Continuous User Verification through Behavior Biometrics

Dipl.-Inform Arik Messerman
Agenda

1. Introduction, Motivation
2. Focus
3. Deeper view to a Keystroke Dynamic Approach
4. Milestones, Discussion
Introduction to Biometrics

- **Physiological Biometric**
  - Passive approach
    - Measure distinct traits that humans have
    - Do not vary over time
  - Iris scans, retina scans, fingerprints, DNA, …

- **Behavioral Biometric**
  - Active approach
    - Measure performed tasks
    - Do vary over time
  - Types of behavioral Biometrics
  - Each subdivision has its own characteristics in terms of
    - usage, deploy ability, user acceptance, quality, …
Motivation (Behavioral Biometrics)

- Humans can be verified traditionally by / through …
  - Knowledge (passwords, PINs, ..)
  - Ownership (software/security token, ID card, …)
  - Inherence (fingerprint, voice, interaction, …)
    - In most cases: Physiological Biometrics

- Risk for traditional solutions:
  - Object is verified, object == actor

- Additional security layer is required → behavioral biometrics
  - Further requirements
Agenda

1. Introduction, Motivation
2. Focus
3. Deeper view to a Keystroke Dynamic Approach
4. Milestones, Discussion
Focus: Transparent Continuous Verification

- **Focus:** “Development and trustable evaluation of reactive, transparent and
  free-action-based, continuous user verification solutions with low error rates
  under real-time environments and conditions with minimum and user-friendly
  requirements to stakeholders through Keystroke Dynamic approaches in the
  field of Behavioral Biometrics.”

- **User verification**
  - **Free text** → Most solutions: Fix
  - **Continuous** → Most solutions: Initial
  - **Transparent** → Most solutions: Defined action to perform required
  - **Low error rates** → Many solutions: Evaluation under unreal environments
  - **Short response times** → Many solutions: Not really considered
  - **User model update** → Most solutions: Static enrollment
  - **Large user data** → Most solutions: Evaluation based on a very limited amount of data
  - **Comparable evaluations (Open DB)** → In the field of Keystroke Dynamics not given

- **Deployment**
  - With minimum effort into real-time environments
  - Without any additional hardware-equipment
Agenda

1. Introduction, Motivation
2. Focus
3. Deeper view to a Keystroke Dynamic Approach
4. Milestones, Discussion
Edit distance calculation (Free text)*

Database of user u:

|   | ab | ll | ff | ce | ef | by | gk | gl | ew | kl | mn | op | oz | qr | th | uv | wx | nt | yz |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
|   | 10ms | 11ms | 14ms | 15ms | 16ms | 17ms | 19ms | 20ms | 20ms | 23ms | 24ms | 27ms | 27ms | 28ms | 30ms | 33ms | 35ms | 36ms | 41ms |

New typing sample: „i will buy a new table ...open the door, ... efficient“

Distance calculation*

Database of user u:

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| ab| 1l| ff| ce| ef| uy| gk| gl| ew| kl| mn| op| oz| qr| th| uv| wx| nt| yz|
|10ms|11ms|14ms|15ms|16ms|17ms|19ms|20ms|20ms|23ms|24ms|27ms|27ms|28ms|30ms|33ms|35ms|36ms|41ms|

New typing sample: „i will buy a new table ...open the door, ... efficient“

Distance calculation*

Database of user u:

```
  ab  ll  ff  ce  ef  ab  yl  ck  ff  gb  by  ew  ew  klo  m  n  t  op  nt  cz  qr  th  uv  wx  nt  yz
10ms 11ms 14ms 15ms 16ms 19ms 20ms 23ms 24ms 25ms 26ms 28ms 30ms 33ms 35ms 36ms 41ms
```

New typing sample: „i will buy a new table ...open the door, ...efficient“

```
  ab  ll  ff  th  uy  ew  op  nt
14ms 16ms 18ms 19ms 21ms 28ms 31ms 38ms
```

Distance calculation*

Database of user u:

<table>
<thead>
<tr>
<th>ab</th>
<th>ll</th>
<th>ff</th>
<th>uy</th>
<th>ew</th>
<th>op</th>
<th>th</th>
<th>nt</th>
</tr>
</thead>
<tbody>
<tr>
<td>10ms</td>
<td>11ms</td>
<td>14ms</td>
<td>17ms</td>
<td>20ms</td>
<td>27ms</td>
<td>30ms</td>
<td>36ms</td>
</tr>
</tbody>
</table>

