IBM Software Group

IBM Http Server/Plugin Performance Tuning.

Naveen Shetty (naveen@us.ibm.com), Advisory Software Engineer.

Marvin Knight (knightm@us.ibm.com), Advisory Software Engineer.

Date: August 26th 2014

WebSphere® Support Technical Exchange

ON DEMAND BUSINESS™

Agenda: IHS & Plugin Performance Tuning

1. Out of the Box Tuning Concerns
2. Configuration features to Avoid
3. Configuration Change Implications on performance.
4. SSL Tuning Considerations
5. Network Tuning Considerations
6. OS related tuning Considerations.
7. Plugin Performance tuning

Out of Box IHS tuning Concerns

- Calculating Maximum Simultaneous connections
- SSL- Cipher Ordering
- Sendfile – may increase CPU utilization.
- AIX® - MALLOCMULTIHEAP settings
- Windows® - FRCA, aka AFPA

1. Determining MaxConnections

On Windows O/S

- 125% of Max Simultaneous connections during peak load.
- IHS on Windows is a 32-bit application.
  - SingleParent process and a single multi-threaded Child process
  - ThreadsPerChild – Suggested upper limit of 2000
  - ThreadLimit – Same as ThreadsPerChild.
  - Raising ThreadsPerChild limits risks child process crashes
  - mod_mem_cache, Rewrite directives restrict the upper limit

1. Determining MaxConnections (contd)

- On Unix® O/S
  - one single threaded Parent process which starts one or more multi-threaded Child processes.
  - Relevant Config directives – StartServer, ServerLimit, ThreadsPerChild ThreadLimit, MaxClients, MaxSpareThreads, MinSpareThreads
  - ThreadLimit and ServerLimit must appear before the other directives
  - Larger ThreadsPerChild (i.e fewer processes) also results in fewer dedicated web container threads being used by the ESI invalidation feature of the WebSphere Plugin.
  - Increasing ThreadsPerchild too high on heavily loaded SSL servers may incur more CPU and throughput issues, as there is additional contention for memory.
  - Memory Contraints – per-server memory overhead.

1. Determining MaxConnections (contd)

- Using mod_status or mod_mpmstats to determine Max Simultaneous connections.
  - mod_status: Gives an idea on total requests currently being processed and total idle workers.
  - mod_mpmstats: Gives dispersion and state of threads.
    - Helps in optimizing MaxClient,
    - Can help in setting a suitable KeepAliveTimeout
    - Configurable scan intervals gives idea into optimal MaxClient settings.
- Netstat command can be used to determine TCP® connection state between client and IHS.

2. Configuration features to Avoid

- HostnameLookups On
- IdentifyCheck On
- mod_mime_magic
- ContentDigest On
- MaxRequessPerChild to non-zero
- .htaccess files
- Disabling Options FollowSymLinks
- detailed logging

3. Configuration Change Implications

A> Higher ThreadsPerChild

- will result in lower memory use as long its less than normal server TCP connections.
- Extremely high ThreadsPerChild may result in address space limitations.
- Lower number of connections with WAS, better sharing of markdown information.
- Higher values for ThreadsPerChild result in higher CPU utilization for SSL processing.
- In Older RH Linux results in high CPU utilization.
- Additionally, RewriteMap, mod_mem_cache, mod_ibm_ldap, mod_ext_filter excerbate high CPU util

3. Configuration Change Implications

B> MaxClients

- Increase in MaxClient warrants increase in MaxSpareThreads
- Else, CPU will be consumed terminating and creating child process' when load changes by a relatively small amount.

C> ExtendedStatus

- When this is set to On, web server CPU usage may increase by as much as one percent.

