MAGAZINE

BSD

FOR NOVICE AND ADVANCED USERS

BSD CERTIFICATION

HOW? WHEN? WHY?

INSIDE

- CUSTOMIZING YOUR PC-BSD 9.0 DESKTOP
- THE MIDNIGHTBSD PACKAGE MANAGEMENT TOOLS
- WHAT CAN'T YOU DO ON THE COMMAND-LINE?
- POSTGRESQL: FROM INSTALLATION TO PITR
- LOAD BALANCERS. ENTERPRISE LOAD & SERVICE AVAILABILITY
- ANATOMY OF FREEBSD COMPROMISE PART 3
- DATA CLASSIFICATION POLICY

VOL.5 NO.02
ISSUE 02/2012(31)
1898-9144

TrueNAS™ Storage Appliance: You are the Cloud

With a rock-solid FreeBSD® base, Zettabyte File System support, and a powerful Web GUI, TrueNAS™ pairs easy-to-manage software with world-class hardware for an unbeatable storage solution.

Expansion Shelves Available

TrueNAS™ 2U System

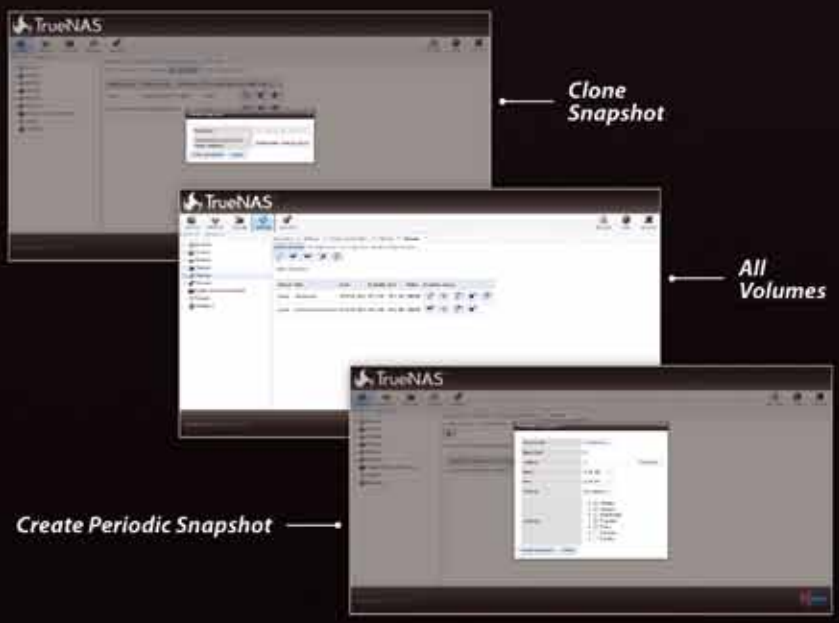
TrueNAS™ 4U System

FreeNAS 8 POWERED

Storage. Speed. Stability.

In order to achieve maximum performance, the TrueNAS™ 2U and 4U Systems, equipped with the Intel® Xeon® Processor 5600 Series, support Fusion-io's Flash Memory Cards and 10GbE Network Cards. Titan TrueNAS™ 2U and 4U Appliances are an excellent storage solution for video streaming, file hosting, virtualization, and more. Paired with optional JBOD expansion units, the TrueNAS™ Systems offer excellent capacity at an affordable price.

For more information on the **TrueNAS™ 2U** and **TrueNAS™ 4U**, or to request a quote, visit: **http://www.iXsystems.com/TrueNAS.**



TrueNAS™ 2U KEY FEATURES

- Supports One or Two Quad-Core or Six-Core, Intel® Xeon® Processor 5600 Series
- 12 Hot-Swap Drive Bays - Up to 36TB of Data Storage Capacity*
- Periodic Snapshots Feature Allows You to Restore Data from a Previously Generated Snapshot
- Remote Replication Allows You to Copy a Snapshot to an Offsite Server, for Maximum Data Security
- Software RAID-Z with up to triple parity
- 2 x 1GbE Network Interface (Onboard) + Up to 4 Additional 1GbE Ports or Single/Dual Port 10GbE Network Cards

TrueNAS™ 4U KEY FEATURES

- Supports One or Two Quad-Core or Six-Core, Intel® Xeon® Processor 5600 Series
- 24 or 36 Hot-Swap Drive Bays - Up to 108TB of Data Storage Capacity*
- Periodic Snapshots Feature Allows You to Restore Data from a Previously Generated Snapshot
- Remote Replication Allows You to Copy a Snapshot to an Offsite Server, for Maximum Data Security
- Software RAID-Z with up to triple parity
- 2 x 1GbE Network Interface (Onboard) + Up to 4 Additional 1GbE Ports or Single/Dual Port 10GbE Network Cards

