

ISPadmin

Billing and Provisioning

Introduction

In this installment of ISPadmin, I look at the aspects of billing and provisioning in an ISP environment. Billing is the process by which the customer pays for his/her service, and is not unlike billing in other service industries such as electrical or cable TV service. As such, the challenges in the ISP environment with respect to billing are:

- Defining bill plans
- Accumulating usage
- Generating bills
- Posting payments
- Interface into the provisioning process

Provisioning, while a separate process from billing, is closely related to it. Provisioning is the process by which services are enabled on the back end systems which provide the actual services to the end customer. For example, when a customer orders a generic ISP dial up account with a POP box and home page, the following actions must be taken:

- An email account is created on the POP mail server, and other related systems are updated as necessary (for example, mail relay(s))
- The dial-up RADIUS (PPP) account is created
- A web home page account is created (usually of the form `www.isp.net/~username`)
- Access to the web home page account is enabled and optionally, an anonymous FTP area is created
- Disk quotas are created on every system with customer data
- A record is created in the billing system which indicates the master account owner, billing information for the master account, associated services billed to the master account, and bill plans for those services

As you can see, this is not a straightforward process! Provisioning and billing are quite tailored to the ISP business, but billing is less specific to the ISP industry. However, while people can and do utilize generic billing systems for ISPs, the best results are achieved with a billing system that has been designed expressly for service providers, and especially those written specifically for ISP's.

One challenge faced when considering an ISP billing/provisioning system is the fact that such systems touch on most if not all areas of an ISP's operations, including: finance, customer service, technical support, back end business support systems, etc. I will focus on only the billing and provisioning aspects in this installment.

Small Provider Goals

A small provider is primarily concerned about cost when it comes to designing and implementing a billing/provisioning system. This means a typical small ISP:

- Utilizes a home grown or low cost billing system, without high end features like real time billing data or real time provisioning capability
- Performs account adds/changes by hand on each back end system, or has written their own limited function non-real time provisioning system in house (which may or may not be integrated with the in house billing system)

A small provider doesn't have the volume of account data that a large provider has, so it is feasible to do many processes via MS Excel spreadsheets (or pencil and paper for that matter) and manual input. The same also goes for billing, where there may be no need for the billing system to input RADIUS usage data in order to produce bills. (Whether or not to utilize RADIUS data is determined largely by what business model the ISP utilizes.) Not utilizing RADIUS accounting records makes things much easier for the smaller provider who doesn't bill for usage.

Of course, accuracy is very important! No one is going to stay in business if they bill customers incorrectly, or aren't

billing their customers at all. Another consideration is what happens when an attacker adds an account to a system under automated provisioning control. The account will get deleted the next time the provisioning process runs. Of course, if the hacker breaks into the provisioning system, and there is no way of comparing the accounts which are being billed against the accounts that have been enabled for services, a company won't be in business long. Audit trails and mechanisms are a great asset operationally as well as from a financial control standpoint, for any system administrator, and especially so at ISPs.

Large Provider Goals

A larger provider who services retail customers has different goals than a smaller provider when it comes to billing and provisioning:

- Functionality (for example virtual ISP [VISIP] and real time billing support)
- Flexibility (how easy and expensive is it to maintain and change over time)
- Reporting
- Cost, but much less sensitive than a smaller provider

Bigger ISPs are much more likely to purchase one of the many commercial billing and provisioning solutions. The only exception to this could be a large wholesale dial up ISP, who might be able to get away with a simple billing/provisioning system with the following limitations:

- No VISIP offering, therefore no need to provide end user services (such as email, RADIUS authentication, and web hosting) and/or bill end users
- RADIUS authentication is via proxy to downstream customers

Of course, not having a fully functional billing system may cause limitations on the growth of your ISP business (as someone once called it, "revenue limiting"). But it can be (and is currently) done this way at some ISPs.

Small Provider Design

Figure 1 contains a diagram that outlines the major pieces in a small dial up ISP's billing and provisioning process, and systems that provide services to end customers. This diagram is illustrative of a small provider, with an automated provisioning/billing system. However, the basis of the design is valid for larger systems, with appropriate scaling mechanisms.