SSL Considerations - Ciphers

- first supported cipher in ordered list which is selected.
- IHS prefers AES and RC4 ciphers over computationally expensive Triple-DES (3DES)
- Order of the SSLCipherSpec directives dictates the priority of the ciphers

SSL-Cipher-Configuration

```
<VirtualHost *:443>
  SSLEnable
  Keyfile keyfile.kdb

  ## SSLv3 128 bit Ciphers
  SSLCipherSpec SSL_RSA_WITH_RC4_128_MD5
  SSLCipherSpec SSL_RSA_WITH_RC4_128_SHA

  ## FIPS approved SSLV3 and TLSv1 128 bit AES Cipher
  SSLCipherSpec TLS_RSA_WITH_AES_128_CBC_SHA

  ## FIPS approved SSLV3 and TLSv1 256 bit AES Cipher
  SSLCipherSpec TLS_RSA_WITH_AES_256_CBC_SHA

  ## Triple DES 168 bit Ciphers
  ## These can still be used, but only if the client does
  ## not support any of the ciphers listed above.
  SSLCipherSpec SSL_RSA_WITH_3DES_EDE_CBC_SHA

  ## The following block enables SSLv2. Excluding it in the presence of
  ## the SSLv3 configuration above disables SSLv2 support.
  ## Uncomment to enable SSLv2 (with 128 bit Ciphers)
  #SSLCipherSpec SSL_RC4_128_WITH_MD5
  #SSLCipherSpec SSL_RC4_128_WITH_SHA
  #SSLCipherSpec SSL_DES_192_EDE3_CBC_WITH_MD5

</VirtualHost>
```

SSL-CipherSpec- IHS-V80 & later

- SSLCipherSpec SSLv3 SSL_RSA_WITH_RC4_128_SHA SSL_RSA_WITH_RC4_128_MD5 TLS_RSA_WITH_AES_128_CBC_SHA TLS_RSA_WITH_AES_256_CBC_SHA SSL_RSA_WITH_3DES_EDE_CBC_SHA
- SSLCipherSpec TLSv10 SSL_RSA_WITH_RC4_128_SHA SSL_RSA_WITH_RC4_128_MD5 TLS_RSA_WITH_AES_128_CBC_SHA TLS_RSA_WITH_AES_256_CBC_SHA SSL_RSA_WITH_3DES_EDE_CBC_SHA
- SSLCipherSpec TLSv11 SSL_RSA_WITH_RC4_128_SHA SSL_RSA_WITH_RC4_128_MD5 TLS_RSA_WITH_AES_128_CBC_SHA TLS_RSA_WITH_AES_256_CBC_SHA SSL_RSA_WITH_3DES_EDE_CBC_SHA
- # TLSv12 is left at the default TLS_RSA_WITH_AES_128_GCM_SHA256 TLS_RSA_WITH_AES_256_GCM_SHA384 TLS_RSA_WITH_AES_128_CBC_SHA256 TLS_RSA_WITH_AES_256_CBC_SHA256 TLS_RSA_WITH_AES_128_CBC_SHA TLS_RSA_WITH_AES_256_CBC_SHA SSL_RSA_WITH_3DES_EDE_CBC_SHA

SSL-LogFormat

- LogFormat Directive:

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"SSL=%{HTTPS}e\" \"%{HTTPS_CIPHER}e\" \"%{HTTPS_KEYSIZE}e\" \"%{HTTPS_SECRETKEYSIZE}e\"" ssl_common

CustomLog logs/ssl_cipher.log ssl_common
```

- ssl_cipher.log:

