Fighting phishing the information-centric way

Nikos Fotiou, Giannis F. Marias and George C. Polyzos
Mobile Multimedia Laboratory,
Athens University of Economics and Business

Fighting phishing the traditional way

- Blacklist-based
  - Their performance is affected by the source of the (blacklisted) URL and its freshness
  - Cannot prevent all attacks
- Usage of host features (IP, WHOIS)
  - Can be bypassed using dynamic DNS or hosting services with high reputation
  - Often leads to false positives

Fighting phishing the traditional way (cont'd)

- Proactively by examining URL features (dots, length,…)
  - Can be bypassed using URL re-write, IFRAMES
- Proactively by examining content and by detecting "suspicious" terms
  - Can be bypassed using code obfuscation, images instead of text

```
<script language="javascript">
($=[$=[]][(__=!$+$)[_=-~-~-~$]+({}+$)[_/_]+
($$=($_=!''+$)[_/_]+$_[+$])]()[__[_/_]+__
[_+~$]+$_[_]+$$]("hello world")
</script>
```

And all these in order to…

- Decide that the site of the right image (phishing site) imitates the site of the left image (original) ….
  - … and this is not coincidence, it has to be like that in order to mislead users!

An information-centric approach

- Step 1 : Capture a screenshot of the site that the user visits (optim. if contains password field)
  - Easy in Chrome : chrome.tabs.captureVisibleTab
- Step 2: Store it in a meaningful way
  - Small in size
  - Allow comparisons between two images
    - ->Perceptual hashing
- Step 3: Decide if it is “similar enough” to an already stored image but from a different URL
  - In that case possible phishing

Perceptual hashing (PH)

- Let H(x) =y, then if x' is similar to x then H(x') =y or "close" to y
- It is impossible to construct a x' perpetual similar to x with H(x') (very different) to y
- When it comes to images y is some bytes
- 3 hash functions of the phash library are considered:
  - Discrete Cosine Transform based hash (DCT) 64bits
  - Marr-Hildreth Operator based hash(MH) 576bits
  - Radial Variance based hash(RAD) 320bits

Dissimilarity

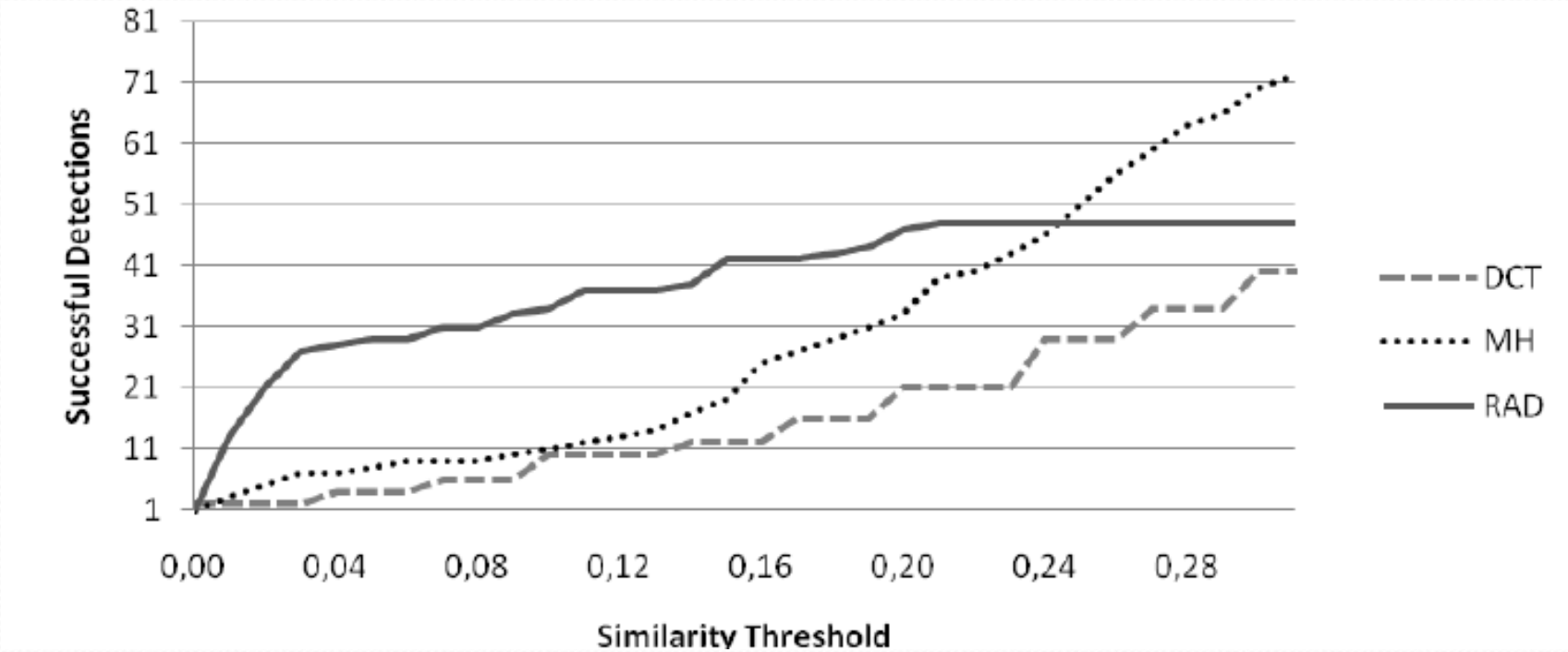
- The normalized hamming distance of two hashes
  - 0.0 = absolute the same, 1.0= completely different

