



“Test by Proxy”

Mike Lee

3 Problems

- Rendering of “my customers content” is not controlled by me, so how do I test effectively ?
- How do I know our servers are coping?
- If all ISN'T OK.. how do we quickly find out WTF HAPPENED ?

A Suggestion

- Generate clientside logging to be used by test, monitoring & analysis scripts

A few Impediments

- JS security problem
- Multi-browser support
- Time (“we want it now”)

The Proposal

- ICAP proxy to intercept & enhance HTML
- Log to file through FF or Chrome
- R4 == RailsReads&RendersResults

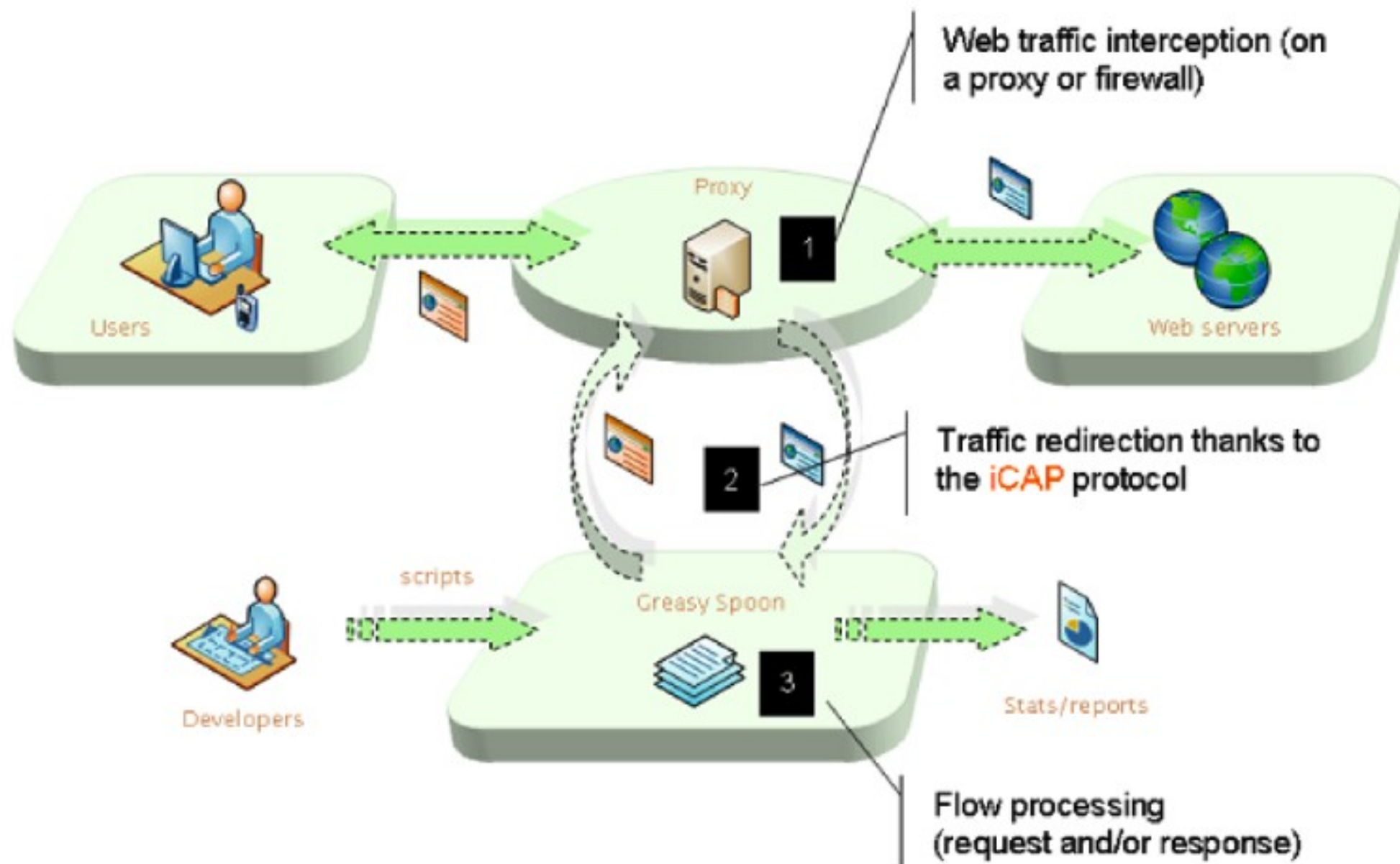
ICAP ?

ICAP, the Internet Content Adaption Protocol,
is a protocol aimed at providing
"simple object-based content vectoring for HTTP services".

ICAP is, in essence, a lightweight protocol for executing a **"remote procedure call"** on HTTP messages. It allows ICAP clients to pass HTTP messages to ICAP servers for some sort of transformation or other processing ("adaptation"). The server executes its transformation service on messages and sends back responses to the client, usually with modified messages. Typically, the adapted messages are either **HTTP requests or HTTP responses**.

[http://www.icap-forum.org/
documents/specification/rfc3507.txt](http://www.icap-forum.org/documents/specification/rfc3507.txt)

Setup...



You will need...

- **Operating Sys:** Linux
- **ICAP Client:** <http://www.squid-cache.org/Download/>
- **ICAP Server:** <http://sourceforge.net/projects/greasyspoon/>
- **Runtime:** Java (SE Runtime Env.)
- **Scripting:** JRuby
- **Browser:** Firefox

Greasyspoon

```
require 'hpricot'
```

```
def process(httpresponse)  
  document = Hpricot(httpresponse)
```

```
  pfhead1 = '<script language="JavaScript">  
function peepingfrog() {alert("Peeping !");} </script>'  
  (document/"html/head").append pfhead1
```

```
  pfhead2 = '<script language="JavaScript">  
function peep() {allloaded = 0; while(allloaded < 1)  
{ var imagesloaded = 1; var images = document.images;for (var i = 0;i<images.length;i++)  
  { if(images[i].complete == false) { imagesloaded = 0; } } allloaded = imagesloaded; }  
window.location = "file://pf.txt"; } </script>'  
  (document/"html/head").append pfhead2
```

```
  (document/"html/body").set(:onload => "peep()")  
  pfbody = '<div> </div>'  
  (document/"html/body").append pfbody  
  return "#{document}"  
end
```

```
$httpresponse = process($httpresponse)
```



End of part 1

• Comments ?

• Questions ?

• Suggestions ?

• mike.lee@europe.com