```
127.0.0.1 - - [18/Feb/2005:10:02:05 -0500] "GET / HTTP/1.1" 200 1582 "SSL=ON" "SSL_RSA_WITH_RC4_128_MD5" "128" "128"
```

SSL-Certificate Size

- Every doubling of key size costs 4-8 times more CPU
- Industry standards changing from1024-bit to 2048-bit certificates
- Large size certificate SSL handshake with new session is primary cost of computation.
- Using keep-alive, re-using SSL sessions aids performance

SSL-Connections Performance

- SSL CPU utilization directly proportional to ThreadsPerChild
- MALLOCMULTIHEAP setting in AIX IHSRoot/bin/envvars
- Use of cryptographic accelerator
- HTTP keep-alive has a much larger benefit for SSL than for non-SSL. A small KeepAliveTimeout is better than setting KeepAlive OFF
- Creating shared-key across loadBalanced connections and reusing SSL sessions for subsequent connections reduces CPU overheads during SSL Handshake for every new TCP connection.
- Sticky Sessions of SessionAffinity to a WebServer in addition to reusing SSL Sessions avoids creation of new shared keys for every new TCP connection between client and webserver.
- The generation of the shared key during SSL handshake is CPU intensive.

Other Performance improving considerations

- Network tuning - Increasing the default size of TCP receive buffers.
- no -o rfc2414=1
- Operating System - http://www.ibm.com/support/knowledgecenter/SSEQTP_8.5.5/com.ibm.websphere.base.doc/ae/tprf_tuneopsys.html?lang=en
- Slow startup, or slow response time from proxy or LDAP
- High disk I/O with IBM HTTP Server on AIX
- High CPU in child processes after WebSphere plugin config is updated.
- Reduce the disk I/O rate due to access logging.

WebSphere plugin considerations

- Tuning IHS to make the MaxConnections parameter more effective
- Tuning IHS for efficiency of Plugin markdown handling
- Tuning IHS for efficiency of ESI invalidation servlet / web container threads

Plug-in Performance Tips

- **Web Server Plug-in**
  - **SSL**
  - **Caching**
  - **Timers**
  - **Connections**
  - **Multi-Process impact**
  - **Load Balancing**
  - **Miscellaneous Setting**

SSL and the Plug-in

- SSL offload
  - Offload in front of web server
    - HTTPSIndicatorHeader
      - Web Container property
      - Name of Header set by device that offloads SSL
      - Header could be set by web server if SSL offload device has no capability
        - mod_headers,
        - RequestHeader set

SSL and the Plug-in

- SSL offload
  - At Web Server
    - HTTP transport only
    - GSKit error forces offload - Initialization
      - PSWD error
      - Personal Certificate expired
      - 8.5.5 changes behavior
        - UseInsecure true ,plugin property

SSL and the Plug-in

- GSKit error after initialization
  - Handshake error – 500 response code
    - Bad Cert
    - Ciphers not allowed
    - Mutual Auth – missing signers
  - Good Handshake
    - Level of security – more overhead based on the higher the security level
    - Mutual Authentication

SSL and the Plug-in

- PK78546
  - SSLConsolidate
    - Use with multiple clusters
    - Shares GSKit environment
    - 7.0.0.3 and higher
      - 6.1.0.23
  - SSLPKCSDriver also added
  - SSLPKCSPassword also added

ESI Cache

- ESIEnable
  - True or False, True is default
- ESIMaxCacheSize
  - Integer in 1K byte units
  - 1024K is default value
  - One cache per process
    - More efficient with less process, so a tradeoff versus performance gain with multiple process
- ESICacheidFull
  - Adds host name to cacheid, false by default
- ESIInvalidationMonitor (false by default)
- ESIEnableToPassCookies (false by default)

Timers

- ConnectTimeout
- ServerIOTimeout
- ServerIOTimeoutRetry
- RefreshInterval
- RetryInterval
- PM94198
  - Environment variable set in Apache or IBM HTTP Server
    - websphere-serveriotimeout
    - websphere-serveriotimeoutretry
    - websphere-shorten-handshake

ConnectTimeout

- Determines how long to wait on a connection to application server
  - 5 seconds usually plenty of time
  - Long timeout can be detremental in large cluster
    - 6 of 12 members powered off for maintenance would be 6 x 5 = 30 second delay
    - Consider removing server from plugin-cfg.xml rather than power off

ServerIOTimeout

- Determines how long the Plug-in will wait on a response from the application server
