SMB Version 2:
Scaling from Kilobits to Gigabits

Jim Pinkerton
Microsoft Corporation
Data generated by Teresa Yao and Jay Hamidi
9/23/2008
Agenda

- Introduction
- SMB Version 2 design goals
- Characterizing enterprise branch offices
- WAN performance analysis
  - Overview of the software layers & bottlenecks
  - Test environment & results
  - Production environment & results
- Conclusions
Introduction to SMB/CIFS

- SMB history goes back to 1983
  - SMB/CIFS: SMB -> CIFS -> SMB -> SMBv2
    - CIFS = SMB as it shipped in NT4 server
  - Post CIFS (Windows 2000 and later)
    - Kerberos and domains
    - Shadow copy
    - Server – to – Server copy
    - SMB signing
- Recent new industry trends
  - Proliferation of branch offices
  - Server consolidation and virtualization
  - Mobile workers
  - WAN accelerators
Reduced complexity

- Simplified opcodes
  - SMB > 100 vs. SMBv2 = 19
- Extension mechanism (e.g. create context, variable offsets)

Better WAN throughput, less chattiness

- Credit based flow control
  - Server can control per client resource consumption
- More Flexible compounding
  - Parallel or chained - Response for every element in the chain
- NAT Friendliness - VC count is gone
SMB Version 2 Design Goals (2/2)

- Increased scalability and security
  - Improved scaling:
    - Limits
      |       | SMB1    | SMB2    |
      |-------|---------|---------|
      | Number of Users | Max $2^{16}$ | Max $2^{64}$ |
      | Number of Open Files | Max $2^{16}$ | Max $2^{64}$ |
      | Number of Shares   | Max $2^{16}$ | Max $2^{32}$ |
  - Improved Signing security
    - HMAC SHA-256 replaces MD5
- Symbolic Links
- Durable Handles - Reconnect on loss of connection
Network Characteristics

- Network Media link speed spans $10^6$
  - Cellular modems
  - Dial-Up networking 9600, 19200 Baud
  - 10Mb/s – 1Gb/s Ethernet
  - Wireless LAN
  - 10 Gb/s Ethernet and 32 Gb/s Infiniband

- Latency
  - $<1$ms to 1200ms

- WAN deployments
  - Branch to Data Center (low speed, high latency)
  - Data Center to Data Center (high speed, high latency)
### Branch Office Bandwidth

#### Continental Bandwidth (Mb/s)

<table>
<thead>
<tr>
<th>Continent</th>
<th>North America</th>
<th>Europe</th>
<th>Asia</th>
<th>South America</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>14</td>
<td>33</td>
<td>26</td>
<td>5.5</td>
</tr>
<tr>
<td>Stddev</td>
<td>28</td>
<td>43</td>
<td>28</td>
<td>7.2</td>
</tr>
<tr>
<td>Min</td>
<td>1.3</td>
<td>0.9</td>
<td>3.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Max</td>
<td>155</td>
<td>155</td>
<td>92</td>
<td>25</td>
</tr>
</tbody>
</table>

- Continental is from any branch in a continent to nearest data center
  - Branches in US, Europe, Asia, South America
  - Data Centers in the US, Europe and 2 in Asia

#### Branch Bandwidth Distribution (Mb/s)

<table>
<thead>
<tr>
<th>Bandwidth Range</th>
<th>North America</th>
<th>Europe</th>
<th>Asia</th>
<th>South America</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-1</td>
<td>0</td>
<td>39</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>2-7</td>
<td>5</td>
<td>15</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>8-10</td>
<td>0</td>
<td>8</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>11-20</td>
<td>4</td>
<td>13</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>21-50</td>
<td>4</td>
<td>13</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>51-200</td>
<td>4</td>
<td>13</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Total sample set = **137 branches**
Branch Office Latencies

- Intercontinental is from any branch in a continent to furthest data center
- All latencies are measured with ICMP ping, on an essentially idle network

### Continental Latencies (ms)

<table>
<thead>
<tr>
<th></th>
<th>North America</th>
<th>Europe</th>
<th>Asia</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>108</td>
<td>109</td>
<td>163</td>
</tr>
<tr>
<td>Stddev</td>
<td>69</td>
<td>91</td>
<td>81</td>
</tr>
<tr>
<td>Min</td>
<td>20</td>
<td>5</td>
<td>20</td>
</tr>
<tr>
<td>Max</td>
<td>375</td>
<td>372</td>
<td>460</td>
</tr>
</tbody>
</table>