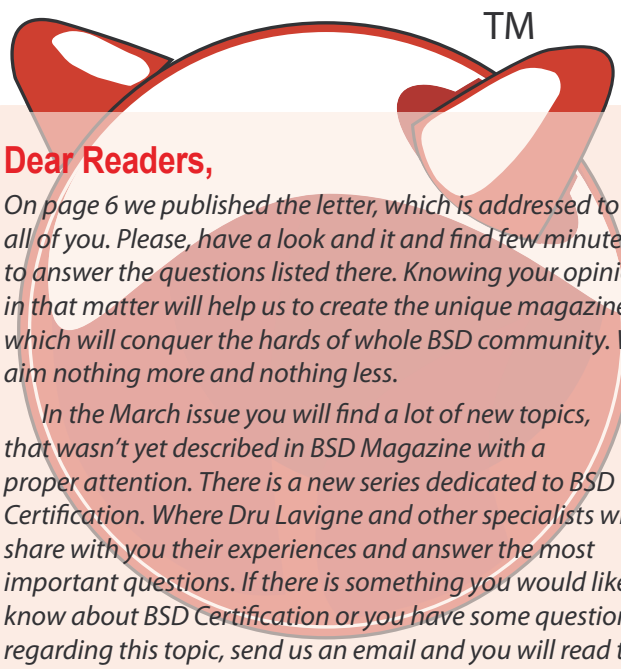
JBOD expansion is available on the 2U and 4U Systems

** 2.5" drive options available; please consult with your Account Manager*

Call iXsystems toll free or visit our website today!

1-855-GREP-4-IX | www.iXsystems.com

Powerful. Intelligent.



Dear Readers,

On page 6 we published the letter, which is addressed to all of you. Please, have a look and it and find few minutes to answer the questions listed there. Knowing your opinion in that matter will help us to create the unique magazine, which will conquer the hards of whole BSD community. We aim nothing more and nothing less.

In the March issue you will find a lot of new topics, that wasn't yet described in BSD Magazine with a proper attention. There is a new series dedicated to BSD Certification. Where Dru Lavigne and other specialists will share with you their experiences and answer the most important questions. If there is something you would like to know about BSD Certification or you have some questions regarding this topic, send us an email and you will read the answers in the next issue. This is the opportunity for all, who consider taking these exams, to find out more about it and prepare better to pass it.

In Developer's Corner apart from Kris Moore article: Customizing Your PC-BSD 9.0 Desktop we welcome a new contributor – Lucas Holt with his article: mport: The MidnightBSD Package Management Tools. We hope you will like it and will be eager to find out more about MidnightBSD, cause Lucas promised us to write soon one more article...

In How To awaits you another surprise – PostgreSQL: From Installation to PITR by Luca Ferrari, who is a truly enthusiast of PostgreSQL. Will he manage to affect you with his passion to this database? The time will show, cause it's just the first article of his series about PostgreSQL in BSD Magazine.

FreeBSD users won't get bored as well. This time Rob Somerville in his series on security for admins prepared for you some exercisers. This part is definitely more practical comparing to the previous ones, so I recommend you to read it and practice a bit. Admins should also read the article: Data Classification Policy by Toby Richards, who claimed the he run out of ideas for articles, but it seems he was only teasing with us:)

The issue ends with Counting Our Losses, where Sander Reiche listed the BSD world heros, who passed in 2011.

Wish You enjoy reading! And don't forget to send us your feedback!

Patrycja Przybyłowicz
& BSD Team

MAGAZINE BSD

Editor in Chief:
Patrycja Przybyłowicz
patrycja.przybylowicz@software.com.pl

Contributing:
Dru Lavigne, Toby Richards, Rob Sommerville, Luca Ferrari, Kris Moore, Lucas Holt, Sander Reiche, Guillaume Duale, Richard Batka

Proofreaders:
Paul McMath, Zander Hill, Sander Reiche, Bjorn Michelsen

Special Thanks:
Denise Ebery
Dru Lavigne

Art Director:
Ireneusz Pogroszewski

DTP:
Ireneusz Pogroszewski

Senior Consultant/Publisher:
Paweł Marciniak *pawel@software.com.pl*

CEO:
Ewa Dudzic
ewa.dudzic@software.com.pl

Production Director:
Andrzej Kuca
andrzej.kuca@software.com.pl

Executive Ad Consultant:
Ewa Dudzic
ewa.dudzic@software.com.pl

Advertising Sales:
Patrycja Przybyłowicz
patrycja.przybylowicz@software.com.pl

Publisher :
Software Press Sp. z o.o. SK
ul. Bokserska 1, 02-682 Warszawa
Poland
worldwide publishing
tel: 1 917 338 36 31
www.bsdmag.org

