In this project author has given concept to detect normal or communal tweets but not given any concept to detect malicious users and most of the time malicious users are only responsible to spread communal tweets and in extension i added concept to detect such malicious users. In almost all tweets users give URL link to post videos or any other greeting text. URLS will contain a long text but tweeter can accept only 140 characters and to avoid such problem twitter has introduced SHORTEN URL concept where big URL will be map to shorten URL. Shorten URL can give in tweets and when user click on such URL then twitter automatically obtained big URL mapping from Shorten URL.

Example

Stack over flow shorten URL

Short URL = http://s.tk/

When u paste above short URL in browser and then press enter key then automatically that URL changes to big URL as below one

Big URL = <https://stackexchange.com/>

Malicious users may take advantage of such technique to spread tweets with such Shorten URL which will redirect user to malicious websites upon click by user on such URLS. Malicious Web site then steals information from user system and sent to malicious users.

Always malicious users will have only one or few websites and they create lakhs of shorten URL which map to such few malicious web sites. Upon user click on such URLs users may be redirected to such web sites.

To detect such malicious link twitter is already using black listed URLS but its not sufficient to detect different shorten URLS. To overcome from this problem we can analyze all URLS to check whether they are redirecting to same website or not. If multiple Shorten URLS redirecting users to same website then we can make such URL as malicious and we don’t require maintaining any black listed URL database. Above work we are using as Extension Concept.

Below is the code to get expand URL from short URL

public static String expandUrl(String shortenedUrl) throws IOException {

URL url = new URL(shortenedUrl);

// open connection

HttpURLConnection httpURLConnection = (HttpURLConnection) url.openConnection(Proxy.NO\_PROXY);

// stop following browser redirect

httpURLConnection.setInstanceFollowRedirects(false);

// extract location header containing the actual destination URL

String expandedURL = httpURLConnection.getHeaderField("Location");

httpURLConnection.disconnect();

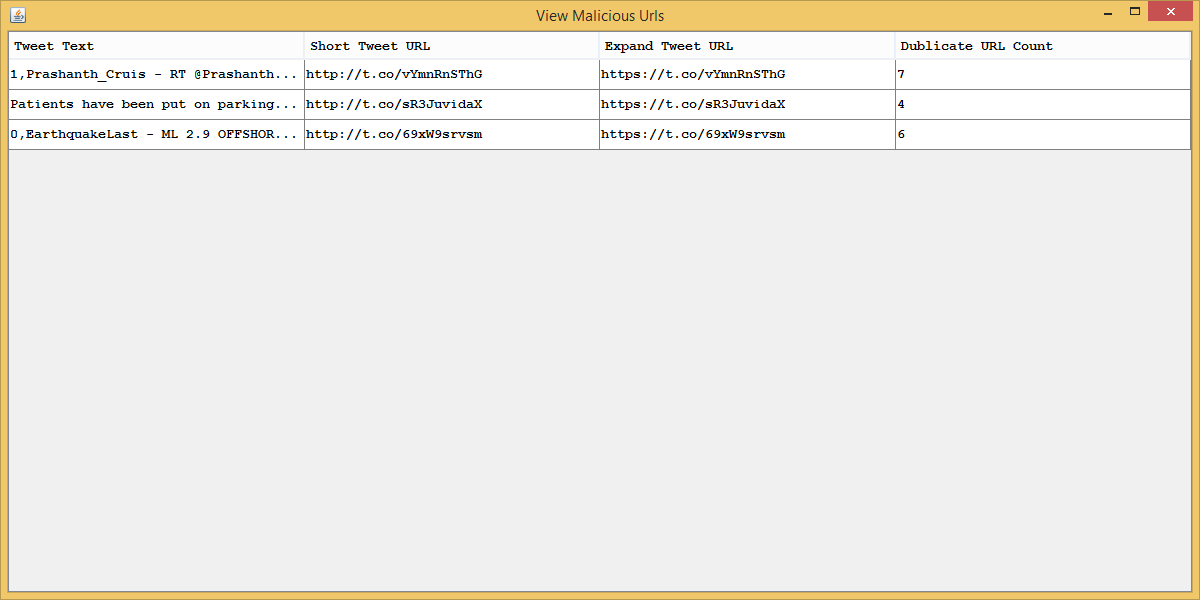
return expandedURL;

