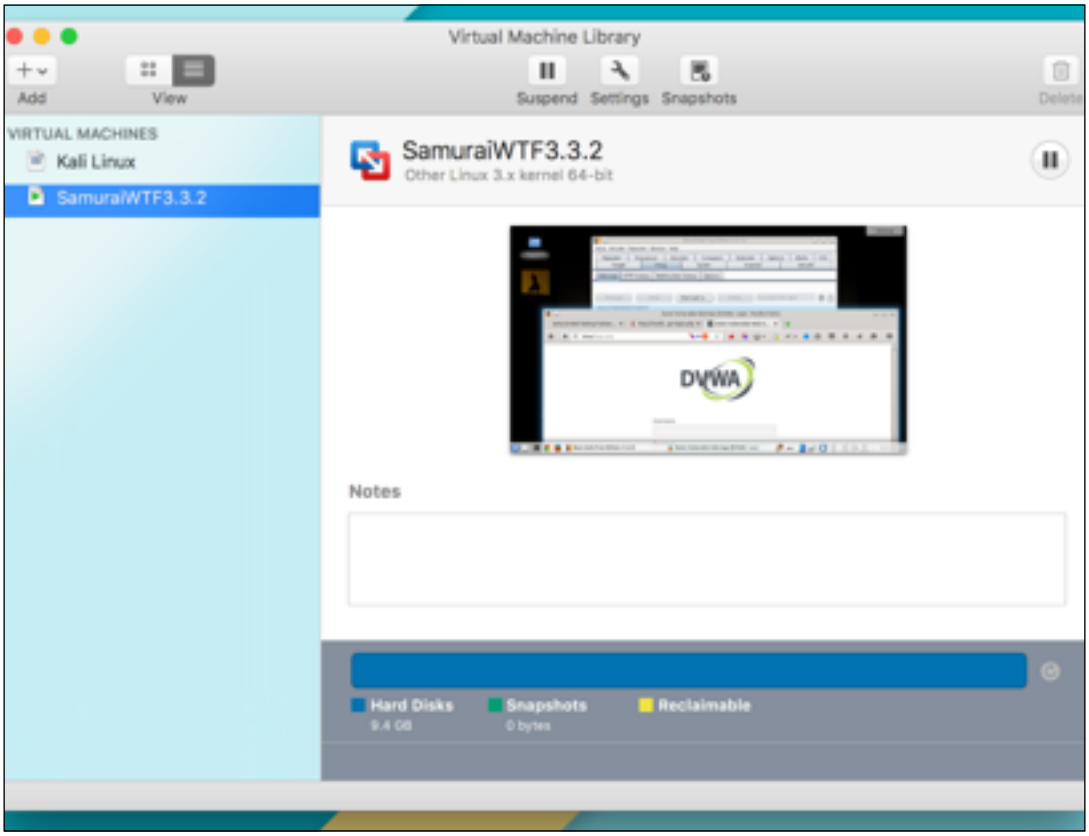
Robert James Gabriel
Part 4 - 13 April 2016

Setup

Open Your Vm Fusion

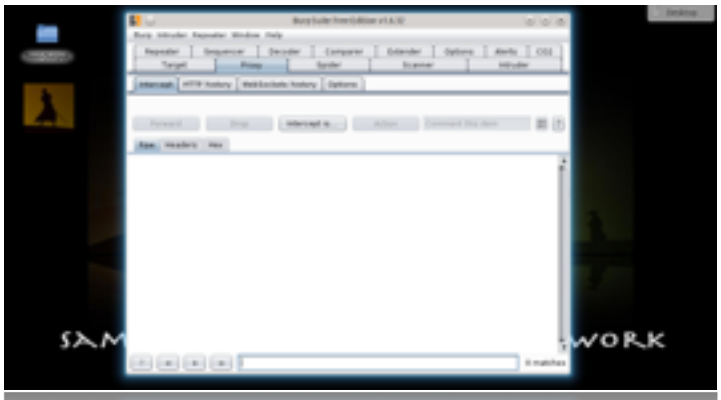
Instructions:

1. On Your Host Computer, Go To
2. Applications --> All Program --> VM Fusion
3. Select Samjuri.wtf and Boot the Vm



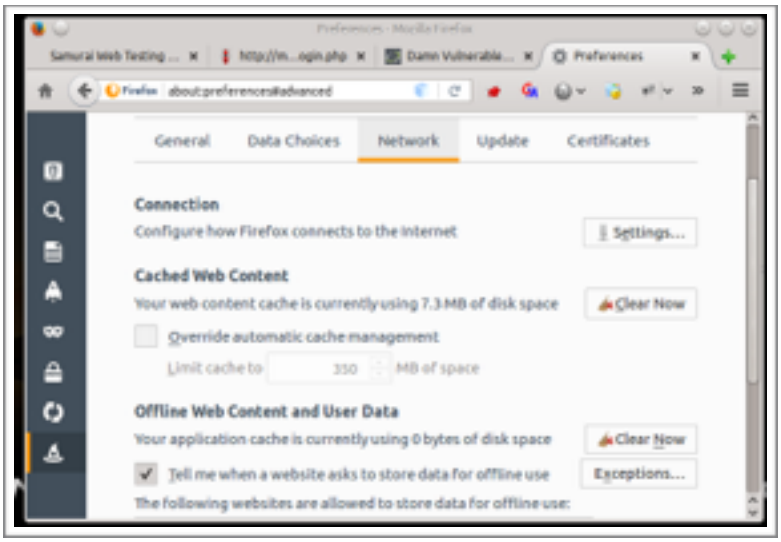
Open the following

1. **Burp Free Suit**
2. **Firefox**

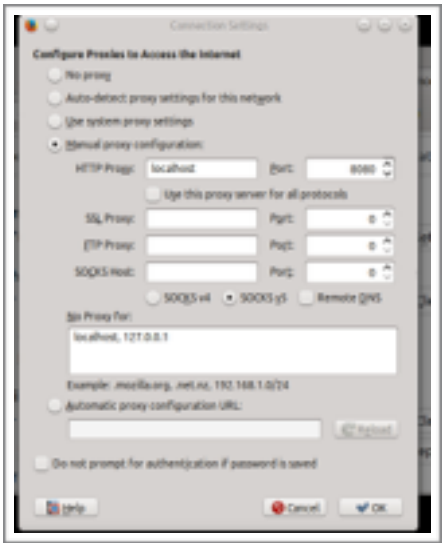


Firefox Setup

1. Switch to Firefox
2. Go to options
3. Click
    1. Advanced
    2. Network tab
    3. Settings

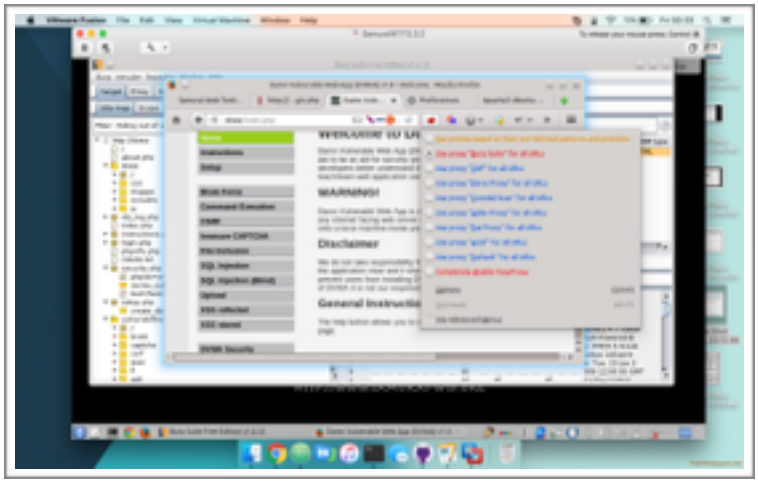


4. When opened, check Manual Proxy Configuration.
5. Make sure the Http Proxy is **localhost**
6. Make sure the Port is **8080**