Software Press Sp z o.o. SK is looking for partners from all over the world. If you are interested in cooperation with us, please contact us via e-mail: *editors@bsdmag.org*

Developers Corner

BSD Certification

How To

Admin

Tips & Tricks

Security

Let's Talk

Questions for BSD Magazine Readers

*This text was inspired by one of BSD Magazine's regular authors, who pointed out few important issues regarding the topics witch the BSD Magazine covers. It concerns the problem whether the BSD Magazine should publish articles that are not entirely "BSD specific", but can by applied as well on Linux or any other *nix. A good example of such article is installing and configuring Nagios, which is not specific to BSD and can be learned by reading the Linux tutorial. Here are some more examples of such articles listed by this author:*

Openfire: A Robust Jabber Server [on BSD]; Shell Scripting for Beginners [on BSD]; Poking at Remote Hosts with Nmap [on BSD]; Reverse Proxies with Nginx [on BSD]; Vulnerability Scanning with Nessus [on BSD]; Sudo vs Su [on BSD]: Why Stallman is Wrong (an editorial); Building a Private WoW Server [on BSD]; Building Your Wiki [on BSD]; Making Life Easier with Webmin [on BSD]; Filtering the Web with Dansguardian [on BSD]; etc...

Another problematic topic could be an article about NAT/PAT/Port Forwarding, which you'd think would be BSD specific because the BSD family have their own firewalls. The problem is that such an article is going to have far more information about routing concepts than about using those concepts in BSD.

So dear readers, here are four questions for you:

1. *Do you want to see articles in BSD Magazine such as the ones above that wouldn't be necessarily "BSD specific", but apply to Linux as well as BSD?*
2. *What ideas do you have about articles that are BSD specific?*
3. *Are you interested in OS X articles? If so, then about what topics?*
4. *Are you more interested in beginner/intermediate articles, or articles intended for BSD experts?*

Please send responses to editors@bsdmag.org with the "BSDfeedback" in the subject.

BSD Magazine is based on the open source community. It exists thanks to your contributions and interest. Your opinion is very valuable for us and helps us to improve the quality of content. Help us to create the magazine you want to read.

Great Specials

On FreeBSD & PC-BSD Merchandise

Give us a call & ask about our
SOFTWARE BUNDLES
1.925.240.6652

$39.95
FreeBSD 9.0 Jewel Case CD Set
or FreeBSD 9.0 DVD

$29.95
PC-BSD 9.0 DVD

$49.95 Save a BUNDLE!
The PC-BSD 9.0 Users Handbook
PC-BSD 9.0 DVD

$99.95
The FreeBSD CD or DVD Bundle

Inside each CD/DVD Bundle, you'll find:
FreeBSD Handbook, 3rd Edition
Users Guide FreeBSD Handbook, 3rd Edition, Admin Guide
FreeBSD 9.0 CD or DVD set
FreeBSD Toolkit DVD

Stylish Dress Attire
Look Your Professional Best

Comfy Hoodies
Stay Warm in Pullovers & Zip Ups

T-Shirts
Lots of Styles to Choose From

FreeBSD 9.0 Jewel Case CD/DVD $39.95

CD Set Contains:

- **Disc 1:** Installation Boot LiveCD (i386)
- **Disc 2:** Essential Packages Xorg, GNOME2 (i386)
- **Disc 3:** Installation Boot LiveCD (amd64)
- **Disc 4:** Essential Packages Xorg, GNOME2 (amd64)

FreeBSD 8.2 CD $39.95
FreeBSD 8.2 DVD $39.95

FreeBSD Subscriptions

Save time and $$$ by subscribing to regular updates of FreeBSD

FreeBSD Subscription, start with CD 9.0 ..... $29.95
FreeBSD Subscription, start with DVD 9.0 ..... $29.95
FreeBSD Subscription, start with CD 8.2 ..... $29.95
FreeBSD Subscription, start with DVD 8.2 ..... $29.95

PC-BSD 9.0 DVD (Isotope Edition)

PC-BSD 9.0 DVD $29.95
PC-BSD Subscription $19.95

The FreeBSD Handbook

The FreeBSD Handbook, Volume 1 (User Guide) $39.95
The FreeBSD Handbook, Volume 2 (Admin Guide) $39.95

The FreeBSD Handbook Specials

The FreeBSD Handbook, Volume 2 (Both Volumes) $59.95
The FreeBSD Handbook, Both Volumes & FreeBSD 9.0 ..... $79.95

PC-BSD 9.0 Users Handbook $24.95

BSD Magazine $11.99

The FreeBSD Toolkit DVD $39.95

FreeBSD Mousepad $10.00

FreeBSD & PCBSD Caps $20.00

BSD Daemon Horns $2.00

Bundle Specials!
Save $$$

Just Plain Fun
Mousepads & Novelty Horns

BSD Magazine
Available Monthly

For even MORE items
visit our website today!
www.FreeBSDMall.com

Customizing Your PC-BSD 9.0 Desktop

One of the most important new features of PC-BSD 9.0 is the ability to customize your desktop with a variety of different FreeBSD packages, such as desktops, window-managers, servers, drivers and more.

