WebProxyCacheSetup

While the current administration guide is a great place to start when configuring Apache Traffic Server,
there are enough "gotchas" that I thought I'd contribute back to the project and document them.

Please keep in mind the following only applies to creating a forward-only web proxy caching setup.

My personal goal here was to replace Squid with Traffic Server as a "drop-in" replacement.

The following lists the initial steps involved in getting a generic Traffic Server install up and running.

NOTE: Please use the following with Apache Traffic Server v5.0.0 and higher.

IP Address Listening And Ports

Unlike Apache HTTP Server, Traffic Server takes a little more work to get things up and running.
The following settings are all located in the main configuration file, which by default is /usr/local/etc/trafficserver/records.config.

Specifically, the following directive should be set unless you want Traffic Server listening on every possible interface:

```
LOCAL proxy.local.incoming_ip_to_bind STRING [2601:d:4880:6c3:426c:8fff:fe3a:43f1]
```

Also, the next directive will tell Traffic Server which ports to listen on:

```
CONFIG proxy.config.http.server_ports STRING 8080:ipv6
```

In this example, Apache Traffic Server will now listen on my home machine's public IP, port 8080 for IPv6 only.

I was originally using localhost, but after looking at the HTTP proxy headers that ATS produced, I decided to be more specific.

DNS Round-Robin

Unlike many applications, the default in ApachTraffic Server is to actually round-robin requests among your configured DNS servers.

I didn't like this much, so I disabled it.

```
CONFIG proxy.config.dns.round_robin_nameservers INT 0
```

Required Remapping

The Apache Traffic Server default install configures URL re-mapping as required.

This will not allow you to use trafficserver as a foward proxy until you disable it in records.config file or configure remapping specifically for your needs.

```
CONFIG proxy.config.url_remap.remap_required INT 0
```

IP-based Access Control List

To setup basic security in your Traffic Server install, you'll have to configure a different file, by default /usr/local/etc/trafficserver/ip_allow.config.

If you've ever done firewall work the theory is very similar...simply list to Traffic Server what is allowed, followed by what is NOT allowed.

```
# Allow anything on localhost (this is the default configuration based on the
# depricated CONFIG proxy.config.http.quick_filter.mask INT 0x482)
src_ip=2601:d:4880:6c3:426c:8fff:fe3a:43f1        action=ip_allow method=ALL
# Deny everything else.
src_ip=0.0.0.0-255.255.255.255                    action=ip_deny  method=ALL
src_ip=::-ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff action=ip_deny  method=ALL
```

Web Cache Size

The Apache Traffic Server default install configures this to be 256MB, a rather small size as is noted in the configuration file.

I eventually went with 1GB. The following is found in the config file /usr/local/etc/trafficserver/storage.config.

```
var/trafficserver 1024M
```

Web Cache Partitions

The Apache Traffic Server default install doesn't really provide for this. I found over time this can cause all sorts of issues relating to disk lock contention.

The following is found in the config file /usr/local/etc/trafficserver/volume.config.

```
volume=1 scheme=http size=25%
volume=2 scheme=http size=25%
volume=3 scheme=http size=25%
volume=4 scheme=http size=25%
```

Start It Up!

Once the above has been completed, it's time to give it all a try.

```
sudo /usr/local/bin/trafficserver start
```

At this point you should have a workable, albeit very default web caching proxy server.

Startup your favorite browser, configure it to use your new proxy server as a web proxy for both HTTP and HTTPS, and watch your browsing speed improve immediately.

Next Page: WebProxyCacheTuning