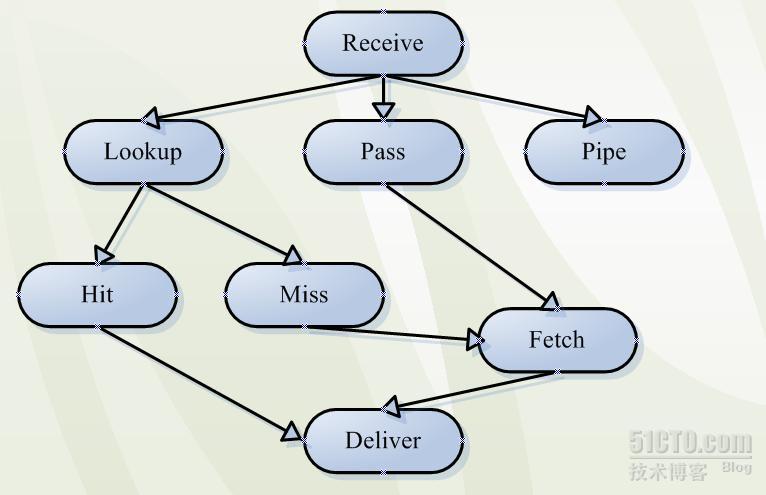
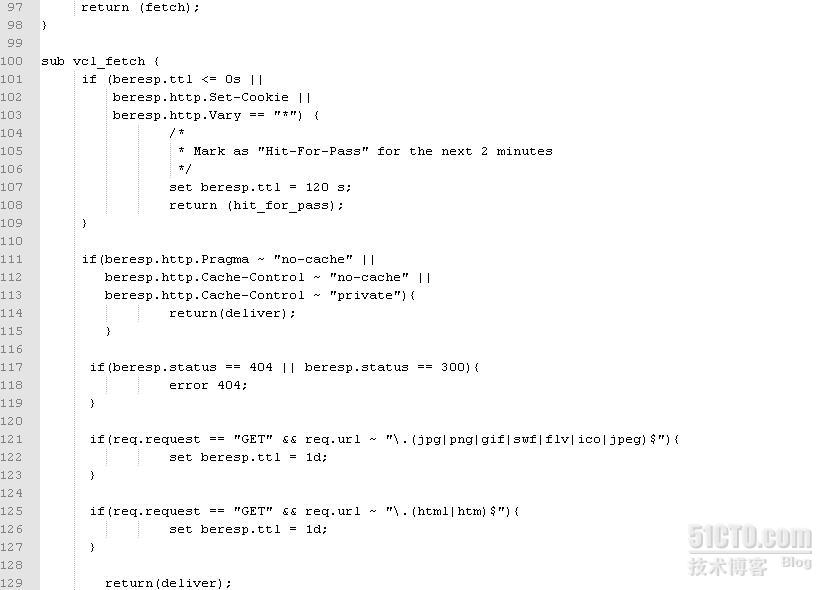
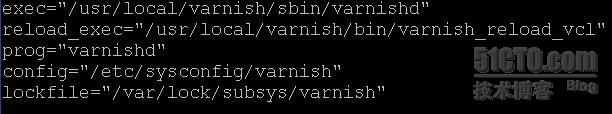
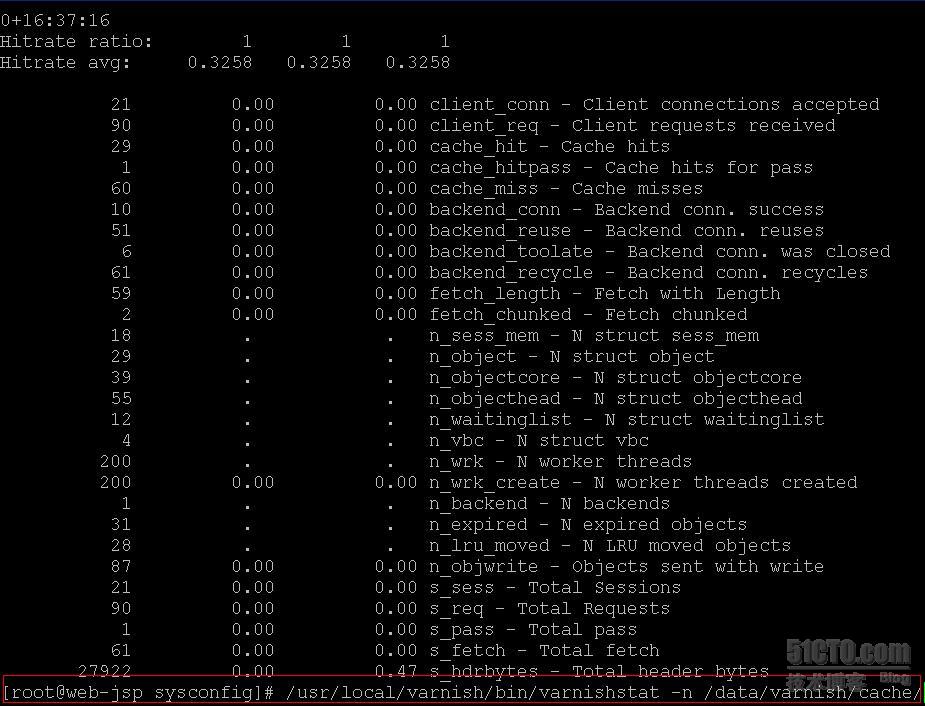
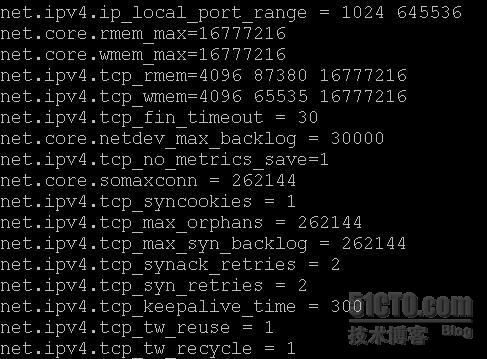
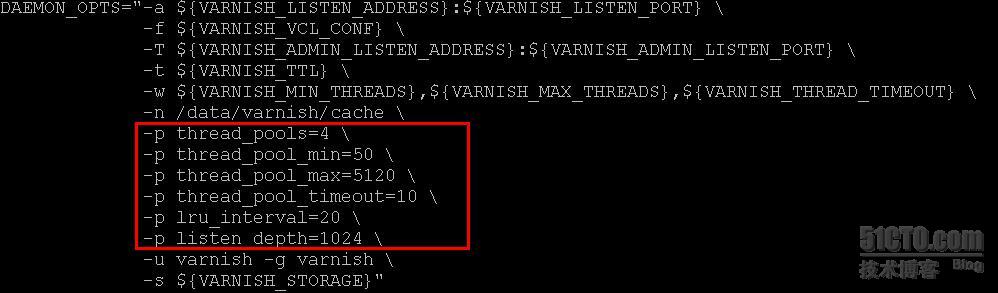
Varnish是一个轻量级的Cache和反向代理软件。先进的设计理念和成熟的设计框架是Varnish的主要特点。下面是Varnish的一些特点：