This is done through the new *pc-metapkgmanager* utility and it's front-ends, which allows users to quickly and easily select a bundle of packages, such as KDE or GNOME, to install. This ability in effect allows a user to deploy a custom FreeBSD based desktop without the hassle of using the command-line and manually resolving dependency issues. Lets take a look at how this can be used in the new PC-BSD 9 desktop. When installing PC-BSD for the first time from a complete media, such as DVD, Network, or USB-Full image, users will be presented with a screen allowing the selection of various meta-packages for the installed system.

Figure 1. *Meta-Package selection during installation*

By checking the desired packages, the PC-BSD installer will customize the system installation to the users liking. More experience users who are curious about what set of FreeBSD packages are going to be installed may find them by right-clicking on a meta-package set and choosing *View Packages*. Below is a complete list of the meta-packages available in 9.0:

Desktops:
- GNOME
  - Accessibility
  - Games
  - Net
  - Utilities
- KDE
  - Accessibility
  - Artwork
  - Education
  - Games
  - Graphics
  - KOffice
  - L10N (Translations)
  - Multimedia
  - Network
  - PIM

Drivers:
- HPLIP
- Handheld
- NVIDIA
- VMwareGuest
- VirtualBoxGuest

Services:
- Database Servers
- Samba
- Web Servers

Misc:
- I18N (Translations)
- Compiz
- MythTV
- XBMC

Development:
- Debug Tools

- SDK
- Toys
- WebDevKit
- LXDE
- XFCE
  - Plugins
- Awesome
- FVWM
- IceWM
- OpenBox
- ScrotWM
- Window Maker
- QT
- Version Control Systems

After the PC-BSD system is installed it is still possible to customize the desktop by adding / removing meta-packages to the users liking. There are a couple possible ways to do this, starting with the easiest via the PC-BSD System Manager. To begin, open the PC-BSD Control Panel: Figure 2. Next Open the *System Manager*: Figure 3. Once the system manager is open, click the *System Packages* tab: Figure 4.

Once at the *System Packages* tab, it is possible to simply check or un-check the respective meta-packages that you wish to add or remove. When installing new packages, the system manager will automatically download them from the currently selected mirror server, which can be changed via the *Mirrors* tab in the system manager.

Should the user accidentally remove or otherwise break their graphical desktop, is it possible to fix the system meta-packages via the command-line using the `pc-metapkgmanager` command. To begin using this command it is helpful to get a list of which meta-packages are available with:

Figure 2. *PC-BSD Control Panel*

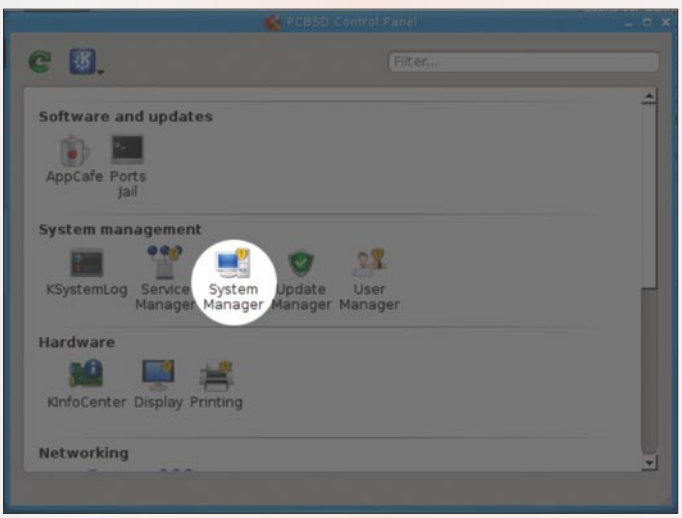


Figure 3. *System Manager*

More Information on the Web

PC-BSD Users Handbook: *http://wiki.pcbsd.org/index.php/PC-BSD_9_Handbook*

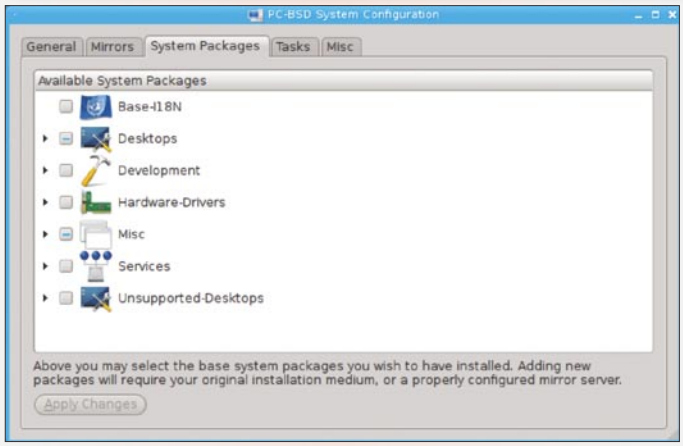


Figure 4. *"System Packages" tab*

```
# pc-metapkgmanager list | more
```

Once a you have determined which meta-package you wish to install or remove, it is possible to do so with the following commands:

```
# pc-metapkgmanager add KDE ftp://mirrors.isc.org/pub/
                    pcbsd/9.0/amd64/netinstall/packages/
# pc-metapkgmanager del KDE
```

The mirror in the example above can be changed to your preferred server URL, taking care to select the correct PC-BSD version / architecture being used. (I.E: 9.0/i386 or 9.0/amd64) After changing the meta-packages, is is possible to confirm the status of a particular one via the *status* flag:

```
# pc-metapkgmanager status KDE
```

The meta-pkg KDE is installed

By using these new tools and utilities it becomes easier than ever to customize your new PC-BSD desktop to suit your unique computing needs. In addition this provides the ability to upgrade your system via the internet to newer versions of PC-BSD, such as 9.1, while selectively preserving and upgrading your selected meta-packages.

KRIS MOORE

Kris Moore is the founder and lead developer of PC-BSD. He lives with his wife and four children in East Tennessee (USA), and enjoys building custom PC's and gaming in his (limited) spare time. kris@pcbsd.org

mport:

The MidnightBSD Package Management Tools

One of the most tedious tasks in setting up and maintaining a personal computer is installing software applications. The BSD community has historically handled this by providing users a ports system to compile software and later some package management software.

What you will learn…

- the history of the mport tool and why it was written,
- basic usage of the mport tool to install and uninstall software,
- hints for power users to script or automate tasks.

What you should know…

- how to install MidnightBSD or download a virtual machine image,
- how to run commands as root using sudo(8) or su(1),
- power users will need familiarity with a scripting language and SQL.

This approach has worked well for power users. The Linux community has built some user friendly package management tools with command line and graphical user interfaces. In minutes, one can install a new web browser or word processor.

When I started the MidnightBSD project, the goal was to bring the benefits of BSD to novice users. Three problems were identified: installing MidnightBSD, managing software, and setting up a graphical user interface. It was clear that package management was essential to solving these problems.

Chris Rienhardt and I started working on these problems in 2007. We planned a new system which required changes to mports, our ports collection, and new tools to manage packages. Chris made the necessary changes to the mports system and started working on a new library to create and manage packages. The system uses SQLite3 to store information about available packages, currently installed software and a list of installed files. By the MidnightBSD 0.3-RELEASE, we had working programs to install and uninstall software from the ports system. The system still lacked a front-end management tool. I decided to develop a tool similar to apt-get, but built around libarchive, libmport, and SQLite3. This new tool will be available in the next release, 0.4 for end users and is available for testing in the development version of MidnightBSD, 0.4-CURRENT.

Using mport

Every task to be performed starts with mport followed by a command verb and optionally a software package name.

Installing Software

Searching for software, will check package titles and descriptions for matching phrases: `mport search gzip`.

Learn about the software package, including if it's currently installed, the version and the license: `mport info gzip`.

If you know the name of the software package to install: `mport install gzip`. Dependencies required by the package are installed automatically.

Table 1. *mport commands*

Command	Description
clean	Clean up old packages and database
list	List installed software
info	Print information about a software package
delete	Delete a software package
deleteall	Delete all software packages currently installed
download	Download a package, but do not install it
install	Install software
search	Search for software
update	Update a single program
upgrade	Update all software installed on the system

Removing Software

To uninstall a package, use `mport delete gzip`; note that this does not delete the package file from the system, it only uninstalls the software.

Listing Installed Software

List all software currently installed on the system: `mport list`. List installed software that is out of date: `mport list updates`.

Update Software

Update all software on the system: `mport upgrade`. Update a specific package, will download the package if necessary: `mport update gzip`.

Maintenance Tasks

Clean up old packages and compress the database with: `mport clean`. Packages will be removed from `/usr/mports/Packages`. Consult the mport (1) manual for more information.

Power Users and Developers

In addition to the mport command, power users can write custom scripts or programs to manipulate the package database. Files used by mport are stored in `/var/db/mport`. The *master.db* file contains all of the installed package data while *index.db* contains the list of available packages from the MidnightBSD package build cluster, magus. Users can open index files with the `sqlite3` command line utility and run SQL queries. Installation logs are available in the logs table, and settings for mirrors to download packages from are stored in the settings table. A program to make it easy to run queries is available at `/usr/libexec/mport.query`.