### Intercontinental Latencies (ms)

<table>
<thead>
<tr>
<th></th>
<th>North America</th>
<th>Europe</th>
<th>Asia</th>
<th>South America</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>311</td>
<td>466</td>
<td>415</td>
<td>615</td>
</tr>
<tr>
<td>Stddev</td>
<td>81</td>
<td>83</td>
<td>138</td>
<td>48</td>
</tr>
<tr>
<td>Min</td>
<td>110</td>
<td>135</td>
<td>91</td>
<td>188</td>
</tr>
<tr>
<td>Max</td>
<td>655</td>
<td>717</td>
<td>830</td>
<td>1256</td>
</tr>
</tbody>
</table>
Bandwidth Delay Product (BDP) = latency * bandwidth

- Defines how much data must be in flight to achieve maximum bandwidth
- Worst: Denmark to Singapore: 155 Mb/s with 592 ms delay = 11 MB BDP
- Example Data Center to Data Center
  - 1 Gbit/s with 76 ms delay results in 9.5 MB BDP

### Continental BDP (MByte)

<table>
<thead>
<tr>
<th></th>
<th>North America</th>
<th>Europe</th>
<th>Asia</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.21</td>
<td>0.25</td>
<td>0.53</td>
</tr>
<tr>
<td>Stddev</td>
<td>0.45</td>
<td>0.30</td>
<td>0.46</td>
</tr>
<tr>
<td>Min</td>
<td>0.008</td>
<td>0.005</td>
<td>0.083</td>
</tr>
<tr>
<td>Max</td>
<td>2.80</td>
<td>1.60</td>
<td>1.80</td>
</tr>
</tbody>
</table>

### Intercontinental BDP (MByte)

<table>
<thead>
<tr>
<th></th>
<th>North America</th>
<th>Europe</th>
<th>Asia</th>
<th>South America</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.71</td>
<td>2.40</td>
<td>1.70</td>
<td>0.46</td>
</tr>
<tr>
<td>Stddev</td>
<td>1.40</td>
<td>3.00</td>
<td>1.60</td>
<td>0.58</td>
</tr>
<tr>
<td>Min</td>
<td>0.05</td>
<td>0.08</td>
<td>0.27</td>
<td>0.06</td>
</tr>
<tr>
<td>Max</td>
<td>6.60</td>
<td>11.00</td>
<td>5.90</td>
<td>1.90</td>
</tr>
</tbody>
</table>
Optimizing for WAN

- For low speed WAN links
  - Small SMB PDU
    - Better responsiveness for mix of file transfers and directory enumeration
  - Limit outstanding data
    - Avoid false I/O timeouts due to head-of-queue blocking
- For high speed WAN links
  - Scale the BDP
    - Each layer in the stack has a roll
  - Ensure congestion doesn’t ruin performance
- This talk will examine copying a file from a local disk to a remote disk to show all the layers involved

Windows Client Layers

- Directly affects BDP
- Affects end-to-end perf
## TCP Layer Optimizations for WAN

### Optimization categories:
- Congestion (lots of issues)
- Scale receive window from a small value to something quite large

### On Windows, can enabled/disable the following with netsh commands:
- CTCP
- ECN
- Disable receive window autotuning

<table>
<thead>
<tr>
<th>Feature</th>
<th>Vista RTM</th>
<th>Windows 2008</th>
</tr>
</thead>
<tbody>
<tr>
<td>Receive Window auto-tuning</td>
<td>Yes (up to 16 MB, wininet 256 KB)</td>
<td>Yes (up to 16 MB)</td>
</tr>
<tr>
<td>High BDP Congestion Control - CTCP</td>
<td>N (default)</td>
<td>Y</td>
</tr>
<tr>
<td>RFC 3782 - NewReno</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>RFC 2883 – SACK extensions</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>RFC 3517 – SACK based loss recovery</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>RFC 4138 – Forward RTO recovery</td>
<td>Y</td>
<td>Y</td>
</tr>
<tr>
<td>RFC 3540 - ECN</td>
<td>N (default)</td>
<td>N (default)</td>
</tr>
</tbody>
</table>
SMB2 Tuning for low speed links

- **Goals:**
  - Attempt to maintain application responsiveness
  - Attempt to not cause false timeouts on I/Os

