ATAQUES DE INFERENCIA DE ENTRADAS EN NAVEGADORES WEB MÓVILES BASADO EN SENSORES Y DEFENSAS

BRANDON MEZA

SE PROPONE UN FRAMEWORK DE INFERENCIA QUE CONSTA DE 6 FASES

Segment-SensorData-With-KeyEvents (T) // Used in the training phase
1 W = Identify-Keystroke-TimeWindows (T)
2 W = Adjust-Keystroke-TimeWindows (W)
3 **return** W

Segment-SensorData-Without-KeyEvents (S) // Used in the attacking phase
1 T = Detect-KeyDown-Timestamps (S)
2 W = Identify-Keystroke-TimeWindows (T)
3 W = Adjust-Keystroke-TimeWindows (W)
4 **return** W

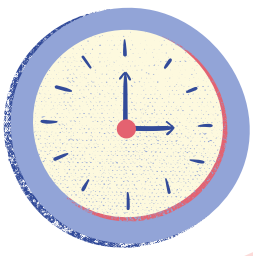
Detect-KeyDown-Timestamps (S)
1 S = Filter-Data (S, start_frequency, end_frequency)
2 $M^A = M^R = ()$ // Magnitude for acceleration forces and rotation rates
3 **for** t **in** $t_1 : t_n$
4 $M_t^A = \sqrt{x_t{}^2 + y_t{}^2 + z_t{}^2}$; $M_t^R = \sqrt{\alpha_t{}^2 + \beta_t{}^2 + \gamma_t{}^2}$
5 T^A = Find-Peak-Timestamps (M^A); T^R = Find-Peak-Timestamps (M^R)
6 T = Merge-Peak-Timestamps (T^A, T^R)
7 **return** T

Identify-Keystroke-TimeWindows (T)
1 **for** j **in** $1 : m$
2 $W_j^S = T_j - offset\_start$; $W_j^E = T_j + offset\_end$
3 **return** W

Adjust-Keystroke-TimeWindows (W)
1 **for** j **in** $1 : m - 1$
2 $overlap = W_j^E - W_{j+1}^S$ // Overlap between two keystrokes
3 **if** $overlap \leq 0$ // No overlap
4 // Do nothing
5 **else if** $overlap > ((W_{j+1}^S + offset\_start) - (W_j^E - offset\_end)) \times overlap\_threshold$ // Heavy overlap
6 mark W_j and W_{j+1} as heavily overlapped time windows
7 **else** // Slight overlap, split the overlapped region
8 $W_j^E = W_j^E - overlap/2$; $W_{j+1}^S = W_{j+1}^S + overlap/2$
9 remove the marked heavily overlapped time windows from W
10 **return** W

1

EL SENSOR DE MOVIMIENTO DE SGMENTACIÓN DE DATOS EN LA FASE DE ENTRENAMIENTO ACEPTA UNA SECUENCIA DE TECLAS COMO LA ENTRADA, IDENTIFICA Y AJUSTA UNA SECUENCIA DE PULSACIONES DE TECLA QUE SE RETORNA COMO RESULTADO. EN LA FASE DE ATAQUE ACEPTA DATOS DEL SENSOR DE MOVIMIENTO COMO ENTRADA



2

ELTRAINING DATA SCREENING CALCULA CARACTERERS ESPECIFICOS PARA PULSACIONES DE TECLA INDIVIDUALES Y SOLO USA EL SENDOR DE MOVIMIENTO DE DATOS PARA ENTRENAR AL CLASIFICADOR

3

EL FILTRADO DE DATOS SE USA PARA DIRIGIR LOS DATOA EN UN ATAQUE DE INFERENCIA DE DATOS CROSS-SITE. ESTE SELECCIONA FRECUENCIAS DE BANDA PARA FILTRAR LOS DATOS Y REDUCIR EL RUIDO EN LOS SENSORES DE MOVIMIENTO DE DATOS

4

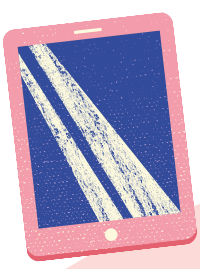
EN LA EXTRACCIÓN DE CARACTERISTICAS SE EXTRAEN POTENCIALES SETS DE CARACTERISTICAS ESTADISTICAS DE LOS DATOS FILTRADOS DE LAS PULSACIONES DE TECLAS

5

EN EL MODELO DE ENTRENAMIENTO SE EXPERIMENTA CON VARIEDAD DE ALGORITMOS DE MACHINE LEARNING USANDO WEKA, ESTOS ALGORITMOS INCLUYEN REGRESIÓN LÓGICA, REDES BAYES, ARBOL DE DECISIONES, ETC

6

EL FRAMEWORK SE IMPLEMENTÓ USANDO JS, HTML Y PHP PARA LOS SENSORES DE MOVIMIENTO DE DATOS Y LA COLECCIÓN DE EVENTOS. TAMBIÉN SE USO JAVA, R Y WKA PARA ENTRENAR LOS CALSIFICADORES.



7

BÁSICAMENTE SE PUEDEN USAR LOS ACELEROMETROS Y GIROSCOPIOS DE LOS DISOSITIVOS MOVILES PARA HACER ATAQUES CROSS-SITE Y COMPROMETER LAS SEGURIDAD DE MUCHOS DISPOSITIVOS MÓVILES

Java