Printing Billing Options

29th April 2003

1 Wasp

Here is how Chris Leishman (the author) describes Wasp:

> Um - you've got the source ;-)
>
> Basically it provides a reasonably flexible way of doing the following:
>
> - Locking print queues and requiring specific authorization to spool jobs.
>
> - Redirection of jobs between printers.
>
> - Collecting page counts for jobs and calculating costs based on specific printer.
>
> - Modularized 'charging' of pages to users (a berkley DB module is provided, and an LDAP one has been written I think).
>
> - Modularized 'quota' management of users.
>
> I don't think there's many other systems to compare with. I guess the best comparison would be just against LPRNG itself, and the following are the advantages:
>
> - already implements much of what would be required to author in an LPRNG setup.
>
> - allows for queue control, such as authenticating users before jobs are spooled and allowing jobs to be redirected between queues. This is useful as it separates the authentication required for billing (eg. authenticating who printed something) from the standard login authentication, which is difficult to trust in the college environment. It is extremely common for people to print from someone else's account (eg. because the computer room machine was already logged in, etc) or to send a job from a system that can't easily be authenticated (eg. old mac systems used to just authenticate as guest for printing purposes). There is an argument that the standard authentication systems should be sufficient as long as people 'do the right thing', but experience tells me that it doesn't work that way and so adding an extra layer of authentication prevents the sticky situation of people contesting bills on their account.
>
> - Already implements much of the billing system, such as quota checks, etc.

2 Printbill

From the PrintBill website:

> This is a print quota and billing program with associated administration utilities and detailed statistics collection functionality. It has the following major features:
>
> * Charge rates may be specified on per-page and per-percent-coverage;
>
> * Different charge rates for different printers are possible;
>
> * CMYK colour printers are supported (separate charge rates for both colour and black ink);
>
> * Out-of-order execution has been implemented - you may prioritise jobs on the basis of size (jobs larger than a threshold can get lower priority or higher priority as desired) and jobs are billed in parallel / overlapping: jobs which finish billing first get printed first. This is not necessarily the same as the order of arrival. This also sets the scene for distributed processing, which will appear in future versions (hopefully 3.1);
>
> * Detailed stats are collected for job size, CPU time for each job, page count and ink/toner coverage;

* Filters are provided for accounting/stats collection only (jobs are spooled very quickly and processed by ghostscript later) if you only want to track printing habits;

* You can safely bill people for multiple printers off one machine - it is not perfect but it works reasonably well on a fast machine with not-too-many print queues.

* It supports an optional user-supplied anything-to-postscript filter, so you can get properly billed for plain text, DVI files, jpegs, and so forth as well as PostScript;

* You can find out how much a job is going to cost, either by running printquote or by printing to a dummy printer with the printbillquote filter.

* Optional web interface for administration of print quotas. USE WITH CAUTION - it is STRONGLY recommended that this be used in conjunction with apache-ssl, so that unencrypted passwords do not fly around the Internet willy-nilly. At the very least restrict the hosts which may execute this CGI script, and restrict the number of concurrent connections. Authentication is delayed by a few seconds to limit brute-force attacks - be sure *not* to use short weak passwords.

* Separate per-user databases maintain records of remaining quota, pages printed, cumulative money paid into the system and cumulative money actually spent. Global records are also kept for these items, as well as the estimated "percentage coverage" used up so far.

Other important notes:

* Programs which are setuid "daemon" (or whatever lpd_user is set to in the configuration file) can theoretically print for free if they send their jobs to the primary print queue. This is not really something which we can avoid. Solution - be careful with setuid binaries!!! There may be other security holes, I'm not a perl security guru by any means. It's probably not ideal in an environment with hostile idiot users (malicious postscript code could probably allow DOS attacks on your sever, but at least everything should get logged).

* The charge rates are defined in /etc/printbill/printbillrc. Per-page and per-percent-coverage (black & colour) charges are supported. If you have a flat per-page rate, just set the per-percent-coverage charge to zero and adjust the per-page rate accordingly (GS can still be used to precisely determine how many pages you have used). If you only want to account for printing, set the both rates to zero, and set the quotas to -1 for users whom you don't want to be able to print. I've not actually tried this but I assume it would work. Remember, it's your toner cartridge.

* Yes, it probably *is* buggy, full of security holes, will fill your server up with rubbish and reformat your hard drive. If it does that or anything else which it shouldn't, send your bug reports to the principal author. It is currently being used in at least 3 low-volume environments and has not caused any major problems. Yet.

How it works

When a job arrives at the head of the user-accessable print queue, the scheduler filter (printbill_scheduler(1)) spawns a process which first pre-processes batches of files which have been sent to a printer with an optional anything-to-PostScript converter, uses ghostscript to generate a set of PNG images (one for each page) then runs the percentblack or percentcolour programs (depending on the type of printer, which may either be a default printer or one explicitly described in printbillrc (5)) on each one to determine page coverage.

