Scalable Server

The purpose of this program is to test the scalability of various networking techniques.

Compilation

SELECT

Compilation is fairly standard, however, with the select server, the Go compiler must be modified. The general instructions for installing the Go compiler from source can be found at https://golang.org/doc/install/source . For this test FdSet.Bits was changed from being an [16]int64 to an [2048]int16. While the original select call as implemented by go only handles 1024 connections the new implementation can handle. Once this has been completed select can monitor 131,072 files, well more than what is used in this test.

Once the above has been completed, the scalable server can be compiled with the following

```
$ cd $GOPATH/src/select_scalable_server
$ go install
```

EPOLL

Epoll is much simpler, if desired it may be compiled without modifying the go compiler.

```
$ cd $GOPATH/src/epoll_scalable_server
$ go install
```

MULTITHREADED

Multithreaded can be compiled similar to the above.

```
$ cd $GOPATH/src/multithreaded_scalable_server
$ go install
```

CLIENT

The client is also a similar compilation

```
$ cd $GOPATH/src/client_scalable_server
$ go install
```

Usage

In the documentation, go suggests setting $GOPATH/bin to part of your path. By doing this the programs can be run by doing the following

CLIENT

```
$ client_scalable_server –i 100 –r 100 –d 1000 –c 10000 x.x.x.x:PORT
```

SERVERS

```
$ multithreaded_scalable_server :PORT
$ select_scalable_server PORT
$ epoll_scalable_server PORT
```