

Go 语言实现 http 共享 (带 trace)

我之前有篇文章

(<http://www.cnblogs.com/MikeZhang/archive/2012/03/13/httpShareGolang20120312.html>) 中提到过用 Go 语言实现 http 文件共享，这个版本的程序比 python 的实现快了点，默认情况下支持的客户端多了些，但是没有客户访问的 trace，程序运行过程中，感觉像是死掉了。我想改进下，让它有 trace。

代码如下：

```
/*
File      : httpShareWithTrace.go
Author    : Mike
E-Mail    : Mike_Zhang@live.com
*/
package main
import(
    "fmt"
    "net/http"
    "io/ioutil"
    "log"
    "time"
    "os"
    "strings"
)

func getFilelist(path string) string{
    m_files,err := ioutil.ReadDir(path)
    if err !=nil{
        // println( "Get filelist error !" )
        return ""
    }
    var strRet string
    for _,f := range m_files {
        // println(f.Name(),f.IsDir())
        if path == "./" {
            strRet += "<p><a href=\""+path+"\"+f.Name()
+ \" \">>\" + f.Name() + "</a></p>"
        }else{
            strRet += "<p><a href=\""+path[1:]+\"/\"+f.Name()
+ \" \">>\" + f.Name() + "</a></p>"
        }
    }
    return strRet
}
```

```

}

func Handler( w http.ResponseWriter, r *http.Request ){
    println("Request ", r.URL.Path, " from ", r.RemoteAddr)
    // path := r.URL.Path[1:]
    path := "." + r.URL.Path
    if path == "./favicon.ico" {http.NotFound(w, r); return}
    if path == "/" || getFilelist(path) != "" {fmt.Fprintf( w, "%s", getFilelist(path)); return}
    fin, err := os.Open(path)
    defer fin.Close()
    if err != nil {fmt.Fprintf( w, "404 : Not found" ); return}
    readLen := 1024 * 1024
    buf := make([]byte, readLen)
    startPos := 0
    println("Transfer file ", path, " ... ")
    for{
        n, err := fin.ReadAt(buf, int64(startPos))
        fmt.Fprintf(w, "%s", buf[:n])
        if 0 == n || err != nil {break}
        startPos += readLen
    }
}

func main(){
    port := "8080" //Default port
    if len(os.Args) > 1 { port = strings.Join(os.Args[1:2], "")}
    http.HandleFunc( "/", Handler)
    s := &http.Server{
        Addr:           ":",
        ReadTimeout:     1 * time.Hour,
        WriteTimeout:    1 * time.Hour,
        MaxHeaderBytes: (1 << 31) - 1, //Max file size is 2048M
    }
    println("Listening on port ", port, "...")
    log.Fatal(s.ListenAndServe())
}

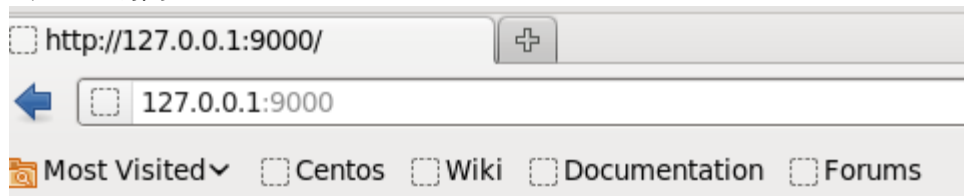
```

运行效果如下：

1、启动 http 文件共享

```
mike@localhost: ~/proj/0.5/httpShare
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
[mike@localhost httpShare]$ ./httpShareWithTrace 9000
E-Mail : Mike_Zhang@live.com
```

2、web 访问



[httpShareWithTrace](#)

[httpShareWithTrace.go](#)

E-Mail : Mike_Zhang@live.com

3、后台 trace

```
[mike@localhost httpShare]$ ./httpShareWithTrace 9000
Listening on port 9000 ...
Request /httpShareWithTrace.go from 127.0.0.1:42986
Transfer file ./httpShareWithTrace.go ...
E-Mail : Mike_Zhang@live.com
```

说明：最大支持 2G 文件的下载，限时为 1 个小时，这里没有充分使用 http 协议，直接用文件 io 做的。时间有限，这里暂时达到了预期功能，够局域网使用，这个等以后有时间了做进一步的优化。