* 基于内存进行缓存，重启后数据将消失；
* 利用虚拟内存方式，I/O性能好；
* 支持设置0~60秒精确缓存时间；
* VCL 配置管理比较灵活；
* 32位机器上缓存文件大小为最大2GB；
* 具有强大的管理功能；
* 状态机设计巧妙，结构清晰；
* 利用二叉堆管理缓存文件，可达到积极删除目的；

在安装Varnish之前，系统如果没有安装pcre，在编译Varnish 2.X以上版本时，会提示找不到PCRE库，而pcre库是为了兼容正则表达式，所以必须先安装pcre库。下面是pcre的安装过程：   
首先，下载pcre软件包：   
[](http://www.newasp.net/attachment/article/2013/0425/092707_97147404.jpg)   
进行加压缩软件包，进行编译安装：   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092707_37832966.jpg)   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092707_30545403.jpg)   
至此，pcre库已经安装完成。接着，建立Varnish用户以及用户组，并创建Varnish缓存目录和日志目录。   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092707_30014215.jpg)   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092707_71663471.jpg)   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092708_28712332.jpg)   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092708_92006703.jpg)   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092708_54075679.jpg)   
现在可以进行安装Varnish，这里将Varnish安装到/usr/local/目录下，操作如下：   
       下载最新Varnish-3.0.3软件包:   
        [](http://www.newasp.net/attachment/article/2013/0425/092708_95992879.jpg)   
        设置安装参数，然后进行编译安装：   
        [高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092709_56556224.jpg)   
        [高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092709_30900123.jpg)   
        [高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092709_51695156.jpg)   
        [高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092709_86429191.jpg)   
        将varnish配置文件和服务写入到系统：   
        [高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092709_50648183.jpg)   
        [高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092709_15008277.jpg)   
至此，Varnish安装完成。现在开始配置Varnish，在配置Varnish之前首先了解一下Varnish处理流程：   
          
        [](http://www.newasp.net/attachment/article/2013/0425/092710_71965917.jpg)   
        Varnish处理HTTP请求的过程大致分为如下几个步骤：   
        1> Receive状态：请求处理入口状态，根据VCL规则判断该请求应该Pass或Pipe，还是进入Lookup(本地查询)。   
        2> Lookup状态：进入此状态后，会在hash表中查找数据，若找到，则进入Hit状态，否则进入Miss状态。  
        3> Fetch状态：在Fetch状态下，对请求进行后端获取，发送请求，获得数据，并进行本地存储。   
        4> Deliver状态：将获取到的数据发送给客户端，然后完成本次请求。   
        现在Varnish的处理原理现在大家都明白了，那么下面来进行配置一个实例。由于版本不同，Varnish配置文件的写法也存在一定的差异，本配置文件以Varnish 3.x版本为基准。   
        Varnish安装完成后，默认的配置文件为/usr/local/varnish/etc/varnish/default.vcl,此文件内容默认全部被注释掉。这里以这个文件为模板，创建一个新的文件vcl.conf，并且将其放到/usr/local/varnish/etc目录下。配置完成的vcl.conf文件如下：   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092710_78640000.jpg)   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092710_64558899.jpg)   
  