Or

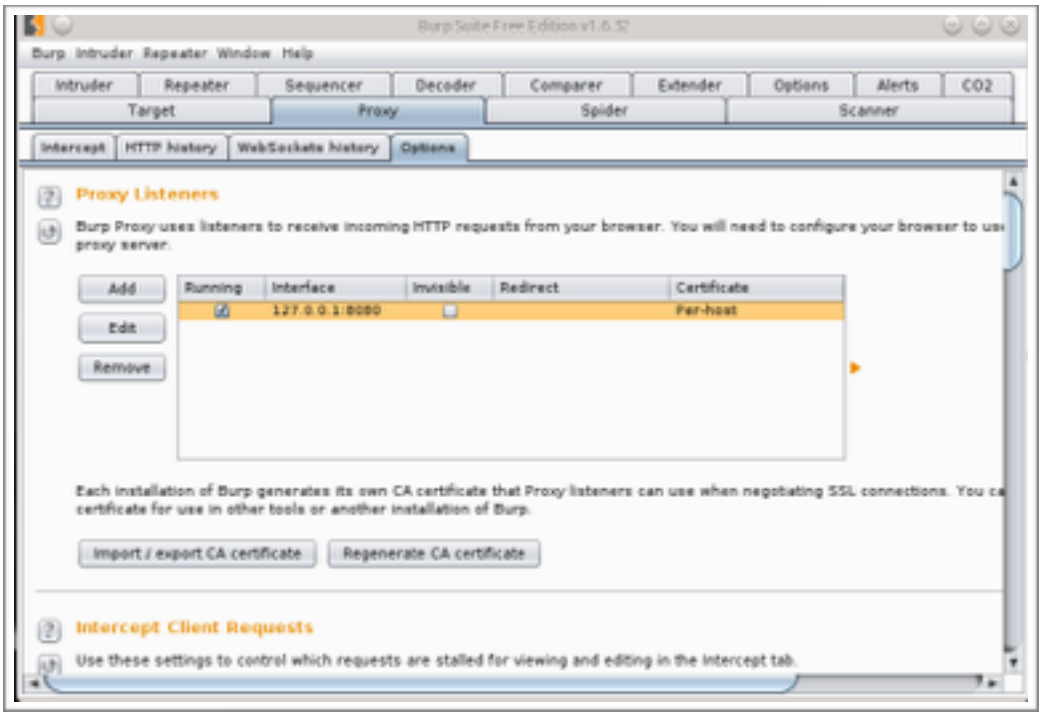
1. Click on foxy Proxy Add-on
2. Click Use “Burp settings”



Visit **http://mutillidae/index.php?page=dns-lookup.php**

Burp Setup

1. Switch Burp suite
2. Make sure you listening to **localhost:8080**
3. To check this, open Burp Suite and click Proxy -> Options

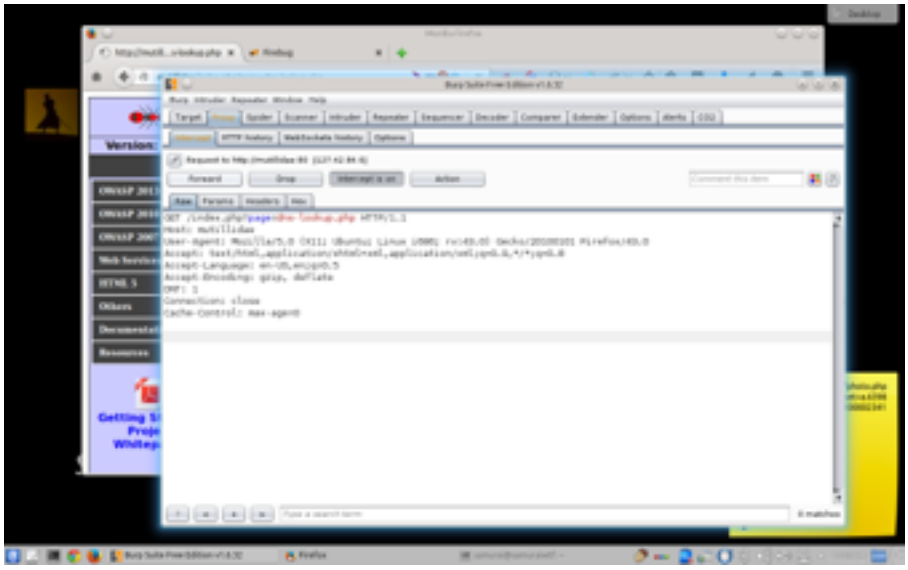


4. Next select Proxy and turn on intercert

Step One

1. Visit http://mutillidae/index.php?page=dns-lookup.php
2. Clear all cookies but visiting -> settings-privacy->cookies->remove all.
3. Make sure that the inceptor is on.
4. Refresh the page.
5. Go to burp and look at the inceptor.

