Continuous and Non-Intrusive Reauthentication of Web Sessions based on Mouse Dynamics

Eric Medvet      Alberto Bartoli      Francesca Boem      Fabiano Tarlao

Department of Engineering and Architecture
University of Trieste
Italy

September 10th, 2014

http://machinelearning.inginf.units.it
Table of Contents

1. Scenario and motivation

2. Our contribution
   - Data capture system
   - Reauthentication by mouse dynamics

3. Experimental evaluation
   - Dataset
   - Results
(Re)Authentication

Credentials stealing is not an exceptional event

- *bad* current user with *good* credentials, possibly for a long time
(Re)Authentication

Credentials stealing is *not* an exceptional event

- *bad* current user with *good* credentials, possibly for a long time
→ verify the user identity over the time
(Re)Authentication

Credentials stealing is *not* an exceptional event

- *bad* current user with *good* credentials, possibly for a long time

→ verify the user identity over the time

  - by other means than credentials
  - possibly non-intrusively
Behavioral biometrics

Non-intrusive continuous verification of the user identity
→ Behavioral biometrics:
  - keystrokes
  - ...
  - mouse trajectories
Scenario and motivation

Behavioral biometrics

Non-intrusive continuous verification of the user identity

→ Behavioral biometrics:
  - keystrokes
  - ...
  - mouse trajectories
Behavioral biometrics

Non-intrusive continuous verification of the user identity

→ Behavioral biometrics:
  - keystrokes
  - ...
  - mouse trajectories

Machine instrumentation for collecting biometrics may be unpractical for large distributed organizations
So, we are concerned in:

- continuous reauthentication
- using mouse dynamics
- collected w/o specific software installed on client machine
Scenario

We chose to address:
- web
- full transparency to server and client

Suitable for:
- large organizations w/ user web access
- (private) cloud hosted enterprise applications
Example

Large organizations w/ user web access:
Example

Large organizations w/ user web access:
1. X authenticates with Alice’s credentials on her organization...
Scenario and motivation

Example

Large organizations w/ user web access:

1. X authenticates with Alice’s credentials on her organization
2. X browses the web (any website) and...
Example

Large organizations w/ user web access:

1. X authenticates with Alice’s credentials on her organization
2. X browses the web (any website) and...
3. ...if X’s behaviour is different enough from Alice’s known behavior, an alert is eventually raised
Example

Large organizations w/ user web access:

1. $X$ authenticates with Alice’s credentials on her organization
2. $X$ browses the web (any website) and...
3. ...if $X$’s behaviour is different enough from Alice’s known behavior, an alert is eventually raised

Authentication
Scenario and motivation

Example

Large organizations w/ user web access:
1. X authenticates with Alice’s credentials on her organization
2. X browses the web (any website) and...
3. ...if X’s behaviour is different enough from Alice’s known behavior, an alert is eventually raised

Authentication, then reauthentication in the web using mouse dynamics.
Large organizations w/ user web access:

1. $X$ authenticates with Alice’s credentials on her organization
2. $X$ browses the web (any website) and...
3. ...if $X$’s behaviour is different enough from Alice’s known behavior, an alert is eventually raised

Authentication, then reauthentication in the web using mouse dynamics. Aim at detecting long lasting systematic fraudulent account usage (*defense-in-depth*).
# Table of Contents

1. Scenario and motivation

2. Our contribution
   - Data capture system
   - Reauthentication by mouse dynamics

3. Experimental evaluation
   - Dataset
   - Results
In a nutshell:

- a system for capturing web GUI-related events transparent for user and web site
- a procedure for performing continuous reauthentication using mouse-generated events
Data capture system: overview

- a web proxy
- a js (collects data)
- a web app (receives and analyzes data)
How it works

Browser $C$

Proxy $P$

Web server $S$

Web app $O$
How it works

Browser C

Proxy P

Web server S

Web app O

GET /index.html

GET /img/img.png

GET /obs/observer.js

POST /obs

C requests HTML document to S, S responds with d

P injects our js URL (src="/obs/observer.js")