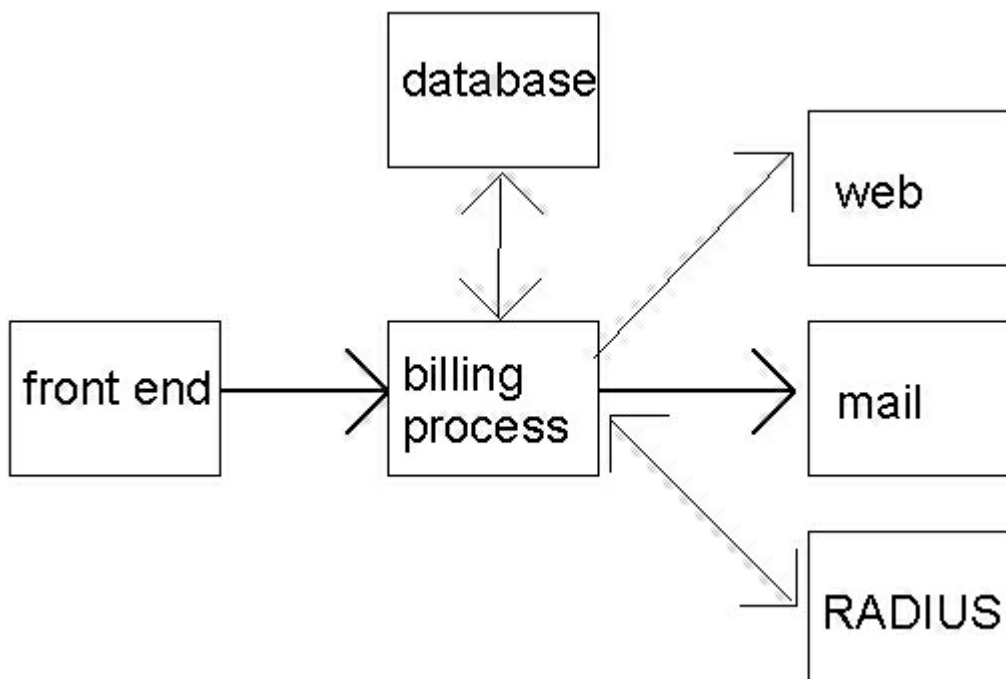


Figure 1

Of course, the center of the system is the billing/provisioning system. For a simple system, this would be a single machine. In the case of a commercial application like Portal, this usually takes the shape of a series of machines. In either case, they both would communicate with an external Oracle, MS SQL or similar database for a back end data store. The flow of data would be two way between the database and billing system.

The box labeled “front end” could be a script local to the billing machine, or web site that inputs data into the provisioning/billing system. It might be on a separate stand alone system, or integrated on the billing platform itself. This front end may also include a web registration system, bulk account registration system, specific customer support screens and/or other functionality.

The boxes to the right, labeled “web,” “mail,” and “RADIUS” (the flow heading towards the box) indicate the provisioning output aspect of the system. This is the billing system creating accounts on the back end systems so customers can access their web sites, email and dial into the Internet. The “RADIUS” box with flow heading out of the box shows RADIUS accounting data flowing into the billing system in order to calculate end subscriber usage.

Large Provider Design

Figure 2 contains an illustration of a possible design of a commercial billing system such as Portal Software’s Infranet. Please note that there are many, many ways to implement commercial billing systems, and this discussion is meant as a high level overview of what is possible. If you are really interested in this topic, I would suggest contacting a major ISP billing company and talking to one of their sales engineers.