As you can see there is no cookies presented in the header.



1. Forward the response
2. There should be a cookie in the response.

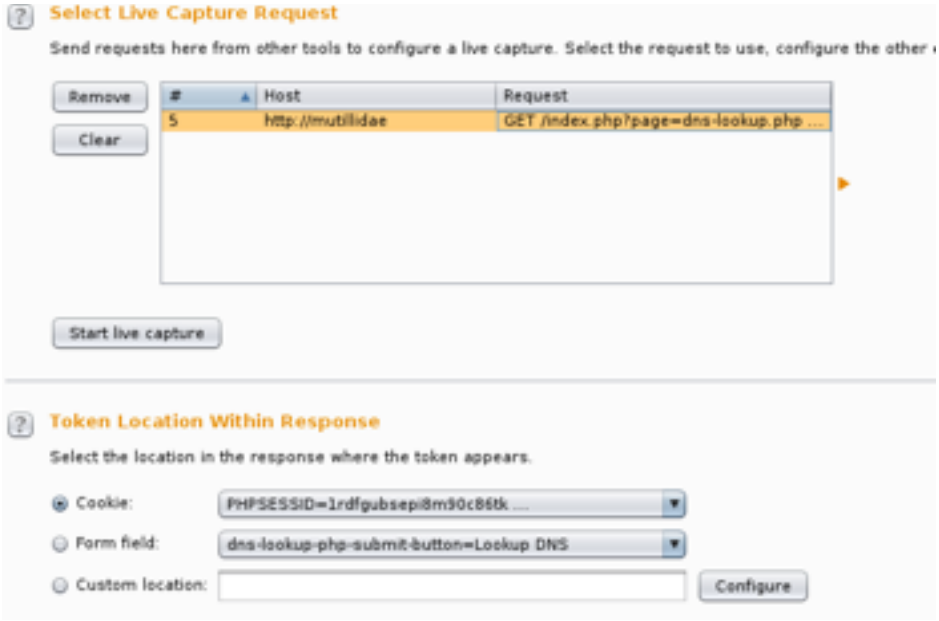
Next refresh the page, but this time delete the cookie from the request.

It will create another cookie session.

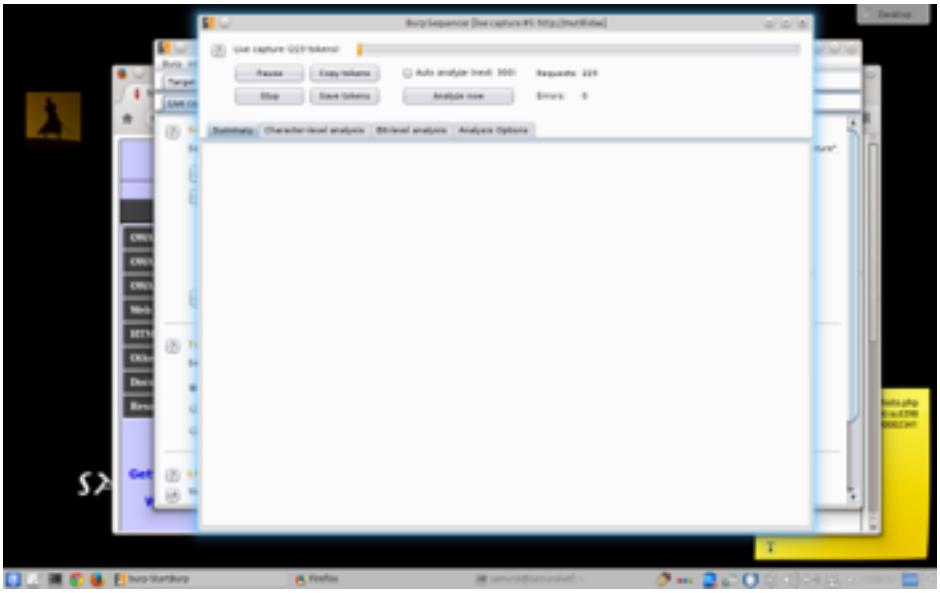
So now were going to refresh the page again, go to burp again and delete the cookie

then right click and click send to sequencer and then forward

As you can see it has already selected the php session cookie, Its important that we can now use this information with a live session information to base it off to get other session ids.