- **Algorithms:**
  - Timer armed when SMB packet sent – not when application posts I/O.
  - Server ramps from a small number of credits to Smb2MaxCredits
    - Starts at 16, automatically scales to 128, as needed
  - Client throttles credits as a function of bandwidth
  - Dynamically vary PDU size as a function of network bandwidth
    - 0-128 Kbps = 16 KB PDU
    - 128-256 Kbps = 32 KB PDU
    - > 256 Kbps = 64 KB PDU
CopyFileEx Optimizations for WAN

- Optimizations are a balance of:
  - Virtual Address pressure (32 bit OS)
  - Non-paged pool (kernel pinned memory)
  - Filling the BDP
  - For SMB1, keeping under 64 KB PDU for read so don’t end up with 2 PDUs

### XP(SMB1)
- Synchronous 64 KB Writes
- Synchronous 60 KB Reads

### Vista SP1 (SMB1)
- Multiple async 32 KB Writes, 16 chunks
- Multiple async 32 KB Reads, 16 chunks

<table>
<thead>
<tr>
<th>File Size</th>
<th>Pipeline Depth</th>
<th>Chunk Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;= 1MB</td>
<td>1</td>
<td>File size rounded to sector size</td>
</tr>
<tr>
<td>&gt; 1MB and &lt;= 8 MB</td>
<td>2</td>
<td>1 MB</td>
</tr>
<tr>
<td>&gt; 8 MB and &lt;= 256 MB</td>
<td>4</td>
<td>2 MB</td>
</tr>
<tr>
<td>&gt; 256 MB</td>
<td>4</td>
<td>8 MB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>File Size</th>
<th>Pipeline Depth</th>
<th>Chunk Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;= 512kB</td>
<td>8</td>
<td>128kB</td>
</tr>
<tr>
<td>&gt; 512kB and &lt;= 2 MB</td>
<td>8</td>
<td>256 kB</td>
</tr>
<tr>
<td>&gt; 2 MB and &lt;= 8 MB</td>
<td>8</td>
<td>512kB</td>
</tr>
<tr>
<td>&gt; 8 MB</td>
<td>8</td>
<td>1 MB</td>
</tr>
</tbody>
</table>

**Vista RTM, For SMB2**

**Vista SP1, For SMB2**
Today’s Data for Copying a File: Varying Latency for WAN/MAN

Can we do better?

- Graph compares Windows 2008 to Windows 2003
- Latency for just WAN/MAN
  - LAN not shown
- WAN BW = 1 Gb/s
  - Theoretical is ~900 Mb/s
- All tests use production servers and networks
  - Congestion is normal
- All tests are disk to disk - RAID5, 12 disks
- Push = Write, Pull = Read

Windows 2008 vs 2003 Default Robocopy Throughput

File Size 799MB
File Size 22MB

RTT (ms)
Testing WAN using Emulation

- **Focus**: Create reliable test infrastructure with very tight variance so that small performance regressions are real (and can be automatically detected)
  - Need to take disks out of the equation – use a RAM disk instead
  - Tolerances for even smallish I/Os are pretty tight (<1-2%)

- **Hardware**:
  - 2 hosts
  - 1 gigabit LAN, h/w WAN simulator
  - 64 bit hardware, Intel Xeon, dual core, 1.6 GHz
  - 4 GB RAM (1.2 GB RAM disk)
  - Boot disk on SATA

- **Software**:
  - Windows XP (64 bit) on SMB1 client, Windows Server 2003 (64 bit) on SMB1 server
  - Vista SP1 (64 bit) on SMB2 client, Windows Server 2008 (64 bit) on SMB2 server

- **Qualifiers**:
  - BECAUSE THIS USES A RAM DISK, THEY ARE “BEST CASE” – i.e. never to be seen in the field
Comparing SMB1 and SMB2

Test details:
- SMB1 is Windows XP
- SMB2 is Vista SP1
- WAN emulator for 2 ms, 100 ms
- Direct connect for 0 ms
- Single outstanding filecopy (i.e. not multiple files)

Observations:
- SMB1 2ms filecopy on XP had serious issues (let alone 100 ms)
  - Filecopy has single ~60KB buffer outstanding