  
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092710_37168054.jpg)   
  
[](http://www.newasp.net/attachment/article/2013/0425/092711_65990840.jpg)   
  
  
  
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092711_51944755.jpg)   
  
[](http://www.newasp.net/attachment/article/2013/0425/092711_38184808.jpg)   
在安装Varnish时，已经将Varnish的管理脚本复制到相应的目录下，这里稍作修改即可。首先修改/etc/sysconfig/varnish文件。配置好的文件如下：   
[](http://www.newasp.net/attachment/article/2013/0425/092711_70159511.jpg)   
这里需要说明的是，在32位操作系统下，最大只能支持2GB的缓存文件Varnish\_cache.data，如果需要更大的缓存文件，则需要安装64位的操作系统。   
接下来修改的文件是/etc/init.d/varnish，找到如下机房，修改相应的路径即可：   
[](http://www.newasp.net/attachment/article/2013/0425/092712_23233046.jpg)   
其中，exec用于指定Varnish的路径，只需要修改为Varnish安装路径下赌赢的Varnishd文件即可;config 用于指定Varnish守护进程配置文件路径。   
两个文件修改完毕，就可以授权、运行/etc/init.d/varnish脚本了。执行过程如下：   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092712_54645012.jpg)   
最后启动varnish，如下所示：   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092712_33958148.jpg)   
查看运行状态：   
[高性能HTTP加速器Varnish-3.0.3搭建、配置及优化](http://www.newasp.net/attachment/article/2013/0425/092712_87761921.jpg)   
如上图，可得知Varnish已经成功启动。现在可以测试是Varnish的作用了，可通过Curl进行测试：   
[](http://www.newasp.net/attachment/article/2013/0425/092712_72225906.jpg)   
通过上图，可得知该URL链接已经被缓存，缓存命中率的高低直接说明了Varnish的运行状态和效果，较高的缓存命中率说明Varnish运行状态良好，Web服务器的性能也会提高很多；反之，过低的缓存命中率说明Varnish的配置可能存在问题，需要进行调整。因此，从整体上了解Varnish的命中率和缓存状态，对于优化和调整Varnish至关重要。   
Varnish提供了一个Varnishstat命令，通过它可以获得很多重要的信息。下面是一个Varnish系统的缓存状态：  
[](http://www.newasp.net/attachment/article/2013/0425/092713_51882703.jpg)   
由于执行完varnishstat命令之后，会自动跳转到一个画面，就无法看到执行命令。为方便大家理解，将命令放在执行结果最底部。这里需要注意一下几点：

* "Client connections accepted"：表示客户端向方向代理服务器成功发送HTTP请求的总数量。
* "Client requests received"：表示到现在为止，浏览器向反向代理服务器发送HTTP请求累计次数。由于会使用长连接，因此这个值一般会大于"Client connections accepted"的值。
* "Cache hits"：表示方向代理服务器在缓存区中查找并且命中缓存的次数。
* "Cache misses"：表示直接访问后端主机请求数量，也就是非命中数。
* "N struct object"：表示当前被缓存的数量。
* "N expired objects"：表示过期的缓存内容数量。
* "N LRU moved objects" ：表示被淘汰的缓存内容个数。

Varnish的安装和配置基本上完成了。安装之后Varnish是否能稳定、快速地运行，与Linux本身的优化及Varnish自身参数的设置有很大关系。在安装配置完Varnish后，还必须从操作系统和Varnish配置参数两个方向对Varnish服务器进行性能优化，从而最大限度地发挥Varnish的性能优势。   
内核参数是用户系统内核之间交互的一个接口，通过这个接口，用户可以在系统运行的同时动态更新内核配置，而这些内核参数是通过Linux proc文件系统存在。因此，可以通过调整proc文件系统达到优化Linux性能的目的。   
修改/etc/sysctl.conf文件进行优化，具体参数如下：   
[](http://www.newasp.net/attachment/article/2013/0425/092713_50286689.jpg)   
修改ulimit设置，默认情况下Ulimit设置完成之后会临时生效。当下次重启机器之后，Ulimit会失效。为了下次机器启动之后，还会生效。将ulimit设置放到/etc/rc.d/rc.local文件下。具体参数如下：   
高性能HTTP加速器Varnish-3.0.3搭建、配置及优化   
至此，LINUX系统方面优化就算完成了。现在还是优化Varnish。打开/etc/sysconfig/varnish启动脚本，优化参数如下：   
[](http://www.newasp.net/attachment/article/2013/0425/092714_12755373.jpg)   
至此，Varnish的安装、配置以及优化基本上完成了。