Click start capture

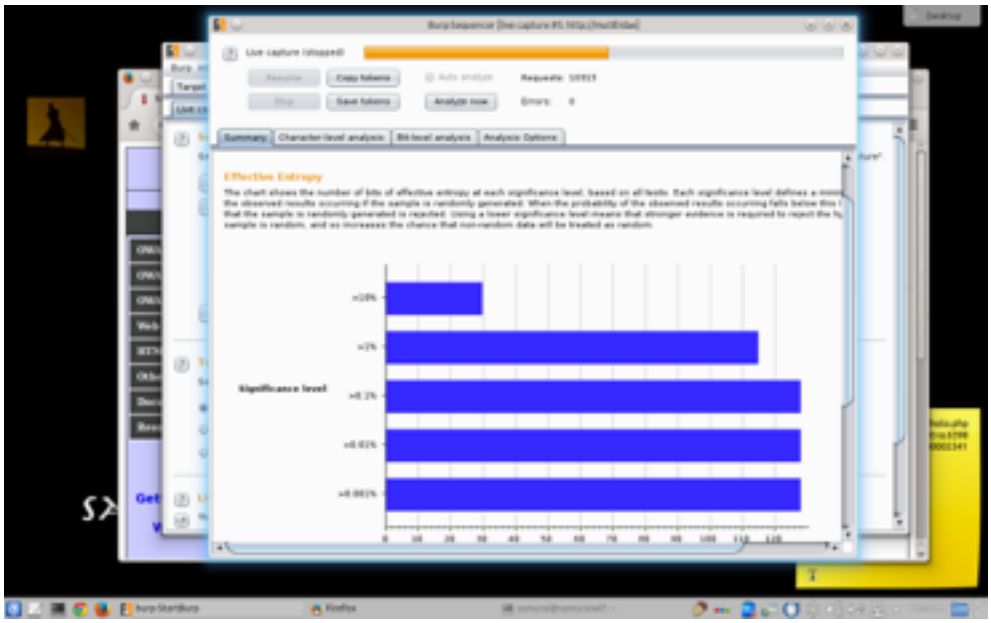


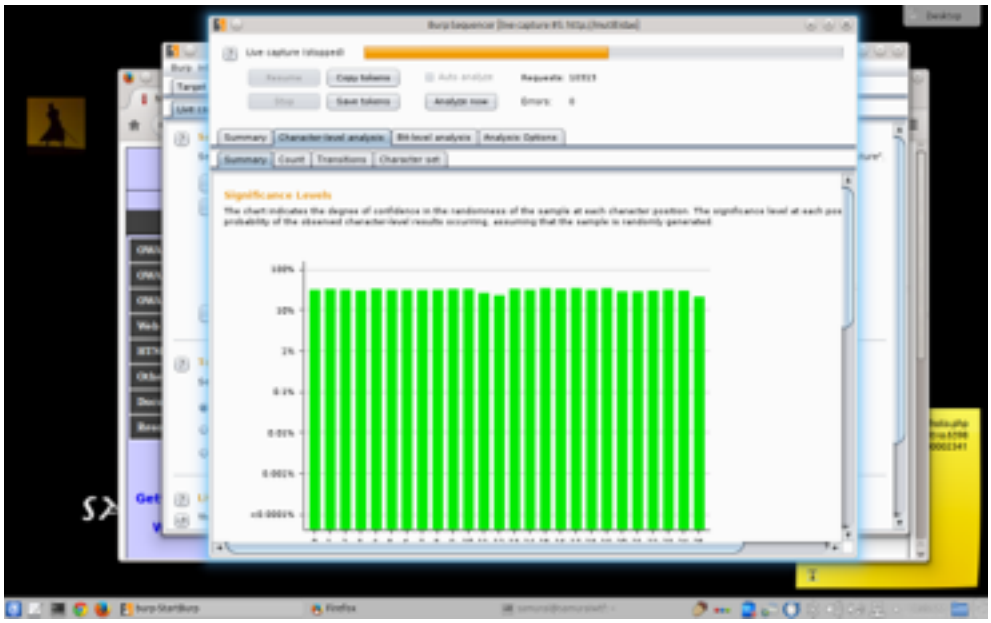
You need a few tokens, I stopped it at 10313.

