

1N-1 ワークステーションサーバの性能監視方法とパフォーマンスチューニング

佐藤弘康 伊藤琢巳

NTTPC コミュニケーションズ

1 はじめに

NTTPC コミュニケーションズでは WWW、メールサービスを提供する仮想ホスティングサービスを運用している。サービス品質を定量的に評価するため、サービスレベルの設定、サーバの性能監視の実施、設定したサービスレベルとの比較を行った。また性能監視の結果よりシステム毎の負荷の傾向を分析し、アプリケーションのチューニングを行った。

2 サービスレベルの設定

システム導入時にサービスレベルを以下のように設定した[1,2]。

2.1 CPU

システムに負荷のかかっている時間帯で CPU 使用率の system が 30%以下、user が 70%以下、idle が 15%以上であること。

2.2 メモリ

ページアウト、スワップアウトがサービスの提供に悪影響を与えないこと。

2.3 ディスク

パーティションのレイアウトを考慮し、パーティション毎のディスク使用量はパフォーマンス面から 60%を限度とする。

2.4 ネットワーク

LAN のコリジョンが送信パケット数の 10%以下であること。送信時のエラーパケット数が全送信パケット数の 0.025%以下であること。TCP の再送信セグメント数が全送信セグメント数の 15%以下であること。

3 サーバの性能監視方法

プロトコルには SNMP(Simple Network Management Protocol)を使用している。監視対象サーバ側の SNMP エージェントとして UCD-SNMP を、また取得したデータの加工に MRTG を使用している。

4 性能監視の実施と結果

ロードアベレージ、メモリ使用率、スワップ使用率、ディスク使用率、ネットワーク、プロセス数について監視を行った。以下にその結果を示す。

4.1 CPU

CPU の使用率は最繁時間帯で system が 25%、user が 50%、idle が 25%程度である。瞬間的に idle が 0%になることもある。最繁時間帯以外では idle が 70%以

上である。

4.2 メモリ

Linux では、RAM を常時 100%近く使用する。ページアウトは常時、数十 page/sec 程度発生している。数百 page/sec 程度のピークが 0~3 回/日程度発生している。これにより RAM に 50~400MByte 程度の空き容量ができる。

スワップアウトは、大容量の RAM を搭載したシステムではほとんど発生しない。RAM の著しく消費するサーバで 2~3 回/年程度である。

4.3 ディスク

一般的にディスク使用率の変動がもっとも大きいのは /var である。その要因は、ログファイルの増加とそのローテート、およびメールキューである。

4.4 ネットワーク

転送データ量については、ピークで 3M~5Mbps 程度である。送信時のエラーパケットについてはいずれのサーバでも発生していない。LAN のコリジョンは、全送信パケット数に対し 1~2%程度である。また TCP の再送信は、全送信セグメントに対し 5%~7%程度である。

5 アプリケーションにおけるパフォーマンスチューニングと効果

パフォーマンスチューニングにおいては、アプリケーションのチューニングが有効である。理由としては直接ユーザにサービスを提供しているのはアプリケーションであること、またアプリケーションにおけるチューニングはハードウェアにおけるそれと比較し、低コストであることがあげられる。

5.1 アプリケーションから見た負荷の傾向

監視対象サーバの負荷傾向は 2 種類に分類できる。1 つは HTTP のアクセスに起因するトラフィックの増加である(図 5.1)。もう 1 つは CGI(Common Gateway Interface)の実行に起因するロードアベレージの増加である(図 5.2)。メール(SMTP、POP)および FTP については、起動されるプロセス数も少なく、定常的な負荷の要因とはなっていない。

5.2 httpd のチューニング例

httpd.conf において、パフォーマンス上重要な設定は KeepAlive、KeepAliveTimeout および MaxClients

である。KeepAliveTimeout をデフォルト値の 15 から 5 に変更すると MaxClients の設定にもよるが、httpd のプロセス数が約 10% 減少する。また KeepAlive をデフォルト値の on から off に変更すると httpd のプロセス数が半分程度に減少する。またロードアベレージが低くかつ MaxClients の設定により httpd のプロセス数が頭打ちしている場合には、MaxClients の値を増加することにより、システムはより多くのリクエストを処理することが可能となる。