If users can cover the cost, the document is printed to the secondary print queue (ordinary users' jobs will be removed from this queue, only jobs from lpd_user will be accepted - see printbillrc (5)) and the charge deducted from the user's account. If the user cannot afford the cost, or if the job was rejected for some other reason, the user is e-mailed with a brief yet vaguely informative message.

The configuration file printbillrc(5) includes a parameter limiting the maximum number of concurrent billing processes. If desired, this may be limited to one or two concurrent processes - two will be sufficient to deal with the head-of-line problems for small systems, although more shouldn't cause a problem. The pool of available processes is shared amongst all printqueues, so even if multiple printers are managed through a single server, the total number of threads remains limited. If all process slots are filled, pending jobs from different queues have an equal likelyhood of getting access to the next available process slot.

3 LPRNG inbuilt accounting

LPRNG (the print spooling software) includes some inbuilt accounting, keeping records of the pages printed by login name, printer and date. The main disadvantages have already been outlined in Chris's description of Wasp above.

4 JetCAPS FollowMe and MegaTrack

From the JetCAPS website:

> JetCAPS FollowMe Printing
>
> JetCAPS FollowMe Printing intercepts documents printed from any Windows PC, and holds them until the correct recipient identifies him- or herself at any of the printers designated for their use, before outputting to paper. Individual users can also be assigned credit limits, to prevent the misuse of expensive resource s or consumables, and designated time zones to allow printing or copying at cert ain times only.
>
> How JetCAPS FollowMe Printing works
>
> JetCAPS FollowMe Printing starts with the port monitor, on the user's computer, that routes print jobs to the FollowMe Queue Server software, which runs on a Windows PC. The JetCAPS FollowMe Printing identification device is connected to t he LAN and located close to any networked HP printer. Depending on the version of JetCAPS FollowMe Printing, the identification device uses PIN codes, fingerpr ints, magnetic swipe cards, or proximity cards to identify the user and release the print job. Between submitting the print job and releasing it, the user can control it using the JetCAPS FollowMe Client software on their PC, which lets th em Print & Retain, Print & Delete, Delete, view or print the job list, manipulat e document details, or request additional copies after proof reading.
>
> JetCAPS MegaTrack
>
> JetCAPS MegaTrack monitors printers connected to local and remote print servers to provide you with comprehensive real-time and statistical information on usage and costs - broken down according to the department, cost-center, group or even employee or printer - to help you optimise printer deployment, workload and the purchase of printer consumables.
>
> How JetCAPS MegaTrack works
>
> The JetCAPS MegaTrack print-server (MPS) monitors the printers attached to as many local and remote Windows NT/2000 and XP print-servers as necessary. It scans the print jobs and queries the printer to compile detailed information on print er usage. BY analyzing this information MPS calculates a cost per page based on a detailed printer profile that includes toner and media cost, depreciation and a host of other factors. The result is accurate information for each print job (color, duplex mode, economode ...). The central JetCAPS Megatrack database-server (MDS) consolidates the information collected at the individual print-servers. Then JetCAPS MegaTrack web-server so ftware (MWS) prepares detailed, customized reports, combining graphics and text that show you the composition of your printing costs.

These software products only allow printing from Windows computers. If we were to implement one of these, we would be removing printing abilities from the significant section of College who run MacOS, Linux, BSD, etc computers. Also, these products require clients to be installed on user PCs, which would be a large amount of work for IT administrators and means that the charging can quite possibly be easily cirumvented.

If we were to implement one of these software packages, we would also need to purchase additional Windows print servers, database servers and web servers.

Evaluation

- PrintBill offers two advantages over raw LPRNG accounting - coverage charging, and pre-paid quotas. Coverage charging is probably unneccessarily complicated for users, and quotas are irrelevant since printing is charged to College accounts, rather than signed up in batches.
- PrintBill does not solve the problem of anonymous printing, or printing from another account. Only Wasp does this.
- MegaTrack is very similar to PrintBill, except that it requires Windows NT/2000/XP on the print servers, and on the database servers, and as a webserver. Its main advantage is that it can query the printer and include depreciation and "a host of other factors" in the cost.
- FollowMe is very similar to Wasp, except that it requires a client to be installed on any computer that wishes to print, and the client is only available for Windows systems. It also requires a Windows NT/2000/XP server, and authentication devices at every printer - we would need to implement printing swipe cards or a similar system.

- We could disable anonymous printing, but old Mac users would no longer be able to print. These constitute about 20% of college - quite a large proportion.
- The only way to accurately account for all printing is to install something that requires authentication at the printer - that is, Wasp or FollowMe.
- There is no need to implement an overly complex, confusing and costly proprietary system (such as FollowMe) when we have the option of a suitable Open Source alternative that provides equivalent functionality where it is required (such as Wasp).

I would recommend that we leave things as they are until we can install Wasp, which should be a high priority once the new servers are live.