

Coderholic

Addicted to Development

Ben Dowling is a software engineer who has launched sites such as [Do Nothing for 2 Minutes](#), [BusMapper](#) and [Geomium](#). He also co-organises the monthly [Hacker News London meetup](#).

You can follow his updates on [twitter](#), his development projects on [GitHub](#), or buy him a book from his [Amazon Wish List](#).

 [Subscribe](#)

 [Follow on Twitter](#)

Recent Posts

- [Invaluable command line tools for web developers](#)
- [Hacker News London Meetup: A Year On](#)
- [Scraping the web with Node.io](#)
- [A New Blog for Mobile Developers](#)
- [Attend events or stay in and work hard?](#)
- [12 Months are up](#)
- [Extending VLC with Lua](#)
- [A brief look at the Fantom programming language](#)
- [django-cities: Countries, Regions, Cities and Districts for Django](#)
- [Django deployment with virtualenv pip and fabric](#)

Popular Posts

- [Blackberry WebBitmapField](#)
- [12 New Programming Languages in 12 Months](#)
- [Firefox 4k XML node limit](#)
- [PyWebShot - Generate website thumbnails using Python](#)
- [Invaluable command line tools for web developers](#)
- [10 Free Python Programming Books](#)
- [Blackberry Custom Button Field](#)
- [django-cities: Countries, Regions, Cities and Districts for Django](#)
- [10 More Puzzle Websites to Sharpen Your Programming Skills](#)
- [25 Free Computer Science Books](#)
- [Signs Banners](#)
- [ContractEdge](#)
- [Web Design School](#)
- [BRS Labs](#)

Invaluable command line tools for web

developers

August 13, 2011

Life as a web developer can be hard when things start going wrong. The problem could be in any number of places. Is there a problem with the request your sending, is the problem with the response, is there a problem with a request in a third party library you're using, is an external API failing? There are lots of different tools that can make our life a little bit easier. Here are some command line tools that I've found to be invaluable.

Curl

Curl is a network transfer tool that's very similar to wget, the main difference being that by default wget saves to file, and curl outputs to the command line. This makes it really simple to see the contents of a website. Here, for example, we can get our current IP from the ifconfig.me website:

```
$ curl ifconfig.me
93.96.141.93
```

Curl's `-i` (show headers) and `-I` (show only headers) option make it a great tool for debugging HTTP responses and finding out exactly what a server is sending to you:

```
$ curl -I news.ycombinator.com
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Cache-Control: private
Connection: close
```

The `-L` option is handy, and makes Curl automatically follow redirects. Curl has support for HTTP Basic Auth, cookies, manually settings headers, and much much more.

Siege

Siege is a HTTP benchmarking tool. In addition to the load testing features it has a handy `-g` option that is very similar to `curl -iL` except it also shows you the request headers. Here's an example with www.google.com (I've removed some headers for brevity):

```
$ siege -g www.google.com
GET / HTTP/1.1
Host: www.google.com
User-Agent: JoeDog/1.00 [en] (X11; I; Siege 2.70)
Connection: close
```

```
HTTP/1.1 302 Found
Location: http://www.google.co.uk/
Content-Type: text/html; charset=UTF-8
Server: gws
Content-Length: 221
Connection: close
```

```
GET / HTTP/1.1
Host: www.google.co.uk
User-Agent: JoeDog/1.00 [en] (X11; I; Siege 2.70)
Connection: close
```

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=ISO-8859-1
X-XSS-Protection: 1; mode=block
```

Connection: close

What siege is really great at is server load testing. Just like `ab` (apache benchmark tool) you can send a number of concurrent requests to a site, and see how it handles the traffic. With the following command we test google with 20 concurrent connections for 30 seconds, and then get a nice report at the end:

```
$ siege -c20 www.google.co.uk -b -t30s
...
Lifting the server siege...      done.
Transactions:                   1400 hits
Availability:                   100.00 %
Elapsed time:                   29.22 secs
Data transferred:              13.32 MB
Response time:                 0.41 secs
Transaction rate:              47.91 trans/sec
Throughput:                    0.46 MB/sec
Concurrency:                   19.53
Successful transactions:       1400
Failed transactions:           0
Longest transaction:           4.08
Shortest transaction:          0.08
```

One of the most useful features of siege is that it can take a url file as input, and hit those urls rather than just a single page. This is great for load testing, because you can replay real traffic against your site and see how it performs, rather than just hitting the same URL again and again. Here's how you would use siege to replay your apache logs against another server to load test it with:

```
$ cut -d ' ' -f7 /var/log/apache2/access.log > urls.txt
$ siege -c<concurrency rate> -b -f urls.txt
```

Ngrep

For serious network packet analysis there's [Wireshark](#), with it's thousands of settings, filters and different configuration options. There's also a command line version, `tshark`. For simple tasks I find wireshark can be overkill, so unless I need something more powerful, `ngrep` is my tool of choice. It allows you to do with network packets what `grep` does with files.

For web traffic you almost always want the `-W` `byline` option which preserves linebreaks, and `-q` is a useful argument which supresses some additional output about non-matching packets. Here's an example that captures all packets that contain GET or POST:

```
ngrep -q -W byline "(GET|POST) ."
```

You can also pass in additional packet filter options, such as limiting the matched packets to a certain host, IP or port. Here we filter all traffic going to or coming from google.com, port 80, and that contains the term "search".

