

KBeezie There's no place like ::1

Nginx as a Proxy to your Blog

Some time ago, back when Google Biz Ops were really popular I've had quite a number of clients use this method of changing the IP address of their auto-blogs. I once had a server hosted by ServerPronto*. The server was physically located in Panama (despite being advertised as a US hosting provider), the IPs were cheap though.

The Idea

The client had a good number of auto-blogs, landing pages and so forth on the same hosting provider (in this case his own dedicated server with multiple non-class-c IPs). The problem was making one or more of those domains appear not to be on the same server, or datacenter in this case. The solution was to use various class-c and class-b IPs on my own server, and use Nginx to proxy any web request to his own existing websites thus making those sites appear on a server in Panama as opposed to their actual location. The same could work if you wanted to use US IPs on a cheap provider and proxy to an off shore provider with cheaper hardware.

The Solution

For this we'll need a Dedicated server or VPS to host Nginx on. (I'll have an article later on how to install Nginx from scratch). You'll probably want a couple extra IPs on hand, in case you want to proxy multiple sites of the same niche.

For every hostname you wish to proxy you'll need a server block similar to this in your nginx configuration.

```
server {
    #Turns access log off
    access_log off;

    #Set this to the IP on the nginx server you want to represent this proxy
    listen nginx-ip-here:80;

    #Matches the requested hostname
    server_name your-hostname.com www.your-hostname.com;

    location / {
        # Tells the destination server the IP of the visitor (some take X-Real-IP,
        # some take X-Forwarded-For as their header of choice
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $remote_addr;
        # Tells the destination server which hostname you want
        proxy_set_header Host $host;
        # Performs a Proxy Pass to that server
        proxy_pass http://ip-address-of-your-server;
    }
}
```

From here you would point your domain name to the Nginx server. If your domain currently uses your existing hosting's nameservers, the easiest way to change this would be to use your registrar's DNS such as namecheap or godaddy (via total domain control), or a free DNS provider such as everydns.com. In any of those you would simply change the A record of your domain name to the IP address of the Nginx server.

Once that is setup any request on port 80 (standard http port) will be proxied to the server actually hosting the website or blog.

The Problems

A couple of downsides may exist with this method.

Control panel access via your domain name

When you point your domain to the Nginx IP, you not only direct web traffic there, but all other forms of traffic such as SSH, Cpanel/whm, FTP, and so forth. While you can in theory forward via iptables and other methods for those additional ports, it would be difficult and not without problems. So to get around this you'll either want to access SSH, Cpanel/DirectAdmin, FTP, and so forth via your actual server's IP address, or use a non-proxied domain. You can use any of your other domains on the same server especially if using a control panel, since the login is what determines which account you need access to, not the domain itself.

Emails

Email access of course will cease if you don't setup Nginx as mail proxy to that specific domain name. Though the work around would be to add an A record for mail.yourdomain.com in the DNS settings, and then change your MX records to point to mail.yourdomain.com. Course the better alternative if you must have email at that domain would be to use google apps, that way a simple whois or DNS lookup won't reveal your actual server's IP address. The best alternative is to simply not use email at proxied domains.

Speed

Normally a visitor takes their provided route from their location directly to your server over whatever backbone connects them. But with a proxied domain the visitor is then sent to the nginx server where ever it may be, and then the time is increased by how long it takes for the nginx server to communicate with your own webserver and then back. If the latency between the Nginx server and your destination is short, then the speed difference to visitors may not even be noticeable.

Security Concerns

Some of you no doubt are looking at this configuration and wondering if it could be changed so you don't have to make a server block for each hostname you have. While it is possible to make `server_name` and the `proxy_pass` destination dynamic to the requested hostname, this is a very bad idea. Doing so will allow not only your own domains to be proxied but any domain pointed to your nginx server. This can result in abuse or even bandwidth overload.

For example, don't do this:

```
server {
    access_log off;
    listen nginx-ip-here:80;

    location / {
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $remote_addr;
        proxy_set_header Host $host;
        proxy_pass http://$host;
    }
}
```

While it may be easy for you, it opens your server up for a floodgate of abuse using your server as some kind of public proxy point. Instead to tighten security of the nginx box, we should deny (or redirect) any traffic to the nginx server that either does not provide a host name, or does not provide one that has been defined. You can do so like this:

```
server {
    #visits not supplying hostname
    server_name _;
    location / { return 444; }
}
```

The return 444, sends a forbidden error back to the browser.

* ServerPronto: While I did use them at one point of time for the proxy purpose, I would not recommend them for a hosting provider or dedi reseller.

- [Previous Entry: SSL: Untrusted Connection in Firefox \[http://kbeezie.com/untrusted-connection-firefox/\]](http://kbeezie.com/untrusted-connection-firefox/)
- [Next Entry: Generating Ioncube Licenses \[http://kbeezie.com/generating-ioncube-licenses/\]](http://kbeezie.com/generating-ioncube-licenses/)

Posted in [Blackhat](#) , [Nginx](#) , [Wordpress](#)

Tags: [blog](#) [class-c](#) [IP](#) [Nginx](#) [proxy](#) [proxying](#)

You can follow any responses to this entry through the [RSS 2.0 Feed](#) . Both comments and pings are currently closed.

2 comments

[Add your comment](#)



BHD says:

2010/07/23 at 17:57

I did this a few months ago to try and be cheap breaking up the IP's on a server. The speed issue was killing me until I moved to a slightly better server.



Justin says:

2010/07/24 at 07:27

Nice post, you've got a new subscriber.