## 来炜 [laiwei.ustc@gmail.com](mailto:laiwei.ustc@gmail.com) 2009-12

## 实现目标

1. 使用多线程传输，加速下载速度
2. 数据在传输过程中,进行流式md5计算，避免在传输完毕之后校验大文件
3. 支持断点续传
4. 支持http协议和ftp协议
5. 代码尽可能的简单，利于维护

## 系统依赖

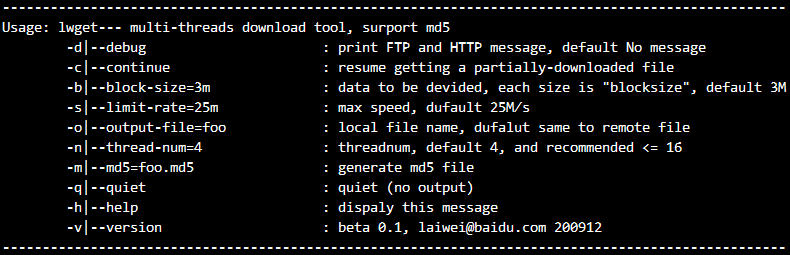
1. perl,版本>=5.8.5
2. linux or windows

## 目前实现的功能

* 支持FTP和HTTP下载
* 支持断点续传（HTTP and FTP）
* 多线程下载加速（线程数目可以任意指定）
* 支持限速
* 下载的同时进行流式的md5计算，现在完毕即可生成整个文件的md5（可以使用参数指定是否需要计算md5）
* 命令行参数选项类似于wget
* 跨平台(windows && linux)

## 使用示例

* ./lwget -v 输出帮助信息



* 命令行参数解释
  + -d|--debug : 输出debug信息，比如FTP或者HTTP的连接信息
  + -c|--continue: 是否需要续传
  + -b|--block-size : 指定数据分块大小，默认为3M，测试得出性价比较高
  + -s|--limit-rate： 最大速度,默认为25M/s
  + -o|--output-file : 下载到本地的文件命名,不指定默认采用源文件名
  + -n|--thread-num：下载线程数目，默认为4，建议保持在16以下
  + -m|--md5 : 指定是否计算文件md5,-m表示计算md5,且md5文件的命名为输出文件.md5;-m foo.md5 则指定md5文件名称为foo.md5
  + -q|--quiet : 屏幕不打印信息,类似于wget -q参数
  + -h|--help : 输出帮助信息
  + -v|-- version : 打印版本信息
* 使用范例
  + ./lwget <http://host:8080/PPT.rar>

按照默认参数通过http下载PPT.rar

* + ./lwget ftp://host/home/work/bid.tar

按照默认参数通过ftp协议匿名下载bid.tar文件

* + ./lwget <ftp://ftp:ftp@host/home/bid.tar>

指定用户名和密码下载bid.tar文件

* + ./lwget <ftp://ftp:ftp@host/home/bid.tar> -m -n 8

以8个线程下载bid.tar,并且边下载边计算md5,md5文件命名为bid.tar.md5

* + ./lwget <http://host:8080/PPT.rar> **-o ppt2.rar** -**m other.md5** **-n 5**

以5个线程下载PPT.rar,本地保存为ppt2.rar,生成的md5命名为other.md5

* + ./lwget <http://host:8080/PPT.rar> **-s 100k**

下载速度限速为100k/s

## 性能测试 (下游 上游机器都为dell 2950)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 情景 (FTP) | 文件大小 | 速度 | 下游负载(最低idle) | 上游负载(最低idle) |
| 单线程  计算md5 | 1G | 1.4m/s | cpu\_idle:98 | cpu\_idle:99 |
| 4线程  计算md5 | 1G | 6.2m/s | cpu\_idle:94 | cpu\_idle:97 |
| 8线程  计算md5 | 1G | 12.6m/s | cpu\_idle:91 | cpu\_idle:97 |
| 12线程  计算md5 | 1G | 18.6m/s | cpu\_idle:87 | cpu\_idle:97 |
| 16线程  计算md5 | 1G | 22m/s | cpu\_idle:84 | cpu\_idle:96 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 情景 (FTP) 大文件 | 文件大小 | 速度 | 下游负载(最低idle) | 上游负载(最低idle) |
| 单线程  计算md5 | 11G | 1.3m/s | cpu\_idle:98 | cpu\_idle:93 |
| 4线程  计算md5 | 11G | 6m/s | cpu\_idle:94 | cpu\_idle:97~80 |
| 8线程  计算md5 | 11G | 10.8m/s | cpu\_idle:91 | cpu\_idle:65 |
| 12线程  计算md5 | 11G | 17.6m/s | cpu\_idle:86 | 前60%:cpu\_idle:97  **后40%:cpu\_idle:50** |
| **16线程**  **计算md5** | **11G** | **21m/s** | **cpu\_idle:85** | **cpu\_idle:40** |

说明:

1. 采用HTTP协议下载对上下游的消耗更少
2. FTP协议下载时,多线程(10个线程以上),大文件(10G左右)会对上游造成较大的压力(cpu\_idle降到50%),对下游无特别影响
3. 如果不计算md5,对下游的压力会变的更小
4. 在高速网络中,性能不如wget,比如jx到tc,wget可以下载到100m/s,而lwget只能达到80m/s的速度,且系统的负载要比使用wget高
5. 限速在20M/s以下的时候,和wget的性能比较接近.

## 优缺点

* 优点
  + 在低速网络中,下载速度很快(很容易的维持在20m/s)
  + 边下载边计算md5,避免了在文件下载完之后计算md5耗时耗资源(11G的文件做md5校验耗时在5分钟左右,使用lwget可以免去这个时间和消耗)
  + lwget代码简洁(只有19K),易于维护
  + 很方便的支持续传(FTP 和 HTTP)
  + 可以在windows下直接使用
* 缺点
  + 在高速下载的时候,性能不如wget
  + 不支持整个目录下载
  + 多线程下载大文件时(10多个线程,10G大文件),对上游消耗较大