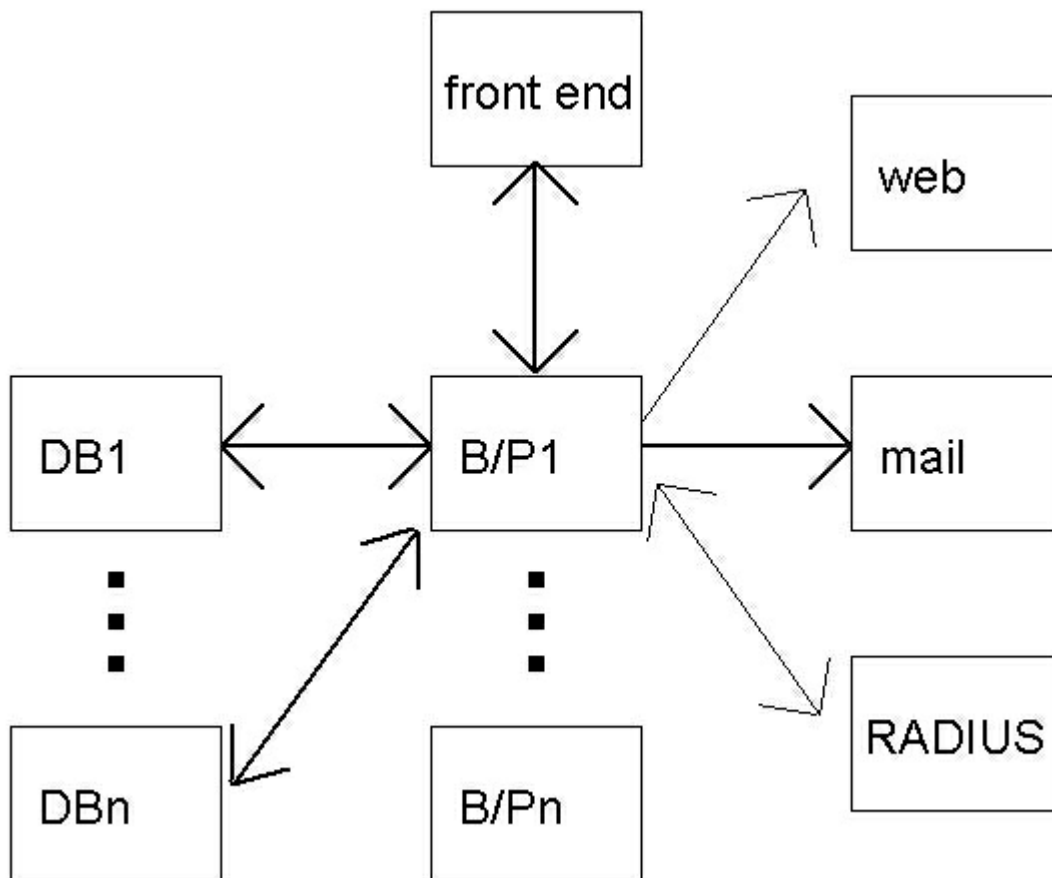


Figure 2

The essential difference between a small provider and large commercial implementation is the ability to scale the database and billing/provisioning server machines. In Figure 2, the boxes marked “DB1” and “DBn” indicate multiple

machines housing the data store. Of course, large database implementations such as Oracle scale relatively easily. Boxes marked “B/P1” and “B/Pn” indicate the multiple billing/provisioning machines that can be utilized for performance and scaling under heavy load and high user counts.

Lightweight Directory Access Protocol (LDAP)

No discussion pertaining to provisioning would be complete without at least a mention of LDAP. For those of you who have read previous installments of ISPadmin, you know that LDAP eliminates much of the work when it comes to provisioning. LDAP is a central repository for authentication and account configuration (for example, what host a customer’s POP mail is on, the spool directory location, etc.). The issue with LDAP has been the lack of support at the application level for LDAP. However, this is slowly changing over time. Full application support of LDAP remains the “holy grail” for many people both in the ISP business, and to some degree, within the entire IT arena.

Open Source Solutions

A search of Freshmeat and/or Sourceforge with the phrase “ISP billing” shows numerous hits. However, many of the listed programs are vaporware, little more than concepts. Some are specific to web hosting or not related to ISP billing at all. Here is a list of a few open source dial up ISP billing and/or provisioning systems with available source code:

- Freeside from Silicon Interactive Software Design, Inc.
- ISPman from Atif Ghaffar
- gcdb
- ISFree from Brian Wolfe
- ISP daemon

While I don’t have space to discuss them all, I will briefly cover Freeside and ISPman.

Freeside

Of the open source billing/provisioning applications, Freeside has been around the longest and is probably the most feature rich and stable. It is a complete billing and provisioning program written in Perl. Freeside has the following features (from the Freeside web site):

- Utilizes Perl’s DBI module; recommended database back ends are PostgreSQL and MySQL
- Web based interface
- Reseller/agent support, including controlling what agents can sell
- Tax rates by state and locale
- Exports to a multitude of Unix file formats, including:
 - o passwd and shadow (or master.passwd) files
 - o ERPCD acp_passwd and acp_dialup files
 - o RADIUS users files
- Works with ICRADIUS and Radiator RADIUS servers for native SQL authentication
- Virtual domain support
 - o Exports Sendmail virtusertable and sendmail.cw files
 - o Exports Qmail virtualdomains, rcpthosts and recipientmap files
- Signup server support with MS IE auto-configuration support
- Real time credit card support

Freeside would be a good choice for an ISP on a limited budget. One major limitation is the lack of LDAP support. Also, scalability might be an issue, unless Oracle or other easily scalable back end database is utilized.

ISPman

ISPman is a relative newcomer to the open source scene, and does not have a billing component. It is strictly for provisioning accounts, with LDAP as a back end. It is still under active development and therefore may not be suitable for your needs. However, it is worth investigating, at least as a starting point for your own development. In fact, I firmly believe that the future for ISP provisioning lies with an end to end LDAP solution, and Atif Ghaffar is off to a great start. The features of ISPman include:

- Full LDAP support; all back end apps are LDAP aware
- Creates DNS configuration files to input into a BIND 8 server
- Web interface for all input
- End subscriber management

Required back end software to run ISPman include:

- OpenLDAP
- Postfix
- Cyrus imapd and SASL
- Apache
- Proftpd (GNU's FTP server)
- BIND
- pam_ldap
- IMP 2.2 (PHP Webmail application)

Commercial Billing/Provisioning Applications

There are a number of commercial billing/provisioning applications on the market. A complete discussion of all of the options is out of the scope of this article, but a brief discussion is worthwhile.