C, C++ and Objective-C programmers can use libmport to write their own package tools. The source code is available in `src/lib/libmport` with public functions prefixed with `MPORT_PUBLIC_API`.

Listing 1. *Example output from mport info command*

```
mport info gzip
gzip
latest: 1.3.13
installed: 1.3.13
license: gpl3
origin: archivers/gzip
A compression utility designed to be a replacement for
                    compress
```

On the Web

- *http://www.midnightbsd.org/* – MidnightBSD Project website,
- *http://www.midnightbsd.org/documentation/mports/* – mports documentation.

Glossary

- mport
- sqlite

Other Programs

The mports collection makes use of several simple applications written for a specific task. For example, when a package is created using the package target in mports, the program `/usr/libexec/mport.create` is run to make the package followed by `/usr/libexec/mport.install` to install the package with the install target. Another useful program is `/usr/libexec/mport.init` as it creates a fresh *master.db* in the event that it's been corrupted or deleted by mistake.

Future Directions

Following the 0.4-RELEASE of MidnightBSD, I plan to write a graphical application to manage package installations. Further improvements to the system may include using xz for index downloads, improving documentation, fastest mirror selection and refining the upgrade logic. While mport and libmport were designed for the MidnightBSD mports collection, it could be used by other BSD projects.

Summary

mport is the package management solution for MidnightBSD that allows a user to install, update, and delete software packages. It is easy to use, but provides many extension mechanisms for power users to script or automate installations.

LUCAS HOLT

Lucas Holt is the founder of the MidnightBSD project and Programmer/Analyst for Mathematical Reviews in Ann Arbor, MI, USA.

Why Should I Become BSDA Certified?

If you are reading this magazine, you are interested in learning more about BSD systems. Perhaps you have seen this magazine's ads for BSD Certification and want to learn more about this certification program or perhaps you think that certification is not for you. This article addresses some common misconceptions about certification and describes why you should be BSDA certified.

Before starting the BSD Certification Group in 2005, I already had nearly a decade's worth of experience in both taking and teaching IT certifications. This experience gave me the opportunity to learn how various vendors handle their certification programs and how their training methods differ from academic programs. It also gave insight into the concerns raised by potential certificants. These concerns haven't changed over the years and generally fall into the following statements:

- Certifications are a waste of time and not worth the paper they are written on.
- There aren't any training materials available or the training materials are too expensive.
- I'm already working so I don't need to be certified.

This article is the first in a two part series. The first article addresses the above concerns, with a focus on the BSDA certification program, and concludes with some additional resources. The second article will go into more detail on how to become BSDA certified.

Certifications are a Waste of Time and not Worth the Paper they are Written on

Unfortunately, this is true for some certification programs. Exam cram books that allow you to pass an exam without ever having to actually use the software being tested, websites and exam prep software that contain most of the exam's questions and answers, terribly written exam questions that have nothing to do with the exam's objectives or the reality of using the software – all of these add up to give certifications a bad name.

Fortunately, this is not true for most certification programs. And the reason why can be summed up in one word: psychometrics. In a nutshell, psychometrics is the science of assessment. It is a requirement of any accredited academic program (think university or college diploma) and the hallmark of a good certification program, such as the BSDA.

You can tell that a certification program is psychometrically valid if it provides the following benefits:

- *exam objectives are the result of a Job Task Analysis (JTA)*: the first step in a psychometrically valid program is to use a JTA to determine what the certification will assess. As the name implies, a JTA is task-oriented. A good JTA will receive input from those already working in the field being assessed (for example, the BSDA assesses BSD system administration) as well as their employers. This ensures that the resulting exam objectives are based on the real world skills required in that field of employment.
- *the exam questions must match the published exam objectives*: psychometrics requires that the exam objectives are part of the *blueprint* used to create the exam questions. This means that the exam can not contain any content that is not specified in the objectives. The objectives themselves must be very specific as to what the testing candidate must know in order to pass the exam and become certified, making the objectives the definitive study reference. If you understand the content of the objectives, you are ready to pass the exam.
- *the exam content must match the published domain percentages*: the exam's blueprint must also contain the percentages for each domain topic or category. For example, if the blueprint states that Security is worth

20% of the exam and Networking is worth 10%, the exam can not contain more or less than 20% worth of security questions and 10% worth of networking questions. If there are multiple versions of the same exam, one version can not contain more or less security questions than the other versions of the exam.

