

# obfuscation nation

## **mobile application design**

week 10: Obfuscators. HTTP POST. Bluetooth

michael sharon

# Menu

- **Obfuscation in da Nation**
- **Tips'n'tricks: Graphics Edition**
- **Presentation**
- **Uploading thangs with HTTP POST**
- **Bluetooth**

# Class outline (new)

- Week 9: Obfuscation, HTTP POST, Bluetooth
- Week 10: WAP + XHTML MP
- Week 11: Python?
- Week 12: More Python?
- Week 13: Final Project Workshop AKA Open Questions / Issues
- Week 14: Final Project Presentations

Not the Phone of the Week

AKA

<http://vertu.com>

THE GPHONE COMETH!!!

# Obfuscation in da Nation

# Obfuscation

- AKA bytecode obfuscator
- What is it?
  - Software build tool
- What does it do?
  - Reduces the size of your class files
  - Makes it difficult to decompile your code
  - Removes unused/unnecessary methods

# Shrink, optimize, obfuscate

- Detect and remove unused classes, fields, methods, and attributes.
- Optimize bytecode and remove unused instructions.
- Rename the remaining classes, fields, and methods using short meaningless names.
- Resulting jars are smaller and harder to reverse-engineer. \* (thanks Proguard docs!)

# Smaller JAR files

- = reduced storage requirements
- = faster downloading of your application
- = faster loading
- & smaller memory footprints

# 3 Obfuscators

- **Proguard**

- <http://proguard.sourceforge.net/>

- **Retroguard**

- <http://www.retrologic.com/>

- **Jode**

- <http://jode.sourceforge.net/>

# Tips'n'tricks: Graphics edition

# Images/Icons

- PNG
  - Lossless competitor to GIF
  - Supports >256 colours and alpha
  - Only use indexed images! 8-bit or 4-bit
- Some support for transparency

# Resolutions

- Nokia Series 60 3rd Edition = 352x216 - definitely not traditional
- Dual portrait + landscape modes

# Resolutions

- 176w x variable height (mostly 208px) = current standard
  - Portrait orientation
  - Nokia Series 60 2nd Edition, Motorola RAZR, ROKR, PEBL, iDEN, Sony Ericsson
- 128w x variable height
  - Old standard
- 96x - even older, painful to resize artwork

# Tools

- Photoshop / Gimp / Pixel
- Debabelizer
- PNGCrush, PNGOut, TweakPNG
- Icon Studio

# Presentation

HTTP Post

# POST Process

1. Get yer **HttpConnection**
2. Add header fields - `setRequestMethod()`
3. Open the output stream  
(**openOutputStream()**) - sends headers
4. Format + send parameters + request body
5. Read the response

Exercise: Create a file uploader

# Useful things

- Variables
  - photo = actual file data
  - name = filename
- URLs
  - Upload: <http://uberthings.com/mobile/u.php>
  - Log file: <http://uberthings.com/mobile/test.log>

# More useful links

- Simple uploader: <http://uberthings.com/mobile/up.php.txt>
- File upload example: [http://mobilenin.com/pys60/ex\\_upload\\_image\\_to\\_url.htm](http://mobilenin.com/pys60/ex_upload_image_to_url.htm)
- If you want to still use the advanced upload script here: <http://uberthings.com/mobile/u.php> (text version - <http://uberthings.com/mobile/u.php.txt>)
- HTTP POST in Python: <http://aymanshamma.googlepages.com/httppostfrompys60tophp>