5.3 CGI のチューニング例

CGI の実行に起因するロードアベレージの増加については、負荷の原因となっている CGI の改修、あるいは CGI の実行権を停止することにより対応している。また、負荷が高い状態で CGI を 10 以上起動しているドメインに対し、3 秒間 CGI の起動を抑止するようにアプリケーションを改修している。

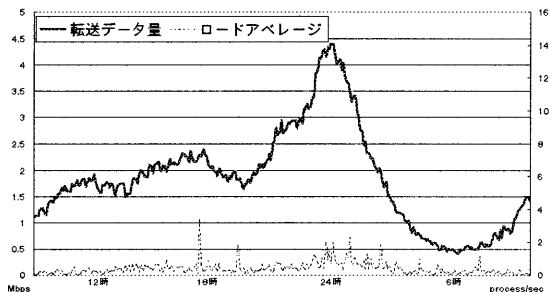


図5.1 サーバA

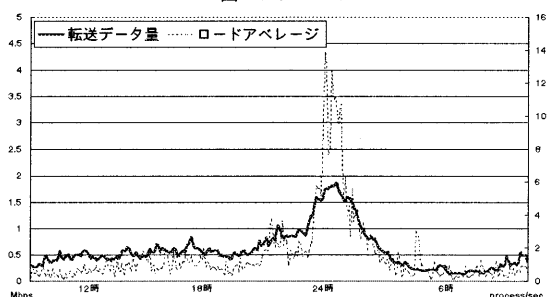


図5.2 サーバB

6 考察

CPU の使用率については、瞬間的に idle が 0% になることがある。サーバ 1 台あたり約 250 ドメインを収容していること、およびインターネットにおけるトラヒックの時間的な偏りを考慮すると致し方ない。CPU の使用率については最大値ではなく、最繁時間帯の 3 時間程度の平均を用いて評価する必要がある。

Linux においては、RAM の使用量の変動に注目する必要はない。しかしスワップアウトが発生すると、ロードアベレージが急増するため、注意が必要である。スワップサイズについては、パニックダンプを収容できる大きさ(RAM の 1.1 倍程度)があれば良い。

ディスク使用率については、/var に注目する。/var のディスクフルは、直接サービスに影響を与えることになる。

ネットワークのエラー、コリジョンについて注目する必要はない。これは、インターフェイスが 100Mbps の全二重であること、および対向がスイッチングハブであることによる。TCP の再送信については、中継経路および端末の影響を受けるため、サーバにおける対処は困難である。しかし、TCP の再送信が 15% 以下と言うのは 1 つの指標になる。

httpd のキープアライブを短縮または設定解除したことによるロードアベレージの増加はほとんど見られない。キープアライブの設定変更による負荷の増加と httpd のプロセス数の減少による負荷の減少が相殺されたためと考えられる。

7 まとめ

サーバの性能監視を実施することにより、サーバ毎の負荷状況とその傾向を把握することが可能になった。また、負荷に対処すべくアプリケーションのチューニングを実施し、httpd については成果があった。CGI については今だ有効な解決がなされていないため、これについては今後の課題としたい。

参考文献および URL

- [1] Mike Loukides/ 著 砂原秀樹/ 訳
UNIX システムチューニング/ 株式会社アスキー
- [2] Greg Schmitz, Allan Esclamado/ 著
サンシステムのパフォーマンス:
システムチューニング
サンマイクロシステムズ株式会社
- [3] 武智洋, 村本衛一, 宇夫陽次朗/ 著
PC UNIX のベンチマーク 1-3
UNIX MAGAZINE 1999.5-7
株式会社アスキー
- [4] Mohammed J. Kabir/ 著 遠藤美代子/ 訳
Apache Server Bible/ 株式会社 IDG ジャパン
- [5] UCD-SNMP
<http://net-snmp.sourceforge.net/>
- [6] MRTG
<http://ee-staff.ethz.ch/~oetiker/webtools/mrtg/>