}

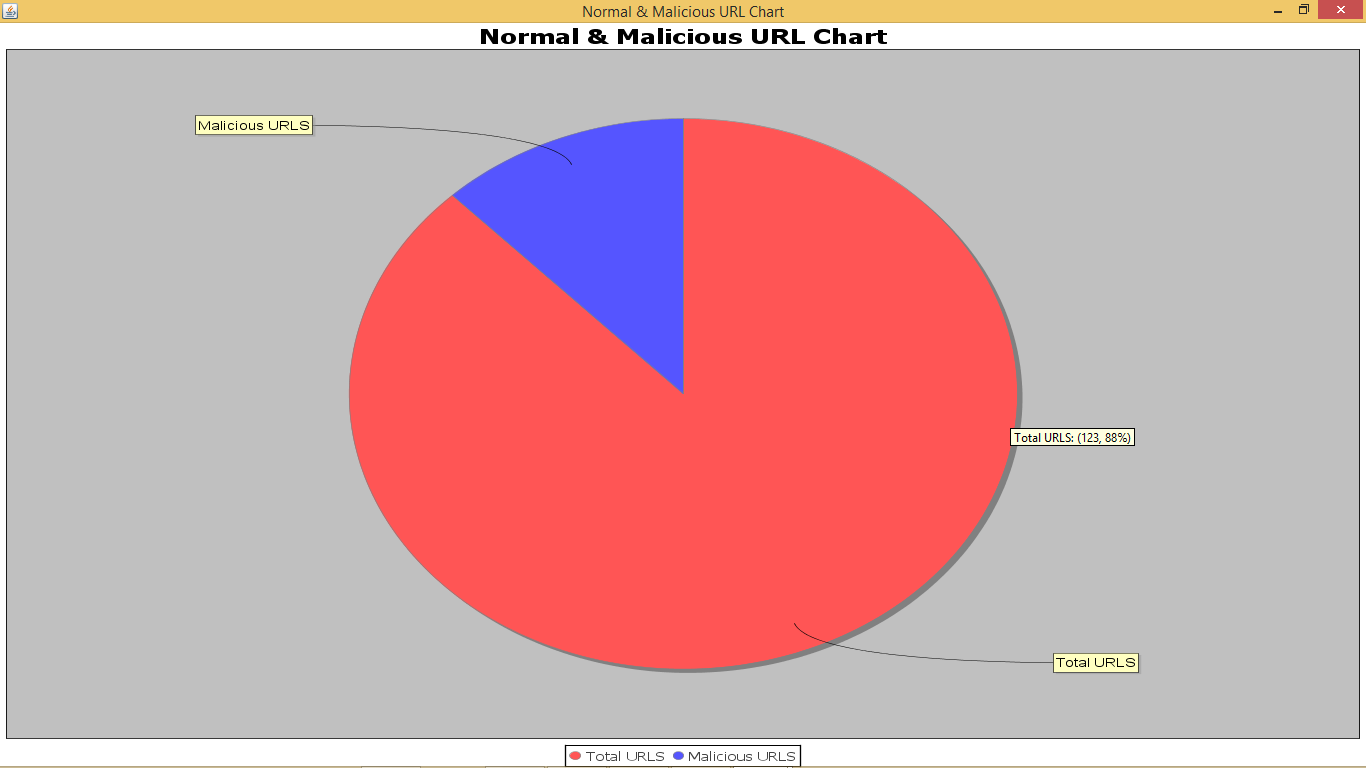
Screen shots



In above screen run project according to previous screen shots upto 5th button. Now click on 6th button to detect malicious users



In above screen we are showing detected malicious URL with short and long URL and showing no of duplicate count of URLS malicious users used in various tweets. Now click on ‘Normal & Malicious URL Chart’ button to get normal and malicious tweets



In above graph total URLS are 140 and 123 are normal URLS and 17 are malicious URLS