Practical exploitation of rounding vulnerabilities in internet banking applications

Adrian Furtună, PhD, OSCP, CEH
adif2k8@gmail.com
Agenda

- Who am I
- Rounding vulnerabilities
- How to fix
- Exploitation techniques
- Digipass automation
- Demo
Who am I

- PhD in Information Security, OSCP, CEH
- Penetration tester at KPMG Romania
  - Web applications, internet banking
  - Network infrastructures
  - Mobile applications
  - Source code reviews
  - + some annoying stuff
- Teaching assistant at Information Security Master programs from Bucharest universities
  - Teaching penetration testing classes
  - Organizing Capture the Flag contests
- Always like to prove my point…
Rounding vulnerabilities
Real life example

- How much do you really pay?
Real life example

- How much do you *really* pay?
- What about: 
  \[2.85e + 3.20e = 6.05e\]?
Real life example

- How much do you *really* pay?
- What about: $2.85e + 3.20e = 6.05e$?
- How much does the seller win from rounding?
Real life example

- How much do you *really* pay?
- What about: 
  \[2.85e + 3.20e = 6.05e\] ?
- How much does the seller win from rounding?
- We are a bit vulnerable…
Rounding vulnerabilities

In Internet Banking apps

- Banks are vulnerable also
Rounding vulnerabilities

In Internet Banking apps

- Banks are vulnerable also
- Amounts are specified with two decimals:

<table>
<thead>
<tr>
<th>IBAN</th>
<th>Currency</th>
<th>Current Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Account</td>
<td>EUR</td>
<td>0.67</td>
</tr>
<tr>
<td>RO6000000001360445</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Current Account</td>
<td>RON</td>
<td>49.00</td>
</tr>
<tr>
<td>RO6600000001360434</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
In Internet Banking apps

- Banks are vulnerable also
- Amounts are specified with two decimals:

<table>
<thead>
<tr>
<th>IBAN</th>
<th>Currency</th>
<th>Current Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Account</td>
<td>EUR</td>
<td>0.67</td>
</tr>
<tr>
<td>Current Account</td>
<td>RON</td>
<td>49.00</td>
</tr>
</tbody>
</table>

- What happens when you transfer 8.3436 EUR to your account?
  Amount += 8.34 EUR => **Bank wins 0.0036 EUR**
Rounding vulnerabilities

In Internet Banking apps

- Banks are vulnerable also
- Amounts are specified with two decimals:

<table>
<thead>
<tr>
<th>IBAN</th>
<th>Currency</th>
<th>Current Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Account</td>
<td>EUR</td>
<td>0.67</td>
</tr>
<tr>
<td>Current Account</td>
<td>RON</td>
<td>49.00</td>
</tr>
</tbody>
</table>

- What happens when you transfer 8.3436 EUR to your account?
  Amount += 8.34 EUR  => Bank wins 0.0036 EUR

- What happens when you transfer 8.3478 EUR to your account?
  Amount += 8.35 EUR  => Bank loses 0.0022 EUR
In Internet Banking apps

- Banks are vulnerable also
- Amounts are specified with two decimals:

<table>
<thead>
<tr>
<th>IBAN</th>
<th>Currency</th>
<th>Current Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current Account RO600210000001360445</td>
<td>EUR</td>
<td>0.67</td>
</tr>
<tr>
<td>Current Account RO660210000001360434</td>
<td>RON</td>
<td>49.00</td>
</tr>
</tbody>
</table>

- What happens when you transfer 8.34\textcolor{red}{36} EUR to your account?
  Amount += 8.34 EUR  => Bank wins 0.0036 EUR

- What happens when you transfer 8.34\textcolor{red}{78} EUR to your account?
  Amount += 8.35 EUR  => Bank loses 0.0022 EUR

- Max to win/lose: 0.005 EUR / transaction
  Rounding is done to the closest value (two decimals)
How to always win?

- Let’s make transactions that will be always rounded in our favor
How to always win?

