Hol

**Estudi**

client: FUOC

projecte: Requeriments Wowza

data: 9/05/2014

# CONFIDENCIALITAT

Drets d’Autor

© 2013, 3&Punt Solucions Informàtiques SL Tots els drets reservats.

# ÍNDEX

CONFIDENCIALITAT 2

ÍNDEX 3

INTRODUCCIÓ 4

Contacte amb Wowza 4

Ample de banda 4

Test Tool 5

Descàrrega 5

Instal·lació del client (Màquina Amazon) 6

Instal·lació del servidor (Màquina ilimit) 6

Configuració dels Tests 6

Resultats 8

# 

# INTRODUCCIÓ

Necessitem poder calcular els requeriments de màquines per diferents escenaris de videochat i langBlog Wowza. Els escenaris són:

* Fins a 1.000 usuaris
* De 1.000 a 3.000 usuaris
* De 3.000 a 10.000 usuaris
* Més de 10.000 usuaris

En el cas de videochat, on es centrarà aquest estudi, tenim present que hi **haurà 2 videos enregistrats i 10 visualitzacions al mes**.

# Contacte amb Wowza

Després de contactar amb Wowza ens indiquen que el problema més gran està en l’**ample de banda**.

Suggereixen que el servidor ha de ser adequat recomenant una darrerar versió de quad o dual-quad CPU 4 – 8 GB de RAM i SO 64 bit amb JDK.

També ens proporcionen una eina de Test, **RMTP Load Test Tool**:

[http://www.wowza.com/forums/content.php?122](http://www.wowza.com/forums/content.php?122" \t "_blank)

## Ample de banda

Per regla general hem de seguir la fórmula:

**Stream bitrate \* max clients concurrent < ample de banda del servidor**

**La mitjana de bitrate és 150 kbps**



## Test Tool

L'eina de prova de càrrega és una forma eficient per generar connexions simulades de Adobe Flash RTMP a Wowza Streaming. Pot simular tant en directe com el vídeo sota demanda (VOD) streaming. Un únic equip que executa l'eina de prova de càrrega pot simular centenars de connexions per a reproductor de Flash RTMP. L'eina s'executa en un o més equips que anomenarem l'ordinador (s) client.

Aquesta eina funciona per Wowza Straming Engine 4, que és la que tenim actualment.

S’ha d’accedir a

<http://www.wowza.com/forums/content.php?122>

### Descàrrega

I sol·licitar l’accés, fan signar un acord de confidencialitat (amb la meva persona) i proporcionen l’enllaç a

<http://www.wowza.com/downloads/WowzaStreamingEngine-4-0-1/WowzaStreamingEngine-loadtool-4.0.1.zip>

INSTRUCTIONS FOR OBTAINING THE LOAD TEST TOOL:

There are two versions of the Flash RTMP Load Test Tool. Download the version of the tool that matches the server version you're running:

Wowza Streaming Engine 4.0:

http://www.wowza.com/downloads/WowzaStreamingEngine-4-0-1/WowzaStreamingEngine-loadtool-4.0.1.zip

or

Wowza Media Server 3.5-3.6:

http://www.wowza.com/downloads/WowzaMediaServer-3-5-0/WowzaMediaServer-loadtool-3.5.0.zip

For best performance, please update your Wowza Media Server(R) or Wowza Streaming Engine(tm) software with the latest patch:

http://www.wowza.com/forums/content.php?199-Development-Builds

If you have technical questions, information can be found on our support forum:

http://www.wowza.com/forums/forumdisplay.php?72

If you still require technical assistance, please email support@wowza.com.

### Instal·lació del client (Màquina Amazon)

1. Instal·lar JDK 6 o superior
2. Instal·lar Wowza Streaming Engine (pot ser la versió developer)
3. Assegurat que el Wowza Streaming Engine està be configurat per tal d’usar totes les capacitats de la màquina (Veure [performance tuning](http://www.wowza.com/docredirect.php?doc=performanceGeneralTuning))
4. L’eina necessita que el **\_defaultVHost\_ (VHost)** tingui el seu pool de threads. Per defecte, Wowza Streaming Engine, està configurat per usar el nivell de thread pool del servidor. Per configurar VHost-level thread pool cal editar /usr/local/WowzaStreamingEngine/conf/VHost.xml I fer els canvas:
   1. HandlerThreadPool/PoolSize: 60 x [total-core-count] . A la màquina d’Amazon 60
   2. TransportThreadPool/PoolSize: 40 x [total-core-count]. A la màquina d’Amazon 40

**Atenció:** després de fer les proves s’ha de deixar com estava inialment

1. Copiar el contingut dels directoris **bin**, **conf** i **lib**  en el directori de Wowza.
2. Instal·lem els exemples /usr/local/WowzaStreamingEngine/examples/installall.sh i reiniciem el servei

### Instal·lació del servidor (Màquina ilimit)

No cal fer cap tipus d’instal·lació ja que es simula tot des del client.

### Configuració dels Tests

La configuració està a **[install-dir]/conf/Tests.xml**. Per defecte està preconfigurades dues eines (**live**  i **vod**). Els tests **live** serveixen per testejar l’stream en viu i el **vod** serà per streaming VOD (que ja ens serveix per l’objectiu d’aquest estudi). Ens bassem en **live** per crear **videochat**.

