A Truly Transparent Proxy

Using Squid and Linux
Need for a HTTP proxy

- Bandwidth in South Asia is extremely expensive
- HTTP still comprises a significant proportion of the total traffic
- About 20% to 30% of this traffic is cachable
- A 20% cache in HTTP traffic is roughly equivalent to 10% saving in total traffic
### Some statistics

<table>
<thead>
<tr>
<th>Traffic</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Cached kBytes/ Hour</td>
<td>32554.13</td>
</tr>
<tr>
<td>Direct kBytes/ Hour</td>
<td>98239.13</td>
</tr>
<tr>
<td>Total kBytes/ Hour</td>
<td>130793.26</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Summary</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Hit Rate (% URL)</td>
<td>40.46</td>
</tr>
<tr>
<td>Bandwidth savings %</td>
<td>24.89</td>
</tr>
<tr>
<td>Apparent Speed increase</td>
<td>22.9</td>
</tr>
</tbody>
</table>

These statistics are for a small ISP with about 200 users
Transparent proxies

- Users need not configure anything at their end
- No need to worry about various browsers
- User cannot bypass the proxy easily
- Need for user support is much less
- ISP can control web access if required
Typical ISP network
Why a truly transparent proxy?

• Cost savings due to proxy at remote POP
• The proxy however is **NOT** transparent to the bandwidth control system.
• Many ISPs prefer simple bandwidth control devices
• These are quite expensive.
• A truly transparent proxy solves these issues.
HOWTO - I

- First download the Linux kernel from http://www.kernel.org/

- Download the ctt-proxy patch from http://www.balabit.com/downloads/tproxy/linux-2.4/devel/ and apply it as per the instructions in the README file.

- Recompile and reboot into the new kernel
HOWTO - II

• Download the squid source from http://www.squid-cache.org/

• Patch Squid with the patch from
  − http://www1.nl.squid-cache.org/mail-archive/squid-dev/200404/att-0032/squid-2.5-cttproxy-04JES.diff

• Compile and install.

• Configure squid
HOWTO – III

• Set in squid.conf:
  - tcp_outgoing_address
  - httpd_accel_host
  - httpd_accel_port
  - httpd_accel_with_proxy off
  - httpd_accel_uses_host_header
  - linux_tproxy
Configuring the OS

- Set ip spoofing on:
  - `sysctl -w net.ipv4.ip_nonlocal_bind=1`
- Set IP forwarding on:
  - `sysctl -w net.ipv4.ip_forward=1`
- Set the transparent firewall rule with the TPROXY module:
  - `iptables -t tproxy -A PREROUTING -j TPROXY -- on-port 3128`
Cost benefits

• Cisco Cache costs about 2000 to 5000 USD
• Most such devices have costs in the same range
• A Linux box needs a simple PC with two network cards
• A good system will cost about 600 USD
Other benefits

- The PC can be configured as a bridge
- The PC can do bandwidth control as well
- Optionally, the ISP can provide limited firewalling here, reducing the load on the router
- This system can scale considerably well, particularly with smaller nodes.
Limitations

• The proxy needs to be in the direct physical path to be effective.

• The patch maintains TCP session state. This can prove to be a limiting factor with large numbers of users.

• The fix is to disable the stateful code in the kernel patch.