

Benchmark of ICE iPush[®] Communication Server V2 Auth. Center

By: ICE Technology Corp., June 21, 2005

Ver.: 1.1

1. Server Host Hardware Spec.

Model: IBM M51 Series (8143I1V)

- ❖ CPU: Intel Pentium 4 (3.0GHz) X 1 / Hyper-Threading enabled
- ❖ RAM: 1GB (DDR400)
- ❖ NIC: 1Gbps
- ❖ HD: 160GB / 7200RPM (IDE)

2. Server OS Tested

- ❖ Windows 2000 Server (with iPush[®] Server V2.1 Build105 Standalone for Windows)
- ❖ Fedora Core 3 Linux (with iPush[®] Server V2.1 Build213 Standalone for Linux)

3. Client Host

- ❖ 2 clients hosts with different number of processes and different number of threads to simulate 3000 concurrent clients
- ❖ 3 clients hosts with different number of processes and different number of threads to simulate 3000 concurrent clients

4. Benchmark Scenario

- ❖ 3000 concurrent connections simultaneously login
- ❖ With 3000 different user accounts (to prevent cache effect)
- ❖ Record the time AC takes to complete
- ❖ Record CPU utilization
- ❖ Record RAM utilization

5. Benchmark Result

- ❖ Average time for 3000 concurrent connections simultaneously login completed: **8,041 ms**
- ❖ CPU usage: **65%-75%**
- ❖ RAM usage:

| | Auth. Center | iPush Messaging Kernel | Total |
|----------------------|------------------|------------------------|----------------------------------|
| Initial Memory Usage | 5,380 KB | 17,048 KB | 23,428 KB (about 22.8 MB) |
| Avg. Memory Usage | 29,038 KB | 118,598 KB | 147,636 KB (about 144 MB) |

6. More Benchmark Explanations

- ❖ The iPush Server versions for Windows or Linux get almost the same average time result.
- ❖ The number of client hosts is not an issue here.
- ❖ We repeated 7~8 times benchmark under each OS, then calculated the average time for 3000 concurrent connections simultaneously login.
- ❖ The bottleneck came from the DB (MySQL) responding to Auth. Center's queries for user's authentication and authorization data. So, in case you need ICE to provide further benchmark on different DB, even file DB (plain text file), it could be arranged.
- ❖ The 3000 concurrent connections simultaneously login have been served in FIRST-COME, FIRST-SERVED base. So, you may consider each user got average 4 seconds to complete login. Please refer to Figure 1. below. We do think in real on-line service case, the login latency time would be shortened since the clients would not simultaneously login in such short time like scenario here.

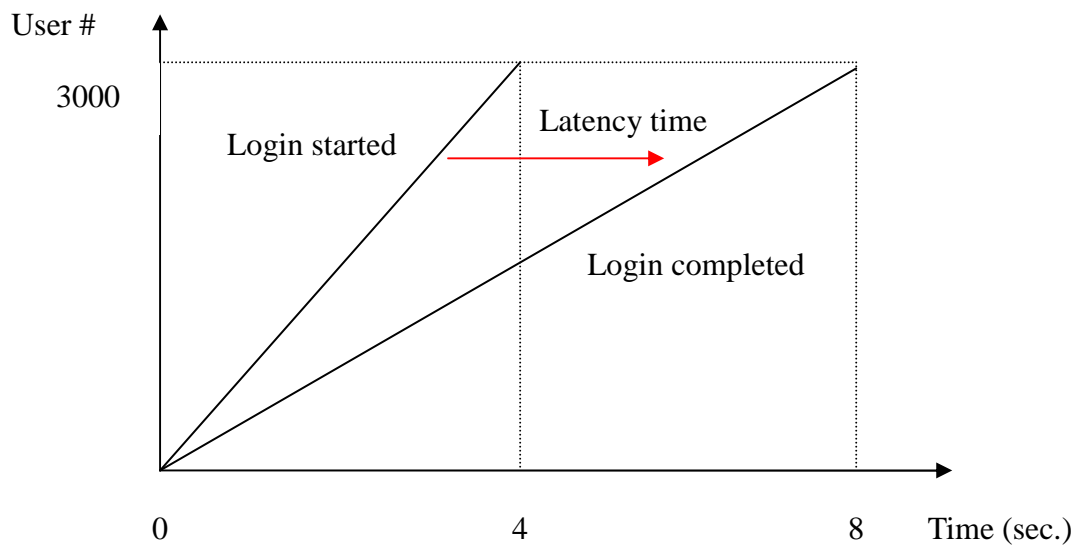


Figure 1.