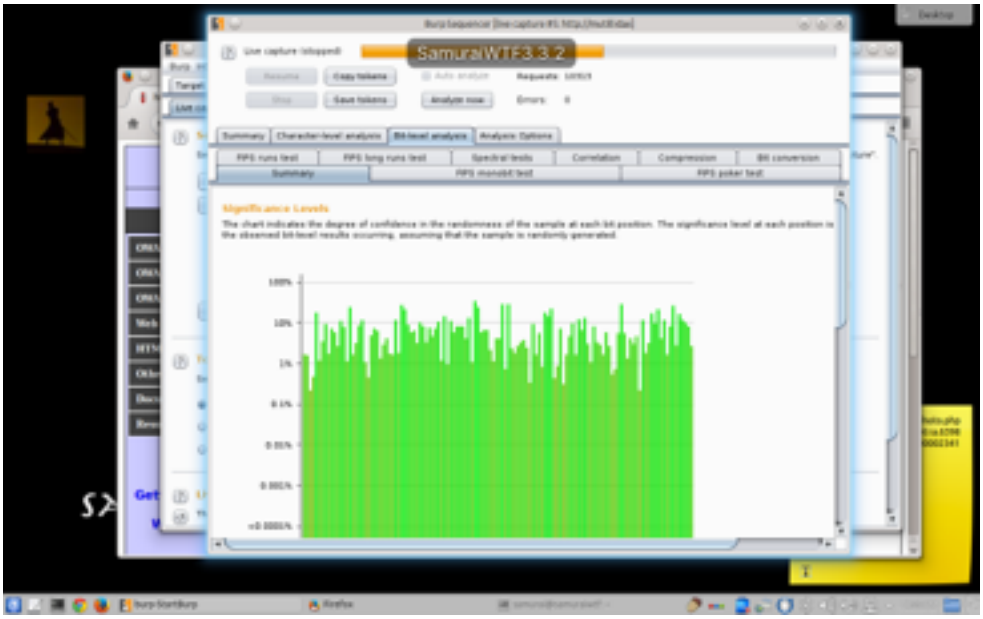
Then click analyse now.

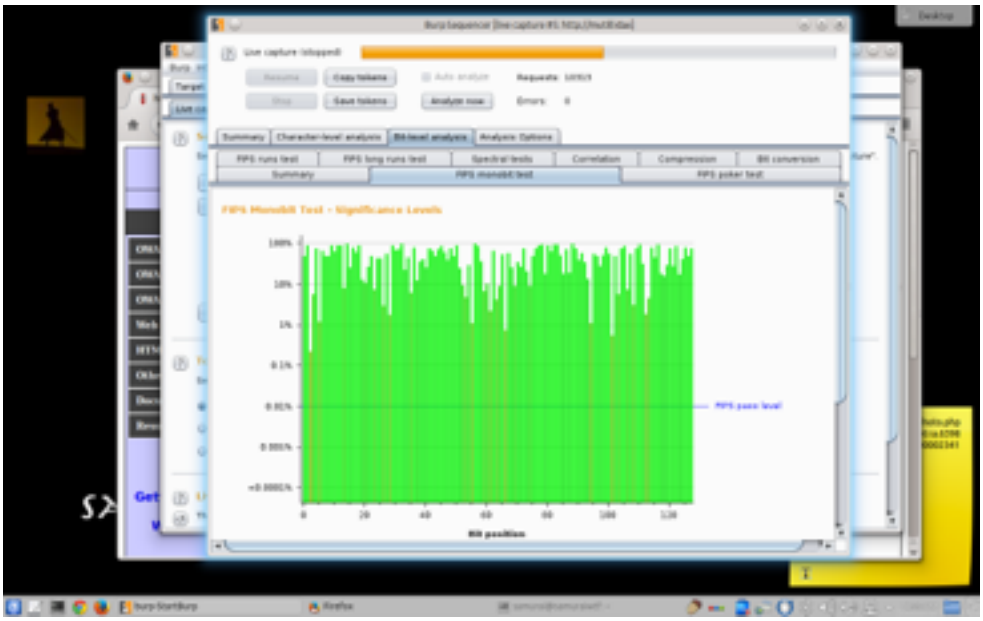
Will display the following information on likely hood of each coming up.

