# C. Analytics Development

## 1. Design an algorithm to shortlist IPs that could be running reconnaissance activities against an enterprise web server. State any assumptions you make in your design.

Step 1: Researched on the format of the dataset[[1]](#footnote-1).

# of Instance: 2048442; # of Attributes: 27

Step 2: Researched the types of reconnaissance activities.

Reconnaissance activities[[2]](#footnote-2) (any form of activity that tries to get information from the server)

1. Finding anomalies in user agent strings
   1. Multiple user agents (Windows 🡪 Intel Mac)
   2. Scanning tools (active & passive footprinting: Nikto[[3]](#footnote-3), Nessus, Googlebot, Nmap[[4]](#footnote-4), Wget[[5]](#footnote-5),[[6]](#footnote-6))
   3. Passing command through user agent ($\_POST[[7]](#footnote-7))
   4. No user agent?[[8]](#footnote-8)
2. IP with numerous failures (using status code)
   1. 400 Bad Request, 404 Not Found
   2. No status code
3. Injections/ Path transversal (using uri/ username)
   1. SQL injections (OR, AND, UNION, SELECT, INSERT, DELETE)
   2. XSS injections (<script>, <img>)
   3. Path traversal (../, \.., /etc/passwd, /shadow[[9]](#footnote-9))

Step 3: Designing algorithm (coding)

1. Enumerate through the http.log file.
   1. Extract information for recon activities
   2. For user agent and status code, store in dictionary
   3. For injections and path transversal, store in array
2. Loop through stored information in dictionary to identify recon activities
   1. Check if they match the cases identified above in step 2.
      1. For IP with numerous failures, users can set their own thresholds.
   2. Store in array if they match
3. Store all unique IP found that could be running recon activities in the output file.

Assumptions:

1. I assume that each user IP will only access the web server from one browser, hence having multiple user agent is flagged out.
2. I assume that having numerous failures is a reflection of brute forcing.
3. Simply the checking algorithm for status code by condensing the status code from server to checking only the first digit (1,2,3,4,5).
4. I assume any form of scripting, sql injection and path traversal is malicious

Improvements to be made:

Instead of comparing each http request with the originator IP and the individual attributes, I could make comparisons with each line as a whole. However, I will have to figure out the relationship between other attributes (E.g. Uri with method and status code).

1. <https://www.secrepo.com/Datasets%20Description/Network/http.html> [↑](#footnote-ref-1)
2. <https://www.sans.org/reading-room/whitepapers/detection/paper/34067> [↑](#footnote-ref-2)
3. <https://www.sevenlayers.com/index.php/57-nikto-user-agent-change> [↑](#footnote-ref-3)
4. <https://null-byte.wonderhowto.com/how-to/hack-like-pro-using-nmap-scripting-engine-nse-for-reconnaissance-0158681/> [↑](#footnote-ref-4)
5. <https://security.stackexchange.com/questions/174805/wget-what-security-issues-am-i-not-considering> [↑](#footnote-ref-5)
6. <https://user-agents.net/string/wget-1-10-2-red-hat-modified> [↑](#footnote-ref-6)
7. <https://www.acunetix.com/blog/articles/keeping-web-shells-undercover-an-introduction-to-web-shells-part-3/> [↑](#footnote-ref-7)
8. <https://webmasters.stackexchange.com/questions/35642/interpretation-of-empty-user-agent> [↑](#footnote-ref-8)
9. <https://www.cyberciti.biz/faq/understanding-etcshadow-file/> [↑](#footnote-ref-9)