- Let’s make transactions that will be always rounded in our favor
- How?
  - Foreign exchange transactions
  - Transfer between your own accounts having different currencies
Obtain a better exchange rate

- Transfer money between your own accounts (e.g. RON -> EUR)
- Specify how much RON you want to sell

<table>
<thead>
<tr>
<th>RON</th>
<th>EUR</th>
<th>EUR (rounded)</th>
<th>Actual exchange rate (RON / EUR rounded)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.40</td>
<td>1</td>
<td>1.00</td>
<td>4.40</td>
</tr>
<tr>
<td>2</td>
<td>0.4545</td>
<td>0.45</td>
<td>4.44</td>
</tr>
<tr>
<td>1</td>
<td>0.2272</td>
<td>0.23</td>
<td>4.34</td>
</tr>
<tr>
<td>0.5</td>
<td>0.1136</td>
<td>0.11</td>
<td>4.54</td>
</tr>
<tr>
<td>0.05</td>
<td>0.0113</td>
<td>0.01</td>
<td>5</td>
</tr>
<tr>
<td>0.03</td>
<td>0.0068</td>
<td>0.01</td>
<td>3</td>
</tr>
<tr>
<td>0.023</td>
<td>0.0052</td>
<td>0.01</td>
<td>2.3</td>
</tr>
<tr>
<td>0.02</td>
<td>0.0045</td>
<td>0.00</td>
<td>not good</td>
</tr>
</tbody>
</table>

100 * (0.023 RON -> 0.01 EUR) => 2.3 RON = 1 EUR
Rounding vulnerabilities

Example (1)

<table>
<thead>
<tr>
<th>Date</th>
<th>Transaction</th>
<th>Debit/Credit</th>
<th>Balance intermediary</th>
</tr>
</thead>
<tbody>
<tr>
<td>13/02/2012</td>
<td>Schimb valutar</td>
<td>+ 0.01</td>
<td>0.03</td>
</tr>
<tr>
<td></td>
<td>Referinta: 7</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Din contul: RO19</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Suma: 0,03 RON</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rata: 4,40580</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13/02/2012</td>
<td>Schimb valutar</td>
<td>+ 0.01</td>
<td>0.04</td>
</tr>
<tr>
<td></td>
<td>Referinta: 8</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Din contul: RO11</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Suma: 0,03 RON</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rata: 4,40580</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13/02/2012</td>
<td>Schimb valutar</td>
<td>+ 0.01</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td>Referinta: 9</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Din contul: RO11</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Suma: 0,02 RON</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rata: 4,40580</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Échanges Valutaires

<table>
<thead>
<tr>
<th>Client</th>
<th>FURTUNA CONSTANTIN-ADRIAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUI / CNP</td>
<td></td>
</tr>
<tr>
<td>Compte vente</td>
<td>RO78...0210000001360412 (RON)</td>
</tr>
<tr>
<td>Montant vendu</td>
<td>0.025 RON</td>
</tr>
<tr>
<td>Compte achat</td>
<td>RO72...0210000001360423 (EUR)</td>
</tr>
<tr>
<td>Montant acheté</td>
<td>0.01 EUR</td>
</tr>
<tr>
<td>Rata de schimb</td>
<td>4.4614</td>
</tr>
<tr>
<td>Date d'ordre</td>
<td>05/06/2013</td>
</tr>
</tbody>
</table>

La transaction a été introduite le 05/06/2013 à 13:10:44 heure par CONSTANTIN-ADRIAN FURTUNA.
La transaction a été signée le 05/06/2013 à 13:10:59 heure par CONSTANTIN-ADRIAN FURTUNA (E).

'autorisé. Referinta bancii: 021FT24131560016
Traité avec succès!
6/15/2013 6:15:33 PM
When is the best deal

- **Foreign exchange transactions:**
  - Specify how much you want to sell => destination will be rounded
  - Specify how much you want to buy => source will be rounded

- Best deal is when you can specify how much of the weaker currency you want to sell/buy because the stronger currency will be rounded

```
From Account: RO62 0000999900307770 [RON] ▼
Balance: 25.00 RON
Amount: 0.11 [RON] ▼
To Account: RO86 0000999901801526 [EUR] ▼
Beneficiary Name: Constantin Adrian Furtuna
Exchange Rate: 1 EUR = 4.5985 RON
```
```
From Account: RO62 0000999900307770 [RON] ▼
Balance: 25.00 RON
Amount: 0.01 [EUR] ▼
To Account: RO86 0000999901801526 [EUR] ▼
Beneficiary Name: Constantin Adrian Furtuna
Exchange Rate: 1 EUR = 4.5985 RON
```
How much can I gain?