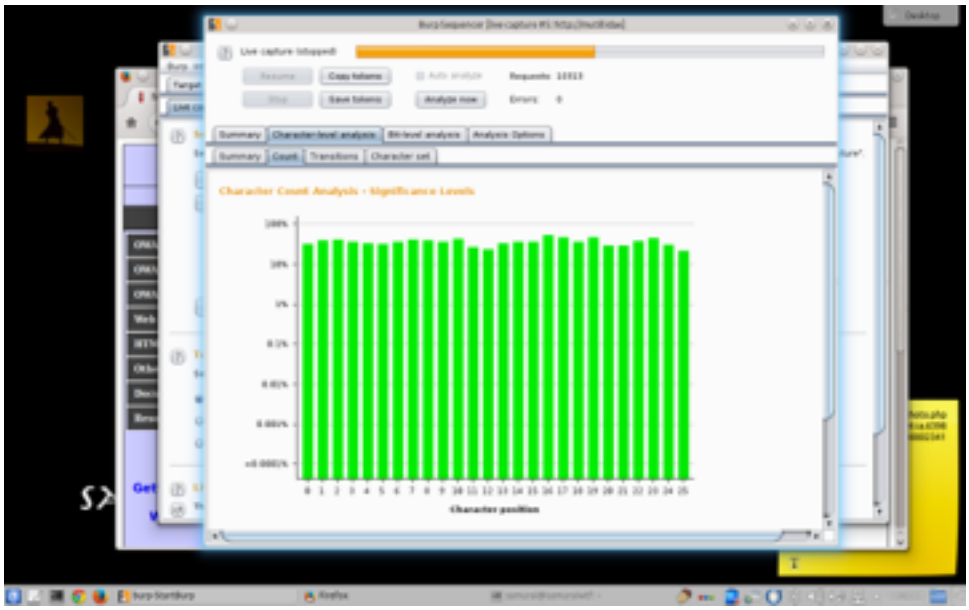
The information gathered was





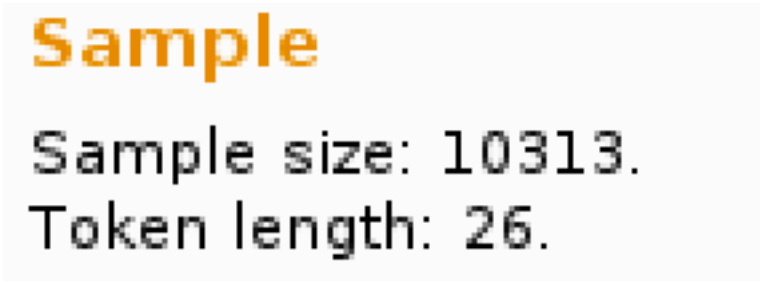

SamuraiWTF3.3.2



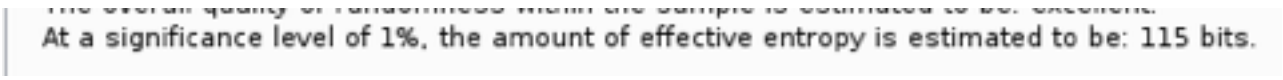


So from the last few graphs we can see that frequency of each character on not he session id.

So a token length produced in my case



produced the



Other highlights include the fact that information the characters are quick random with some being slightly more likely to be used. But nothing user noticeable.