```
ngrep -q -W byline "search" host www.google.com and port 80
```

[View Comments](#)

Like



and 18 others liked this.

Add New Comment

[Login](#)



Showing 21 comments

Sort by popular now ▾



ViewDNS.info

There's a collection of useful web-based tools as well at <http://viewdns.info/> that are quite useful.

[1 day ago](#) [3 Likes](#)

[Like](#) [Reply](#)



Michael Bianco

Never heard of siege before, just installed via brew. Thanks for the tip.

[1 day ago](#) [3 Likes](#)

[Like](#) [Reply](#)



Ross Anthony

What is 'brew'?

[1 day ago](#) [in reply to Michael Bianco](#)

[Like](#) [Reply](#)



Chris Thomson, CS student, software junkie, javascript & ruby enthusiast, o...

It's a package manager for OS X: <https://github.com/mxcl/homebr...>

[1 day ago](#) [in reply to Ross Anthony](#)

[Like](#) [Reply](#)



Ross Anthony

Thanks Chris, but don't have a mac at home unfortunately, only at work. Looks identical to linux command line though, will see if I can get these installed and working in Ubuntu.

[1 day ago](#) [in reply to Chris Thomson](#)

[Like](#) [Reply](#)



Anonymous

I think it's important to note that the -I option to curl doesn't really mean "show only header." Whereas -i means "include the header in the output", -I actually sends a HEAD

request to the server as opposed to another method like GET. It's not uncommon for applications to respond differently to a HEAD request than they might to a GET-- although, strictly speaking, I don't think they should.

1 day ago 1 Like

Like Reply



Ben Dowling

Good point, thanks for mentioning it. Strictly speaking you should get the same headers from a HEAD request as you do from a GET request, but that won't necessarily always be the case.

1 day ago in reply to Anonymous

Like Reply



Miki

I find piping JSON output via "python -m json.tool" useful as well.

6 hours ago

Like Reply



Ben Dowling

Awesome tip, didn't know this one! More about it here: <http://docs.python.org/library...>

6 hours ago in reply to Miki

Like Reply



DAddYE, Padrino developer, Partner at LipsiaSOFT

Excellent article, thanks especially for point me on siege.

23 hours ago

Like Reply



Brian Morearty

Thanks for these tips. These tools look great.

You might want to correct the last ngrep example: insert the word "and" just before "port 80". I couldn't get it to work until I did that.

1 day ago

Like Reply



Ben Dowling

Thanks - fixed now.

1 day ago in reply to Brian Morearty

Like Reply



Guillermo Garron

The last use of `siege` you showed is really a good one.
I did not know about it.

1 day ago

Like Reply



Anonymous

scapy is great for packet manipulation. It is however kind of advanced with a steep learning curve but it works great.

1 day ago

Like Reply



Ivan Malopinsky

ab is a similar program to siege: <http://httpd.apache.org/docs/2...>

very useful for benchmarking load performance.

1 day ago

Like Reply



Justinavery.me

I like the performance benchmark testing tool there.

Another good one to know about is wget to scrape a site.... For legitimate reasons only of course.

1 day ago

Like Reply



recluze

Great tools. I'd never heard of siege before either. How about sharing some more.

1 day ago

Like Reply



mog_man

an invaluable tool for web developers is GOOGLE. Use it! :)

[1 day ago](#) [in reply to recluze](#)

[Like](#) [Reply](#)



Rich Jones - Gun.io

If you like NGrep, you might also like NetCat: <http://netcat.sourceforge.net/>

[1 day ago](#)

[Like](#) [Reply](#)



Ben Dowling

Yeah netcat is a great tool - I should probably have mentioned that here too!

[1 day ago](#) [in reply to Rich Jones - Gun.io](#)

[Like](#) [Reply](#)



Dave

Curl is more than enough for me. I didn't know about the others. Cheers!

[1 day ago](#)

[Like](#) [Reply](#)

[✉ Subscribe by email](#) [RSS](#)

Reactions



[Show more reactions](#)

[blog comments powered by DISQUS](#)

Categories

- [apache](#)
- [blackberry](#)
- [books](#)
- [C](#)
- [C#](#)
- [css](#)
- [image processing](#)
- [java](#)
- [javascript](#)
- [jQuery](#)
- [linux](#)
- [mobile](#)
- [mysql](#)
- [objective-c](#)
- [PHP](#)

- [programming](#)
- [python](#)
- [scripts](#)
- [software design](#)
- [startups](#)
- [web dev](#)

Pages

- [Guest Posts](#)
- [PyRadio](#)

Recent Posts

- [Invaluable command line tools for web developers](#)
- [Hacker News London Meetup: A Year On](#)
- [Scraping the web with Node.io](#)
- [A New Blog for Mobile Developers](#)
- [Attend events or stay in and work hard?](#)
- [12 Months are up](#)
- [Extending VLC with Lua](#)
- [A brief look at the Fantom programming language](#)
- [django-cities: Countries, Regions, Cities and Districts for Django](#)
- [Django deployment with virtualenv pip and fabric](#)

Demos [JavaScript Tree Generator](#) [JavaScript Boids](#)

Links [@coderholic on Twitter](#) [GitHub](#) [Plasis Innovation](#) [Plasis Jobs: Freelance Job Site](#)