Difference	DCT	MH	RAD
Original in Chinese	0.23	0.28	0.01
Original without area 1	0.13	0.03	0.01
Original without area 2	0.1	0.11	0.01
Original without area 3	0.1	0.07	0.01
Original without area 4	0.13	0.18	0.01
Original without areas 1 and 2	0.13	0.07	0.01
Original without areas 1 and 3	0.2	0.19	0.02
Original without areas 1 and 4	0.46	0.11	0.01

  - **Similarity Threshold**: A dissimilarity value s.t. if two screenshots dissimilarity is less than that, they belong to the same site

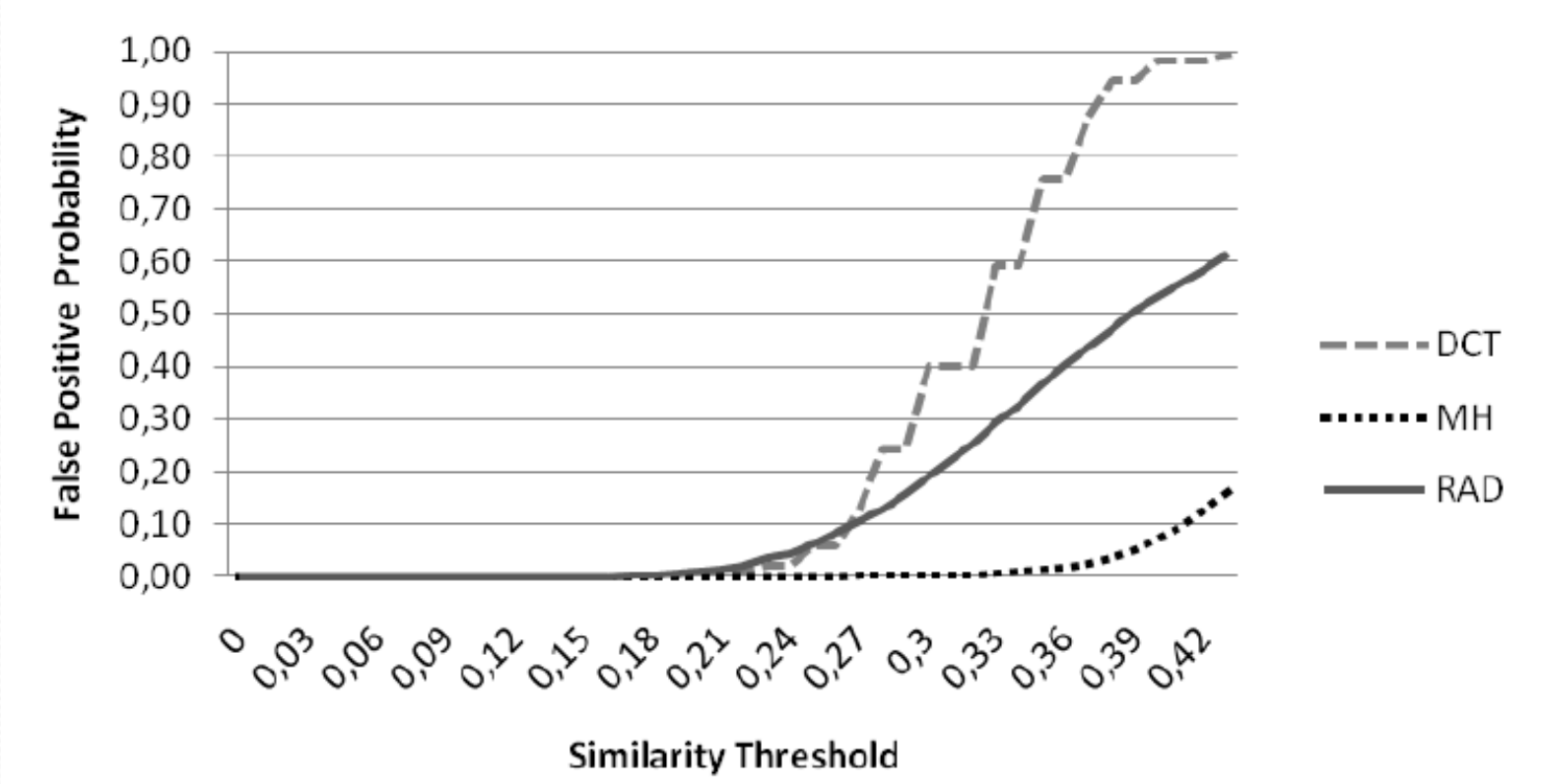
Phishing detection

- Phishtank, 100 **unique** phishing sites, in isolated server:
  - Chrome 12, IE 9, Netcraft anti-phishing tool bar: no detection