At the lower price range, there are NT based solutions which focus exclusively on ISPs and don't have features like real time accounting or complete support for wireless, DSL or other such enhanced services. A few of the better known players in this space are:

- Rodopi
- Boardtown Platypus

At the more expensive end of the spectrum there are a number of players; here are a few examples:

- Portal Infranet
- Amdocs/Solect
- Lucent/Kenan

The trend at the higher end of the billing/provisioning market is towards convergence. This is where telcos and service providers can utilize one platform for billing and provisioning all of the services they offer, including wireless, wireline, ISP/ASP and others. Historically, telcos and larger service providers have implemented provisioning/billing platforms for each line of their business, which leads to high maintenance costs. (If you investigate this area, you will find that Lucent and other players who came out of the telco market call billing "Business Support Systems" or BSS and provisioning "Operational Support Systems" or OSS.)

There is an effort to open the (historically closed) interface between providing end subscriber services and billing for those services by defining an open protocol. IPDR.Org is a consortium of leading billing/provisioning software vendors who is developing this standard. CLEC Planet also has a short article which talks about the current status of billing/provisioning convergence.

Outsourcing

No discussion of ISP billing would be complete without touching upon the area of outsourcing. There are several companies who will, for a monthly fee, do your billing, customer care, provisioning etc. for you. This may be an option for a larger ISP, as a smaller provider wouldn't have the numbers of subscribers to make this a viable option for them.

Typically, these billing companies have implemented Portal, Kenan or other such commercial billing system in a "virtual" manner. Then, the outsourcing company will customize their system to accommodate you. They typically sell "al la carte" and will provide just the parts you need (only billing, billing plus provisioning, etc.). GlarNet is a company that provides an outsourced ISP billing and provisioning system.

Conclusion

Billing and provisioning is a very important aspect of an ISP's operation. For a smaller ISP, billing and provisioning by hand is acceptable. But for a medium or large ISP, billing and provisioning systems usually are automated, either by building a system in house, deploying an open source solution like Freeside or implementing a commercial billing system like Rodopi or Portal. A third option is to utilize an ISP billing outsourcing company, and pay a service fee month to month.

Next time, I will look at ways ISP's deal with unsolicited commercial email or spam. In the meantime, please send your feedback on ISP topics and system administration to me!

References

Oracle: <http://www.oracle.com>

Portal Software: <http://www.portal.com>

LDAP starting point: <http://www.ldapman.org>

Freshmeat: <http://freshmeat.net/>

Sourceforge: <http://sourceforge.net/>

Freeside: <http://www.sisd.com/freeside/>

ISPman: <http://ispman.sourceforge.net/ispman.php3>

ISPman article by Atif Ghaffar (ISPman author): <http://www.linuxfocus.org/English/September2000/article173.shtml>

gcdb: <http://sourceforge.net/projects/gcdb/>

ISFree: <http://advogato.org/proj/ISFree/>

ISP daemon: <http://ispd.eburg.com/>

Perl DBI: <http://dbi.symbolstone.org/index.html>

PostgreSQL: <http://www.postgresql.org/index.html>

MySQL: <http://www.mysql.com/>

ICRADIUS: <ftp://ftp.cheapnet.net/pub/icradius/>

Radiator: <http://www.open.com.au/radiator/>

Sendmail: <http://www.sendmail.org/>

Qmail: <http://www.qmail.org/>

MS Internet Explorer Administration Kit (IEAK): <http://www.microsoft.com/windows/ieak/en/default.asp>

OpenLDAP: <http://www.openldap.org/>

Postfix: <http://www.postfix.org/>

CMU's Cyrus IMAP server: <http://asg.web.cmu.edu/cyrus/imapd/>

CMU's SASL library: <http://asg2.web.cmu.edu/sasl/>

Apache: <http://httpd.apache.org/>

ProFTPD: <http://proftpd.org/>

ISC's BIND: <http://www.isc.org/products/BIND/>

pam_ldap: http://www.padl.com/pam_ldap.html

IMP: <http://www.horde.org/imp/>

Rodopi: <http://www.rodopi.com/>

Boardtown's Platypus: <http://www.boardtown.com/>

Solect from Amdocs: <http://www.solect.com/>

Amdocs: <http://www.amdocs.com/>

Lucent: <http://www.lucent.com/software/>

IPDR.Org: <http://www.ipdr.org/index.htm>

CLEC billing convergence article: <http://www.clec-planet.com/tech/0004phifer.htm>

GlarNet: <http://www.glarnet.com>