C1 = minimum amount of currency 1 that can be exchanged (e.g. 0.023 RON)
C2 = minimum amount of currency 2 that can be exchanged (e.g. 0.01 EUR)
Ex_b = exchange rate for buying C2 with microtransactions (e.g. 2.3)

$$\text{Ex}_b = \frac{\text{C1}}{\text{C2}}$$

Ex_s = exchange rate for selling C2 (e.g. 4.4) – real exchange rate – fixed by the Bank

Rounding vulnerabilities

- \( z = y \times \text{Ex}_s = \left(\frac{x}{\text{Ex}_b}\right) \times \text{Ex}_s = x \times \left(\frac{\text{Ex}_s}{\text{Ex}_b}\right) \)
- multiplication rate = \( \text{Ex}_s / \text{Ex}_b \)
- transactions required = \( x / \text{C1} \)

<table>
<thead>
<tr>
<th>Currency</th>
<th>Multiplication rate</th>
<th>Initial amount (x)</th>
<th>Final amount (y)</th>
<th>Gain</th>
<th>Transactions required</th>
</tr>
</thead>
<tbody>
<tr>
<td>RON</td>
<td>4.4 / 2.3 = 1.9</td>
<td>100 RON</td>
<td>190 RON</td>
<td>90 RON ~ 20 EUR</td>
<td>100 / 0.023 = 4347</td>
</tr>
</tbody>
</table>
Different exchange rates (buy / sell)

- Banks have different exchange rates for buying and for selling so they can always win

- Let’s say...
  - Official exchange rate: 4.45
  - You buy from the Bank: 4.50
  - You sell to the Bank: 4.40

- But for small amounts it is not true!
  - I buy from the Bank (RON → EUR)
    - 0.45 RON / 4.40 = 0.102 EUR → 0.1 EUR
    - 0.45 RON / 4.50 = 0.100 EUR → 0.1 EUR
    - 0.45 RON / 4.60 = 0.097 EUR → 0.1 EUR
    - 0.45 RON / 4.70 = 0.095 EUR → 0.1 EUR
How to fix
How the Banks should protect themselves

- Limit the number of transactions that can be performed in a given time by a regular person
- Introduce a small fee for currency exchange operations (e.g. 0.01 EUR)
- Limit the minimum amount that can be transferred in a foreign exchange operation
- Monitor for suspicious transactions (numerous transactions, very small amounts)
- State in the contract that such transactions are illegal
Exploitation techniques
General ideas

- Find a way to do lots of transactions in a relatively short time
- Transactions are made in two steps:
  - Initialization (can be automated)
  - Authorizing / Signing (requires human interaction)
- Automate / bypass transaction signing mechanism (digipass, SMS, token, etc)
Exploitation techniques

Technique 0: No signing required 😊

- 3000 transactions, 90 minutes, 30 RON → 73 RON, gain ~10 EUR
Technique 1: Init lots and sign once

- Initiate lots of transactions automatically and sign once
Technique 1: Init lots and sign once

- Initiate lots of transactions automatically and sign once
- Use Burp Suite to initiate transactions
Exploitation techniques

Technique 1: Init lots and sign once

• Initiate lots of transactions automatically and sign once

Sign operations
To sign operations created prior to current date use the search filter.

<table>
<thead>
<tr>
<th>Fill-in date</th>
<th>Foreign exchange</th>
<th>Exchange rate Negotiated rate</th>
<th>Sell amount Buy amount</th>
<th>Status</th>
<th>Message</th>
<th>Signed</th>
</tr>
</thead>
<tbody>
<tr>
<td>15/06/2013 14:22:35</td>
<td>4.4960 Without negociation</td>
<td>0.025 RON 0.01 EUR</td>
<td>Approved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15/06/2013 14:22:35</td>
<td>4.4960 Without negociation</td>
<td>0.025 RON 0.01 EUR</td>
<td>Approved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15/06/2013 14:22:34</td>
<td>4.4960 Without negociation</td>
<td>0.025 RON 0.01 EUR</td>
<td>Approved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15/06/2013 14:22:34</td>
<td>4.4960 Without negociation</td>
<td>0.025 RON 0.01 EUR</td>
<td>Approved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15/06/2013 14:22:33</td>
<td>4.4960 Without negociation</td>
<td>0.025 RON 0.01 EUR</td>
<td>Approved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15/06/2013 14:22:33</td>
<td>4.4960 Without negociation</td>
<td>0.025 RON 0.01 EUR</td>
<td>Approved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15/06/2013 14:22:33</td>
<td>4.4960 Without negociation</td>
<td>0.025 RON 0.01 EUR</td>
<td>Approved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15/06/2013 14:22:33</td>
<td>4.4960 Without negociation</td>
<td>0.025 RON 0.01 EUR</td>
<td>Approved</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15/06/2013 14:22:33</td>
<td>4.4960 Without negociation</td>
<td>0.025 RON 0.01 EUR</td>
<td>Approved</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 2

SIGN ALL SELECTED ORDERS FILTER
Technique 1: Init lots and sign once

- Initiate lots of transactions automatically and sign once
- Signing can also be automated – stay tuned for next chapter
Technique 2: Payment files