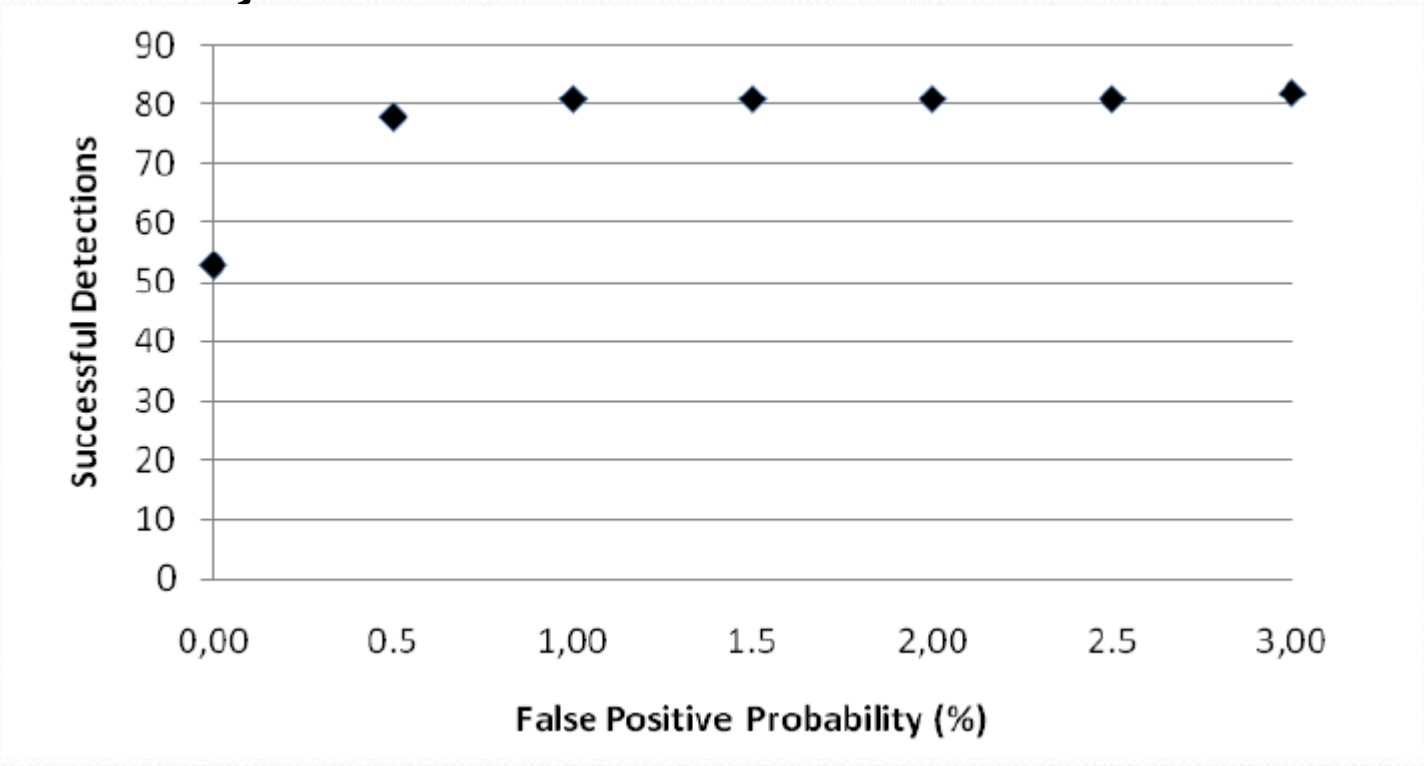
False positives

- Top 100 most visited sites in U.S (Google)



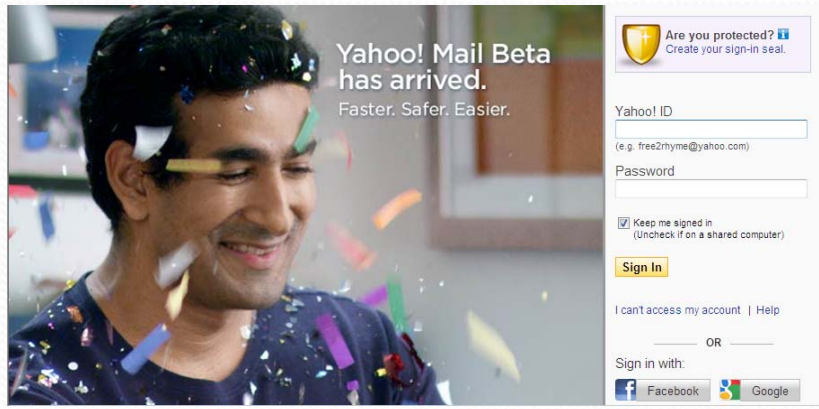
Cumulative performance

- Set the similarity threshold for each mechanism to the value that achieves the desired false positive probability



What went wrong?

- Some web sites change the login page every day (login form in main page, ads in login page)



- Multiple login pages, login pages in case of wrong username or password much different than the original pages
- Fake OpenID, Facebook Connect,.., sites

Thank you

fotiou@aueb.gr