BBQ

Task target is to demonstrate knowledge in network programming, ability to find and apply third-party libraries and to read and understand published RFC's,

Task summary:

Write a simple client-server application with server push events, that passes application layer HTTP proxies by using HTTP long poll.

Task description:

Basics

Write a very simple client and server console application. Client will talk to server over socket connection with extremely simple protocol, named *BBQ*. BBQ protocol consist of client requests *"I AM HUNGRY, GIVE ME BBQ"*, *"NO THANKS"*, *"I TAKE THAT!!!"*. Server responses are *"OK, WAIT"*, *"CLOSED BYE"*, *"SERVED BYE"*, server pushes are *"CHICKEN READY"*, *"BEEF READY"*, *"LAST MONTH MAMMOTH READY"*. Design a meaningful state transition out of this command set.

Server need not accept more than one client at a time (hard to do this in ACE, but its OK). Client does not need CLI interface, but needs to log communication on the console.

Advanced

Client and server features from *Basics*, extended to go over HTTP application level restrictive proxy. Application level restrictive proxy will let you pass through it, but will not be happy with BBQ protocol. BBQ protocol must be embedded in HTTP requests body. HTTP parser might not be written for this tasks - grab some third party lib. Implement long poll over HTTP proxy - see *References*.

References:
1) Server push and long poll:

http://en.wikipedia.org/wiki/Push_technology

2) HTTP

http://www.w3.org/Protocols/rfc2616/rfc2616.html

3) Application layer Firewall

http://en.wikipedia.org/wiki/Application_layer_firewall

Restrictions:
HTTP Proxy that will be used to test task execution will be squid, without Application Layer restrictions, but client-server traffic will be sniffed.