C requests resources mentioned in d: our js comes from O (rather than S via P)

our js on C sends mouse events data to /obs, i.e.,
How it works

Browser $C$

Proxy $P$

Web server $S$

Web app $O$
How it works

Browser C

Proxy P

Web server S

Web app O

GET /index.html

GET /img/img.png

GET /obs/observer.js

POST /obs

C requests HTML document to S, S responds with d

P injects in d our js URL (src="/obs/observer.js")

C requests resources mentioned in d js: our js comes from O (rather than S) via P

our js on C sends mouse events data to /obs, i.e,
**How it works**

1. C requests HTML document to S, S responds with d
How it works

1. $C$ requests HTML document to $S$, $S$ responds with $d$
2. $P$ injects in $d$ our js URL (src="/obs/observer.js")
How it works

1. C requests HTML document to S, S responds with d
2. P injects in d our js URL (src="/obs/observer.js")
3. C requests resources mentioned in dj

Our contribution
Data capture system
How it works

1. C requests HTML document to S, S responds with d
2. P injects in d our js URL (src="/obs/observer.js")
3. C requests resources mentioned in d<sup>js</sup>: our js comes from O (rather than S) via P
How it works

1. C requests HTML document to S, S responds with d
2. P injects in d our js URL (src="/obs/observer.js")
3. C requests resources mentioned in djs: our js comes from O (rather than S) via P
4. our js on C sends mouse events data to /obs, i.e., O
Data capture system

- fully transparent to both user and web sites, requires only to set the proxy
- redirection of /obs/* traffic allows to circumvent Same Origin Policy
- low bandwidth usage ($\approx 2.5 \text{ kB s}^{-1}$)
- can work with HTTPS (w/ self-signed certificate)

---

1 Bartoli, Medvet, Mauri, Recording and Replaying Navigations on AJAX Web Sites, Int. Conf. on Web Engineering (ICWE), 2012
Our contribution

Data capture system

- fully transparent to both user and web sites, requires only to set the proxy
- redirection of /obs/* traffic allows to circumvent Same Origin Policy
- low bandwidth usage ($\approx 2.5 \text{ kB s}^{-1}$)
- can work with HTTPS (w/ self-signed certificate)
- could be used also for other purposes: web app testing\(^1\), web app misuse detection, ... 

---

\(^1\)Bartoli, Medvet, Mauri, *Recording and Replaying Navigations on AJAX Web Sites*, Int. Conf. on Web Engineering (ICWE), 2012
Data capture system generates an event \( e = (x, y, t) \) every \( \approx 25 \text{ ms} \), then we:

1. split sequence of events on pauses \( \geq 500 \text{ ms} \) and consider the last 10 events before a pause (trajectory)
2. transform a trajectory \( T \) into a vector \( \mathbf{f}(T) \in \mathbb{R}^{39} \)
3. classify \( \mathbf{f}(T) \) as anomalous/normal, w.r.t. current authenticated user
Our contribution
Reauthentication by mouse dynamics

Features

\( f(T) \) includes:

- directions and direction changes
- speeds
- accelerations
- \( x \)- and \( y \)-extents
Classification

Two phases:
- training
- actual classification
Classification

Two phases: \((U^-)\) is the authenticated user

- training based on trajectories of \(U^-\) and other users \(U_1^+, U_2^+, \ldots\)
- actual classification
Classification

Two phases: ($U^-$ is the authenticated user)
- training based on trajectories of $U^-$ and other users $U_1^+, U_2^+, \ldots$
- actual classification based on trajectories of current unknown user $U$ claiming to be $U^-$
Our contribution
Reauthentication by mouse dynamics

Training phase

Once, at the beginning:

1. train a $\text{SVM}_{U-}$ on the training set
Actual classification phase

For each $T$ trajectory of $U$:

1. apply $\text{SVM}_{U^-}$ to $f(T)$
2. consider last $w$ trajectories and...
3. ...if too many positives, raise an alert
Last $w$ trajectories

Aggregation of several classifier outcomes:
- often used with mouse dynamics
- the higher $w$,
  - the higher the accuracy and
  - the longer the *Time to Detection* (TtD)
# Table of Contents

