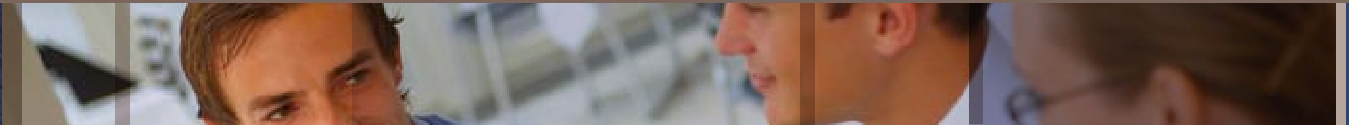


SGOS 4, 5 Series

Enhanced Caching of Static Flash Video

Smoothing popularity based traffic bursts

We all have received an email or instant message from friends/family/co-workers about some new "must see" video. The proliferation of user-generated content and video sharing sites has made for a constant source of "new" material. However the bandwidth demands of such media often congest WAN links. While blocking is one approach, another is to aggressively cache and server content from local storage. This combined with appropriate content prioritization can allow video and business critical applications to co-exist.

Most video sharing sites use Adobe FlashPlayer to view static (non-streamed) video files. These files are actually retrieved via the HTTP protocol and thus can be object cached, then later requests are served from the proxy's disk rather than the utilizing the WAN link. While many sites make these video files cacheable the bandwidth sensitive may choose to enhance and extend the normal caching behavior to maximize savings.

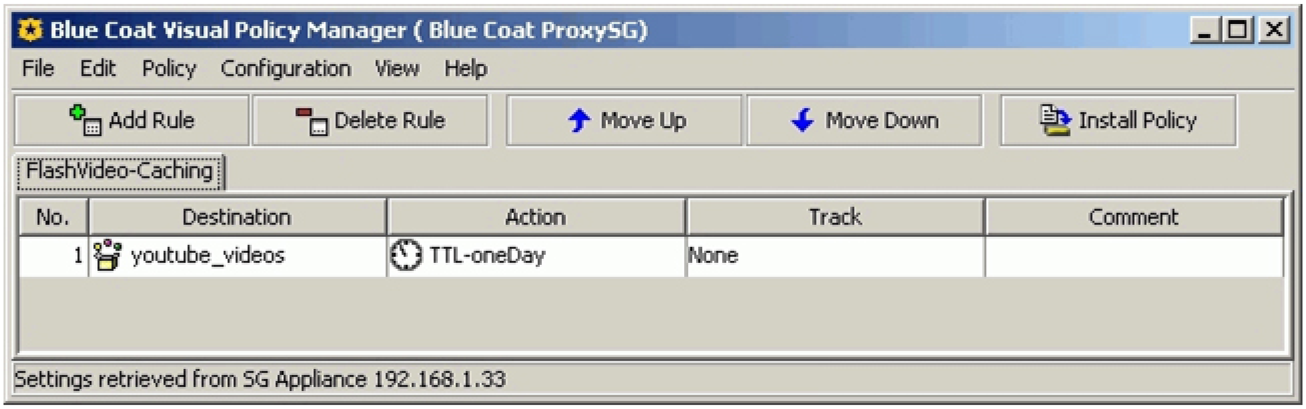
Typically only a fraction of the videos on a site will be viewed, but those that are likely will be viewed multiple times due to popularity bursts. This further increasing the value of object caching so that only one copy of such a video needs to be transferred across the WAN.

YouTube

The Google subsidiary YouTube actively marks it's videos as cacheable for one hour (3600 seconds) with a header such as "Cache-Control: max-age=3600". This refresh interval can be extended using the "ttl()" policy property. The following policy makes YouTube video files be cached for one day (86400 seconds):

```
define condition youtube_videos
    url.domain=youtube.com url.path.regex="/get_video"
    url.path.regex="/get_video.+youtube.com"
end
<cache>
   Condition=youtube_videos  ttl(86400)
```

The above policy can be created in VPM with a "Web Content Layer" containing a combined Destination object for the URL tests and a TTL action.



No.	Destination	Action	Track	Comment
1	youtube_videos	TTL-oneDay	None	

Settings retrieved from SG Appliance 192.168.1.33

An additional policy property not available in the VPM "never_refresh_before_expiry(yes)" will enforce that the proxy performs at most one freshness check per time-period.

Yahoo Video

Videos are served from numerically named servers with a "Content-Type" response header of "video/flash".

MSN Video

The user is forced to watch a commercial delivered via the RTMP flash-streaming protocol (not HTTP; TCP port 1935) before they are shown the static flash video file (delivered by HTTP). The HTTP flash video file arrives with response headers identifying "Content-Type: video/x-flv" and a short expires time (ex: three minutes).

AOL Video

Videos are served from "movies.myspace.com" with a url extension of ".flv". These objects were labeled with short expires time (ex: five minutes).

General Flash Video Policy

The above and many other websites provide Flash Player video files. This policy should match most:

```
define condition flash_video_files
   url.extension=flv
   url.extension=swf
   request.x_header.x-flash-version=""
   response.header.Content-Type="video/(x-|)flv"
   response.header.Content-Type="video/(x-|)flv"
   response.header.Content-Type="video/(x-|)flash"
   http.response.data.4.regex.case_sensitive="^CWS[0-9]"
   http.response.data.4.regex.case_sensitive="^FWS[0-9]"
end condition flash_video_files
<cache>
   condition=flash_video_files  ttl(86400)
```

The defined condition above can be used to test multiple aspects of the transaction in a logical OR, meaning any one line of true tests will make the entire condition true. This would be created in the VPM via a defined destination condition. The "force_cache(all)" action could be added to instruct the proxy to ignore non-cacheable markings on objects (such as with the "Cache-Control: no-cache" response header).