#### Configuració detallada en Tests.xml

**workerCount**: Nombre de clients de Flash RTMP per simular.

**fileCount**: Nombre de noms diferents simular l’stream. Si s'estableix a 0 (zero), llavors només un sol stream per totes les connexions dels clients. El nom de la seqüència es deriva mitjançant la concatenació [streamName] i [streamExt]. Si és més gran que zero, llavors el nom de la seqüència inclourà un valor d'índex ([index]) que es troba entre 1 i [fileCount]. En aquest cas, el nom de la seqüència exercit per cada connexió de client és [streamName] [índex] [streamExt]. Si [doRandom] s'estableix en false, el [index] valor és simplement s'incrementa per cada connexió de client. Si [doRandom] és cert, llavors el [index] valor és seleccionat a l'atzar.

**streamName**: Base nom de la seqüència.

**streamExt**: extensió Stream. Per a la transmissió en viu, el valor ha d'estar buit. **vhostName**: nom d'amfitrió virtual.

connectionString: Aplicació i instància de sol · licitud de nom a utilitzar per a la transmissió. El nom d'aplicació especificat ha d'estar configurat en l'equip client local mitjançant la creació d'una carpeta amb el mateix nom a la carpeta [install-dir] / aplicacions. Aquesta és també l'aplicació i l'aplicació instància amb nom utilitzat per la connexió de jugador simulat.

**bufferTime**: Les connexions de clients simulats memòria intermèdia en el client.

**doRandom**: Si això és cert, i si [fileCount] és diferent de zero, llavors el [index] valor per a cada client la connexió es genera aleatòriament.

**doRepeat**: Si és cert i si la reproducció de la prova vídeo a la carta, i després al final de la transmissió d'un arxiu de vídeo a la carta del corrent es reiniciarà des del principi.

**Ipaddress**: adreça IP de l'ordinador de destinació del motor Wowza Streaming.

ports: Llista de ports (delimitat per comes) que s'utilitzarà per transmetre des de l'equip de destinació. Els ports es canvien per a cada nova connexió de client.

Configurem:

|  |  |
| --- | --- |
| Paràmetre | Valor |
| workerCount | 1000 |
| fileCount | 6 |
| streamName | myStream i sample per **vod** |
| streamExt | Per live buit i per vod mp4 |
| vhostName | \_defaultVHost\_ |
| connectionString | live/\_definst\_ i vod/\_definst\_ |
| bufferTime | 300 |
| doRandom | true |
| doRepeat | true |
| ipAddress | 80.94.2.189 (ilimit) |
| ports | 1935 |

#### Execució del Test

Una vegada tenim configurat correctament el Test.xml anem a executar els tests:

##### Per videochat:

Anem al directori **[install-dir]/bin**

**chmod +x performance.sh**

**./performance.sh videochat**

Comença tot el procediment i es fa un log

##### Per vod:

Si has indicar el paràmetre **[fileCount]** t’has d’assegurar que estaran els fitxers amb l’extensió indicada a **[streamExt]** i el nom a **[streamName]** per exemple sample1.mp4, sample2.mp4.... En el nostre cas creem del 1 al 6

Anem al directori **[install-dir]/bin**

**chmod +x performance.sh**

**./performance.sh vod**

#### Interpretació de resultats

Podem anar monitoritzant el servidor des de la interfície web però els tests generen una sortida és de la forma

Status running: 1000 currBehind:0

El valor **running** indica el número actual de connexions RMTP simulades i que estan actualmente executant-se

El valor **currBehind** és el número de connexions que no estan rebent dades lo suficientment ràpid del servidor per tenir el buffer del client ple.

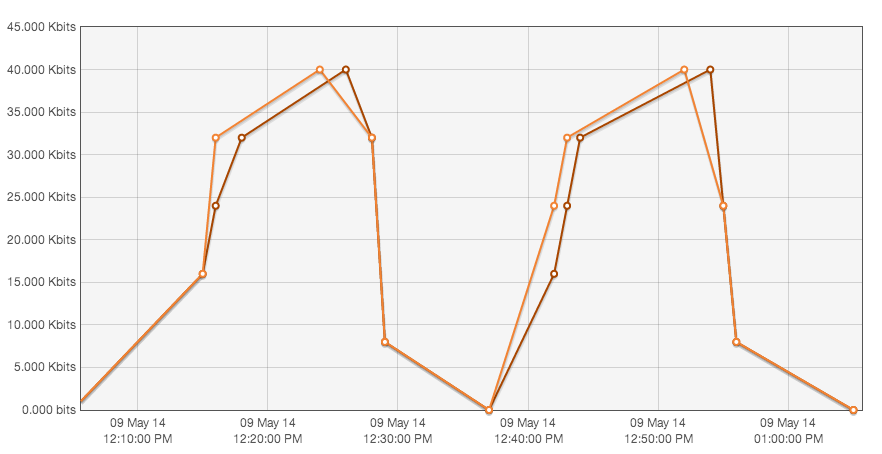
### Resultats

#### Servidor

#### videochat

S’ha executat durant 10 minuts (comptant quan tots els streams estan inciats) i veiem que el currBehind sempre ha estat 0

Veiem que baixa el rendiment de la xarxa en el moment de tenir 900 connexions simultànies



Quedant estabilitzat a **10.320 kbits/s**.

#### vod

S’ha executat durant 10 minuts (comptant quan tots els streams estan inciats) i veiem que el currBehind ha arribat a 1