RAM DISK USED – DO NOT QUOTE THESE NUMBERS
SMB2 Issue with 12 MB BDP

Observations
- Additional latency requires larger files before start of climbing BW curve (as expected)
- 0 ms:
  - Slope changes substantially at 64 KB
  - Prefetch/full disk reads kick in
  - SMB2 pipelining kicks in
- 100 ms:
  - Bandwidth tops out due to unknown issues
  - Slope changes substantially at 8 MB
  - CopyFileEx uses 8 buffers one MB in size

RAM DISK USED – DO NOT QUOTE THESE NUMBERS
SMB2 Bottlenecks Solved

CopyFile(L→R), SMB2
Varying Latency, File Size,
Algorithm

Observations
- At 100 ms latency, Windows 2008 RTM tops out at ~250 mb/s
- Experiments:
  - Removing credit throttling due to BW estimation issue moves BW to ~500 mb/s
  - Removing credit throttling and increasing credits to 512 moves to ~800 mb/s

RAM DISK USED – DO NOT QUOTE THESE NUMBERS
Analyzing Real Networks

- **Disk-to-disk transfers**
  - Raid5, 12 disks

- **WAN Characteristics**
  - 70 ms, 1 Gb/s WAN = 9.5 MB BDP
  - Production network (losses are normal)

- **Top graph: File Copy**
  - Poor performance was due to wide BW variance over single robocopy transfer (average ~250 Mb/s)
  - Poor interactions between SMB bandwidth throttling, TCP congestion window

- **Bottom graph: NT-TTCP**
  - Three NTttcp transfers in a row
  - TCP doesn’t recover well under some loss events combined with high BDP
Source Disk Bottleneck

- For SMB2, source disk for filecopy must be able to keep up

**Windows 2008 Robocopy Throughput (Mbits/sec) RTT = 76ms Push File size is 4.45GB**

<table>
<thead>
<tr>
<th>Time (HH:MM:SS)</th>
<th>Throughput (Mbits/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10:48:34</td>
<td>0</td>
</tr>
<tr>
<td>10:48:41</td>
<td>0</td>
</tr>
<tr>
<td>10:48:48</td>
<td>0</td>
</tr>
<tr>
<td>10:48:55</td>
<td>0</td>
</tr>
<tr>
<td>10:49:02</td>
<td>0</td>
</tr>
<tr>
<td>10:49:09</td>
<td>0</td>
</tr>
<tr>
<td>10:49:16</td>
<td>0</td>
</tr>
<tr>
<td>10:49:23</td>
<td>0</td>
</tr>
<tr>
<td>10:49:30</td>
<td>0</td>
</tr>
<tr>
<td>10:49:37</td>
<td>0</td>
</tr>
<tr>
<td>10:49:58</td>
<td>0</td>
</tr>
<tr>
<td>10:50:05</td>
<td>2 physical disks</td>
</tr>
<tr>
<td></td>
<td>configured in RAID 1</td>
</tr>
<tr>
<td>10:50:12</td>
<td>0</td>
</tr>
<tr>
<td>10:50:19</td>
<td>0</td>
</tr>
<tr>
<td>10:50:26</td>
<td>0</td>
</tr>
<tr>
<td>10:50:33</td>
<td>12 physical disks</td>
</tr>
<tr>
<td></td>
<td>configured in RAID 5</td>
</tr>
<tr>
<td>10:50:40</td>
<td>0</td>
</tr>
<tr>
<td>10:50:47</td>
<td>0</td>
</tr>
<tr>
<td>10:50:54</td>
<td>0</td>
</tr>
</tbody>
</table>

2 physical disks configured in RAID 1
12 physical disks configured in RAID 5.
Problems Solved

- Fixed bugs in TCP congestion window
- Fixed bug in SMB2 credit throttling
- Extended maximum credits to 512

Windows 2008 x64 QFE pull
70ms RTT, 799.64MB File
Final Production Data with QFE

- Installed SMB2 and TCP QFE and increased SMB2 max credits
- Data collected between many different data centers to test different latencies

![Robocopy Throughput Comparison](image-url)
Conclusions on WAN Performance

- BDP theory matches practice for finding bottlenecks
  - Credit management and credit throttling issues resolved
  - Each layer of the stack must be optimized for BDP

- TCP congestion control with loss is tricky, and Windows 2008 was not initially optimized for long, fat pipes (100 ms, 1 gigabit)

- Branch analysis
  - Latency rules of thumb
    - Continental: ~100 msec
    - Intercontinental: ~500 msec
  - Average branch BDP today is ~ 1 MB, and growing
Questions?