1. Scenario and motivation

2. Our contribution
   - Data capture system
   - Reauthentication by mouse dynamics

3. Experimental evaluation
   - Dataset
   - Results
Two groups of users, each observed for several working days:

- 6 users, with different hardware equipment
- 18 users, with homogeneous hardware
## Results

<table>
<thead>
<tr>
<th>w (min)</th>
<th>TtD (min)</th>
<th>Dataset 1</th>
<th>Dataset 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc.</td>
<td>FAR</td>
<td>FRR</td>
</tr>
<tr>
<td>50</td>
<td>83.3</td>
<td>16.6</td>
<td>16.7</td>
</tr>
<tr>
<td>100</td>
<td>88.5</td>
<td>12.8</td>
<td>10.2</td>
</tr>
<tr>
<td>200</td>
<td>93.5</td>
<td>9.2</td>
<td>3.8</td>
</tr>
<tr>
<td>350</td>
<td>95.6</td>
<td>7.9</td>
<td>1.0</td>
</tr>
<tr>
<td>500</td>
<td>96.5</td>
<td>6.1</td>
<td>0.8</td>
</tr>
</tbody>
</table>
### Results

<table>
<thead>
<tr>
<th>w (min)</th>
<th>TtD (min)</th>
<th>Dataset 1</th>
<th>Dataset 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc.</td>
<td>FAR</td>
<td>FRR</td>
</tr>
<tr>
<td>50</td>
<td>83.3</td>
<td>16.6</td>
<td>16.7</td>
</tr>
<tr>
<td>100</td>
<td>88.5</td>
<td>12.8</td>
<td>10.2</td>
</tr>
<tr>
<td>200</td>
<td>93.5</td>
<td>9.2</td>
<td>3.8</td>
</tr>
<tr>
<td>350</td>
<td>95.6</td>
<td>7.9</td>
<td>1.0</td>
</tr>
<tr>
<td>500</td>
<td>96.5</td>
<td>6.1</td>
<td>0.8</td>
</tr>
</tbody>
</table>

- Accuracy up to 96%
## Results

<table>
<thead>
<tr>
<th>$w$</th>
<th>TtD (min)</th>
<th>Dataset 1</th>
<th>Dataset 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Acc.</td>
<td>FAR</td>
</tr>
<tr>
<td>50</td>
<td>13.5</td>
<td>83.3</td>
<td>16.6</td>
</tr>
<tr>
<td>100</td>
<td>27.1</td>
<td>88.5</td>
<td>12.8</td>
</tr>
<tr>
<td>200</td>
<td>54.1</td>
<td>93.5</td>
<td>9.2</td>
</tr>
<tr>
<td>350</td>
<td>94.7</td>
<td>95.6</td>
<td>7.9</td>
</tr>
<tr>
<td>500</td>
<td>135.3</td>
<td>96.5</td>
<td>6.1</td>
</tr>
</tbody>
</table>

- accuracy up to 96%
- works better if attacker uses different hardware
Results

<table>
<thead>
<tr>
<th>$w$ (min)</th>
<th>TtD (min)</th>
<th>Dataset 1</th>
<th>Dataset 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc.</td>
<td>FAR</td>
<td>FRR</td>
</tr>
<tr>
<td>50</td>
<td>13.5</td>
<td>83.3</td>
<td>16.6</td>
</tr>
<tr>
<td>100</td>
<td>27.1</td>
<td>88.5</td>
<td>12.8</td>
</tr>
<tr>
<td>200</td>
<td>54.1</td>
<td>93.5</td>
<td>9.2</td>
</tr>
<tr>
<td>350</td>
<td>94.7</td>
<td>95.6</td>
<td>7.9</td>
</tr>
<tr>
<td>500</td>
<td>135.3</td>
<td>96.5</td>
<td>6.1</td>
</tr>
</tbody>
</table>

- accuracy up to 96%
- works better if attacker uses different hardware
- time to detection of tens of minutes
Time to detection of tens of minutes: is it practical?

- fits the threat model
- we can only monitor web usage (browser)
  - user could unfocus the browser for minutes
  - we consider sessions without pauses $\geq 10$ minutes
Thanks!