New typing sample: „i will buy a new table …open the door, … efficient“

Distance:
\[
\frac{(1+1+1+3)}{(0.5 \times 8^2)} = 0.1875
\]

Distance calculation*

Pattern of a user can be regarded as an **array** with values

- Calculation of the **distance** between patterns from the user data base and new one is to reduce to the calculation of the position of elements in permutations

Verification*

$n = 60 \Rightarrow n! = 60! = 8,321 \times 10^{81}$ (after 2 Month on 8 CPU)

$n = 60$ (Random experiment after 20,249,500,000 (1 Month)/ 44,457,500,000 (2 Month) randomly chosen permutations)

**Theory**

```
%   0.0   0.1   0.2   0.3   0.4   0.5   0.6   0.7   0.8   0.9   1.0
0.0%  0.1%  99.9%
```

January 12, 2010

Behavioral Biometrics

Slide 14
Experiment with real user data

Number of other Users (Extended1_R)

Reason: Maximum distance (0.45)

FAR < 3% without additional requirements

Practice!
Bioinformatics approach

- Distance calculation → Edit distance
- Evolution-Theory:
  - Combinations of amino acids are specified through sequences of nucleotides in DNA → Genes
    - Edit distances between DNA, RNA or protein strings
    - Protein: Sequence of units = amino acids
  - Example: glyceraldehyd3-phosphate dehydrogenase (GADPH) protein
    - Fly: GAKKVIISAPSAD–APM–F
    - Human: GAKRVIISAPSAD–APM–F
    - Yeast: GAKKVSTAPSS–TPM–F
  - How closely related are two strings which represent the amino acid sequence of a particular gene between two species?
  - From a computer science perspective this issue is one of pattern matching and search
Bioinformatics approach

From a computer science perspective this issue is one of pattern matching and search

- Idea:
  - Apply the huge amount practical and theoretical research that have been successfully developed in bioinformatics to the task of authentication/verification [/Anomaly detection]

- Other distance metrics:
  - Levenshtein distance:
    - Levenshtein distance between two strings is given by the minimum number of operations needed to transform one string into the other, where an operation is an insertion, deletion, or substitution (???) of a single character
  - Hamming, Euclidian, Cayley, Ulam, Spearman’s Footrule, Spearman’s rank correlation, Kendall’s tau
  - From a computer science perspective this issue is one of pattern matching and search
Agenda

1. Introduction, Motivation
2. Focus
3. Deeper view to a Keystroke Dynamic Approach
4. Milestones, Discussion
Milestones (1)

- Parallel work in project “Activity-based Verification” (until 06.2010)
  - Work packages for DAI reflect next steps of my dissertation
    - e.g. new distance metrics/approaches

- IEEE ISI 2009: “Identity Theft, Computers and Behavioral Biometrics” (Jun 09), Dallas, TX

- Survey journal: Draft version (45 pages)
  - State of the art, deeper discussions, review and novel views in the field of B.B.

- Verification & Evaluation Service Platform paper: Pre-draft version
  - Conception and Implementation of a generic platform was made

- Several verification approaches
  - Focus: Continuous Free Text Verification
Milestones (2)

- **Web mail application**
  - Enable transparent collection of behavioral data
  - Large dataset of user behavior
  - Currently: 52 users, ~5000 ‘KeyDown’ events, Goal: ~100 users

- **Theoretical/Scientific work paper: Paper planned (Start Sept. 09)**

- **Smart Senior: "Erkennung von Notsituationen im häuslichen Umfeld durch sensorbasierte Analyse von Verhaltensanomalien" Paper planend (Start Sept. 09)**
  - Adapt knowledge made in the field of Behavioral Biometrics to anomaly detection

- **Bachelor/Diploma thesis**
  - 1. Adoption of existing (own) methods to login verification (Start 08.2009)
  - 2. Generic evaluation engine of AbV verification methods (Start 07.2009)
  - 3. Adoption of existing (own) methods to Smart Phone environments (Start 08.2009)
Some References

- K. Hempstalk, *Continuous Typist Verification using Machine Learning*, PhD, University of Waikato, New Zealand, Department of Computer Science, 2009