  - Should be long enough to allow for longest running request for application server
  - 0 – no timeout
    - Should not normally be used
  - Retries due to large cluster can be detremental
  - Negative number
    - Mark down
  - Positive numer
    - Don't mark down

ServerIOTimeoutRetry

- Limit the number of retries on a request that times out due to ServerIOTimeout
  - -1
    - No limit
    - Can be number of members in cluster +1
  - 0
    - No retries
  - N
    - Specify the actual number of times to retry

RefreshInterval

- How often the Plug-in checks for a change in plugin-cfg.xml
  - Stat of plugin-cfg.xml
    - Web Server child process needs permission
  - Frequest changes to plug-in could cause performance issue with a large plugin-cfg.xml

RetryInterval

- Time for when the Plug-in will retry a server marked down
  - N
    - To small a value could lead to frequent long responses
    - To long could delay server being marked back up
    - Trade off depending on reason for being marked down

PM94198

- Introduces new environment variable for Apache or IBM HTTP Server, url based override
  - Websphere-serveriotimeout
  - Websphere-serveriotimeoutretry
  - Websphere-shorten-handshake
    - SetEnvIf Request_URI "\.jsp$" websphere-serveriotimeout=10
    - SetEnvIf Request_URI "\.jsp$" websphere-serveriotimeoutretry=-1
    - SetEnvIf Request_URI "\.jsp$" websphere-shorten-handshake=1
    - 7.0.0.31, 8.0.0.8, 8.5.5.2

Connections

- MaxConnections
  - 0, -1
    - No limit
    - Controlled by appserver
  - N
    - Number of pending connections allowed
      - Pending connection is a request open to appserver where the plugin has received no response
      - Per process
      - Hard to control at plugin level

Connections

- Persistent connections
  - Set by application server
    - Web container transport chain
      - Use persistent (keep-alive) connections check box
  - ConnectionTTL
    - Plugin_PersistTimeOut_Reduction custom property
    - PM76420 – 7.0.0.29, 8.0.0.6, 8.5.0.2
    - Time when plugin closes idle socket
    - 28 second default

Multiple Processes

- Plug-in has separate cache for each process
- Plug-in has separate counters for each property
- Crash would only bring process down
- Complicates load balancing
- Trade off for web server performance and reliability

Load Balancing

- LoadBalance
  - Random
    - Usually better for large clusters
  - Round robin
  - Use LogLevel=”Stats” to check
  - Load balances new requests
    - Not new sessions

Miscellaneous Settings

- LogLevel
  - Trace – very verbose, avoid except for debugging
    - Reset back to Error after debugging
    - Start with fresh log
  - Error
    - Normal setting
- PostBufferSize
  - Value of 0 – can't be retried

Reference

- http://publib.boulder.ibm.com/httpserv/ihsdiag/
- http://publib.boulder.ibm.com/httpserv/ihsdiag/unix_index.html
- http://www.ibm.com/software/webservers/appserv/was/library/index.html

Connect with us!

1. **Get notified on upcoming webcasts**
   Send an e-mail to wsehelp@us.ibm.com with subject line “wste subscribe” to get a list of mailing lists and to subscribe
2. **Tell us what you want to learn**
   Send us suggestions for future topics or improvements about our webcasts to wsehelp@us.ibm.com

Questions and Answers

Additional WebSphere Product Resources

- Learn about upcoming WebSphere Support Technical Exchange webcasts, and access previously recorded presentations at:
  http://www.ibm.com/software/websphere/support/supp_tech.html
- Discover the latest trends in WebSphere Technology and implementation, participate in technically-focused briefings, webcasts and podcasts at:
  http://www.ibm.com/developerworks/websphere/community/
- Join the Global WebSphere Community:
  http://www.websphereusergroup.org
- Access key product show-me demos and tutorials by visiting IBM Education Assistant:
  http://www.ibm.com/software/info/education/assistant
- View a webcast replay with step-by-step instructions for using the Service Request (SR) tool for submitting problems electronically:
  http://www.ibm.com/software/websphere/support/d2w.html
- Sign up to receive weekly technical My Notifications emails:
  http://www.ibm.com/software/support/einfo.html