- Upload a payment file containing lots of transactions and sign once
Exploitation techniques

Technique 2: Payment files

- Upload a payment file containing lots of transactions and sign once

- Signing can also be automated – stay tuned for next chapter
Technique 3: Real time transactions + rainbow tables

- Do real time transactions automatically and sign using pre-computed digipass responses
Technique 3: Real time transactions + rainbow tables

- Do real time transactions automatically and sign using pre-computed digipass responses
- Applicable when signing is done using challenge-response mechanism, with challenge code <= 5 digits
Technique 3: Real time transactions + rainbow tables

- A challenge-response digipass returns the same response for the same challenge code every time
  \[
  \text{Response} = f(\text{challenge, timestamp, client ID, other data}) \\
  = f(\text{challenge, static data})
  \]

- Build rainbow tables with digipass responses
  - Feasible for max 5 digit challenge codes
  - Max 99999 possibilities
  - Can be automated, stay tuned
Exploitation techniques

Technique 4: Real time transactions + digipass automation

- Do real time transactions automatically and sign using digipass responses computed in real time
Exploitation techniques

Technique 4: Real time transactions + digipass automation

- Do real time transactions automatically and sign using digipass responses computed in real time
- Requires automation of the signing device (digipass, phone, etc)
Digipass automation
LimID project (for VASCO GO3)

- [http://limid.sitadella.com](http://limid.sitadella.com)
- Code regenerates at 30 seconds
- Video
My machine (for VASCO 550)

Requires PIN authentication

Used for:
- 2\textsuperscript{nd} factor authentication
- Transaction signing
My machine - video
My machine - current performance

- 10 transactions / minute (1 transaction / 6 seconds)
  - max 14400 transactions / day
  - enter PIN, type challenge code, read response image, do OCR

- Our previous example:
  100 RON → 190 RON (gain ~20 EUR)
  => 4347 transactions * 6 sec/transaction = 26082 sec
  = 7h:14m:42 s

- Maximum amount to multiply per day:
  14400 * 0.023 RON = 331.2 RON  =>  final 629.28 RON
  gain 298 RON ~= 68 EUR/day
My machine - current performance

- 10 transactions / minute (1 transaction / 6 seconds)
  - max 14400 transactions / day
  - enter PIN, type challenge code, read response image, do OCR

- Our previous example:
  100 RON → 190 RON (gain ~20 EUR)
  => 4347 transactions * 6 sec/trans = \(26082\) sec
     = 7h:14m:42 s

- Maximum amount to multiply per day:
  14400 * 0.023 RON = 331.2 RON => final 629.28 RON
  gain 298 RON =~ 68 EUR/day

- What about doing in parallel (on multiple bank accounts)?
- Money making machine? 😊
External vs Internal instrumentation

- **Internal instrumentation (direct electrical connections):**
  - **Pros:**
    - more reliable and faster
    - almost error free
  - **Cons:**
    - might not be possible – some digipasses deactivate when opened
    - must know the pinout of LCD screen (lots of pins!)
    - sensitive soldering required
    - mistakes can lead to deactivation

- **External instrumentation:**
  - **Pros:**
    - No interference with digipass’s internals
    - Can be applied to any digipass model
  - **Cons:**
    - Pretty slow (but good for the “low and slow” approach)
    - Some (mechanics) errors occur on pressing buttons (resolvable by a more professional construction)
    - OCR process needs special (lighting) conditions to produce correct results
My machine – implementation details (1)
Optical Character Recognition

<table>
<thead>
<tr>
<th>Original</th>
<th>Cleared background</th>
<th>Blurred</th>
<th>Threshold applied</th>
<th>OCR-ized gocr / ocrad</th>
</tr>
</thead>
<tbody>
<tr>
<td>7169309</td>
<td>7169309</td>
<td>7169309</td>
<td>7169309</td>
<td>7169309_16g309</td>
</tr>
<tr>
<td>1757450</td>
<td>1757450</td>
<td>1757450</td>
<td>1757450</td>
<td>1757450_5_G50</td>
</tr>
<tr>
<td>043i_616</td>
<td>043i_616</td>
<td>043i_616</td>
<td>043i_616</td>
<td>043i__616</td>
</tr>
<tr>
<td>9236414</td>
<td>9236414</td>
<td>9236414</td>
<td>9236414</td>
<td>9a_641_4</td>
</tr>
</tbody>
</table>
Digipass automation

My machine – development stages
Live Demo
Q & A
Thank you!

Adrian Furtună, PhD, OSCP, CEH
adif2k8@gmail.com
http://pentest-tools.com