- *exam questions must be clear*: the intent of a psychometrically valid exam is not to try to trick you, but to determine if you understand the material being tested. Additionally, if an English version of the examination is available in areas where English is not the native language, great care must be taken to ensure that the questions do not contain grammar or colloquialisms that are confusing to persons whose first language is not English. This means that you would not see the following types of questions on a psychometrically valid exam:
  - a 3 page long question where you have to hunt for the question being asked as it is buried somewhere within all that extraneous information
  - a question that contains double negatives or other confusing grammar
  - correct or incorrect answers that are contextually obvious to someone who doesn't understand the material being tested
- *exam questions must match the level of the intended audience*: the exam's blueprint must clearly state the intended audience of the exam and exam questions should not be too easy or too hard for the intended audience to answer. For example, the intended audience for the BSDA exam is described as follows: *The BSDA certification is designed to be an entry-level certification on BSD Unix systems administration. Testing candidates with a general Unix background and at least six months of work experience as a BSD systems administrator, or who wish to obtain employment as a BSD systems administrator, will benefit most from this certification. Human resource departments should consider the successful BSDA certified applicant to be knowledgeable in the daily maintenance of existing BSD systems under the direction and supervision of a more senior administrator.*

The Additional Resources section at the end of this article contains the URL to the blueprint for the BSDA examination as well as links to more information about psychometrics.

There aren't any Training Materials Available or the Training Materials are too Expensive

Sadly, the perceived need for *authorized* training materials and expensive class instruction is a by-product of vendor certifications. Creating and maintaining a certification program is expensive and training materials and official courseware are often used to create an additional revenue stream.

Think of it this way: if someone is studying for a system administration certification, there is no training material or week of classroom instruction that can turn a novice into a system administrator. Training materials can tell you what will be on the exam, but this is redundant as the exam objectives already tell you this for free. And if the training materials tell you which questions and answers are on the exam, we're back in the territory of *certifications are a waste of time and not worth the paper they are written on*.

If a certification is psychometrically valid, it is assessing real world skills. And the only way to obtain real world skills is to sit down with the software being tested and to work your way through the exam's objectives. Some skills you will already have. Some you will need to learn.

This does not mean that you need to learn these skills in a vacuum or that training materials or classroom instruction are a bad thing. It means that you need to sit down and figure out which objectives you need to learn and which resources are available to help you learn the skills associated with those objectives.

Some people just need a man page, a command prompt, and a bit of time to practice in order to figure out a new skill. To assist these types of learners, each BSDA exam objective contains the man pages associated with that skill and a downloadable Command Reference mapping the man pages to each BSD operating system.

Some times the man pages are not enough and you need another person to demonstrate how to use a command or to explain that bit of knowledge that is alluding you. Often that person will be a co-worker or a knowledgeable friend. If these are in short supply, a question on an IRC channel or mailing list might do the trick. And, if there is a sysadmin, UNIX, BSD, or Linux user group in your area, you're quite likely to find someone who can assist you.

If you are looking for reading materials, don't limit yourself to *official* courseware. Skim through system administration books to see if they provide walkthroughs for the skills that you need help with. A good system administration book makes a valuable reference on any administrator's physical or digital bookshelf.

If you are looking for a training course, ask for the trainer's credentials as well as how many hours of supervised, hands-on lab work is provided during the course. If the course offers little lab time, save your money and buy a good book. A good trainer is worth their weight in gold as they can answer your questions, show you how to do something, and offer tips on how to do things better. A good trainer will have real world, hands-on experience in the field that they are teaching. If you do sign up for a

Additional Resources

- BSDA Certification Website: *http://www.bsdcertification.org/certification/associate.html*
- BSDA blueprint (Certification Requirements Document): *http://www.bsdcertification.org/downloads/pr_20051005_certreq_bsda_en_en.pdf*
- Psychometrics and Exam Construction: *http://www.bsdcertification.org/downloads/psychometric.pdf*
- How to Create a Psychometrically Valid Certification Examination: *http://www.slideshare.net/dlavigne/eurobsdcon-2011*
- Playing the Certification Game: *http://www.slideshare.net/dlavigne/lisa2011*
- Why Certification Exams Suck (series of 4): *http://it.toolbox.com/blogs/bsd-guru/?page=30*

course, write down the questions that you need to have answered, take them with you to the course, and make sure that they all get answered by the end of the course.

I'm Already Working so I don't Need to be Certified

It is a common misconception that certifications only provide value to those trying to break into a new field of employment or those who are currently looking for work. If you are already working, perhaps you are in a Windows or Linux environment and administering BSD systems is not part of your daily duties. This does not mean that you can't benefit from being BSDA certified. Even if you are already working as a system administrator, the BSDA certification provides the following benefits:

- *it allows you to fill in knowledge gaps*: many system administration skills are learned on the job on an as-needed basis, are repetitive, and are limited to a particular operating system version. This means that you can get very good at the specific tasks required by your particular environment, but may not have the opportunity to learn a broader range of skills. Studying for a certification exam forces you to learn new skills and to determine which skills are considered important within an industry. Filling in these knowledge gaps allows you to widen your current skillset, and these new skills can be an asset in both your current work environment and future employment opportunities. Doing this while employed allows you to proactively increase your skillset without the additional stress of being unemployed and needing to find a job.
- *it increases your value to your employer*: certifications can provide a competitive advantage to an employer. It is reassuring to customers and partners to know that a company has certified professionals available to meet their needs. If your manager is smart, he'll notice that you take that extra initiative to obtain certifications relevant to your industry and this could make a difference when it comes time for promotions or layoffs.
- *it increases technology adoption*: if you are already good at BSD system administration, you're probably interested in seeing it more widely used within the industry. Even if you already know all of the skills covered by the exam objectives and can easily pass the exam, becoming BSDA certified shows your support for the field of BSD system administration. It demonstrates to your employer that BSD is relevant and that other certified professionals are available should you become promoted or additional BSD systems are added to the company's infrastructure.

Conclusion

This article outlined some of the benefits provided by a psychometrically valid certification program as well as some tips for learning the skills needed to pass a certification exam. Hopefully, it has piqued your interest in becoming BSDA certified.

The next article will concentrate on the BSDA certification program: where to take the exam, how much it costs, what to do if a testing event is not available in your area, and how to become involved with the BSD certification community.

DRU LAVIGNE

Dru Lavigne is author of BSD Hacks, The Best of FreeBSD Basics, and The Definitive Guide to PC-BSD. As Director of Community Development for the PC-BSD Project, she leads the documentation team, assists new users, helps to find and fix bugs, and reaches out to the community to discover their needs. She is the former Managing Editor of the Open Source Business Resource, a free monthly publication covering open source and the commercialization of open source assets. She is founder and current Chair of the BSD Certification Group Inc., a non-profit organization with a mission to create the standard for certifying BSD system administrators, and serves on the Board of the FreeBSD Foundation.

MidnightBSD

the BSD for everyone

Come join MidnightBSD:

- desktop BSD
- over 2,000 ports
- unique mports package installer
- helpful community
- hands-on learning for new developers

www.midnightbsd.org

What Can't You Do

On The command-line?

Most of us have all grown accustomed to some form of true graphical interface to our computers. But there's always that group of so-called 'geeks', which remain to work at a simple 80x24 console. But that's just for geeks, or is it?

What you will learn…

- Writing
- While listening to music
- Publishing it
- Be social

What you should know…

- How to open a console/terminal
- Your way around the shell

First things first. There are downsides. Text-based browsing used to be a lot better, back in the days. With all the crazy techniques used on web pages out there these days and the lack of proper web design with graceful degradation back to pure and simple text, text-based browsing can mostly be done on older or simpler websites. But any day is a new day out there on the big Wide Web. Maybe your favourite site some day, out of the blue, supports text-based browsing a bit better than before. Never stop trying!

And because the WWW today is more media than old-skool plain HTML, that brings us to media in general on the command-line. Except for music, the rest isn't really an option on the 80x24 terminal. If you do have X11 installed on your system, then a complete whole new world of possibilities open up, but I might get into that in another article.

Writing

Using something which usually is installed on most of the *NIX systems, is a version or form of troff. Currently most will be distributing GNU's roff, groff(1). Most people will know it from the man-pages, but it's so much more than that. You can write articles, papers and even whole books using troff. There will be a pretty steep learning curve and practice make perfect, but the results are simply awesome. Consider the following example: Listing 1.

Run this through `groff(1)` which outputs PostScript and note, `ps2pdf(1)` comes from ghostscript:

```
$ groff -ms example.ms | ps2pdf - example.pdf
```

And view it in your favourite PDF reader, i.e.:

```
$ xpdf example.pdf
```

And yes, you need a graphical interface for that. You could, to escape the graphical interface, output the PostScript towards a PostScript-capable printer of course, but paper is expensive so viewing your draft PDFs on a X11 desktop might be better for your wallet.

If that didn't already blow your mind, try the following example: Listing 2.

Render the result using the following:

```
$ pic pictest | groff | ps2pdf - 1.pdf
```

View the awesomeness:

```
$ xpdf 1.pdf
```

So that's troff in a very, very small nutshell. Just consider the possibilities right there at your fingertips at this moment! For a far better read on the history of troff and

Listing 1. *example.ms*

```
.\" Example of -ms macro runoff. This is a comment
.RP no
.P1
.ds LH Left header
.ds CH Center header
.ds RF Right footer
.TL
This is a title
.AU
S. Reiche
.AI
ls-al.eu - public \s-2UNIX\[rg]\s0 Access
.AB no
.AE
.LP
We begin a nice paragraph all the way to the left.
.PP
Then we have a indented starting paragraph.
.XP
An exdented (yes) paragraph.
.QP
And of course, we couldn't live without a quoting
                    paragraph.
.NH 1
Heading
.NH 2
Subheading
.SH
Unnumbered subheading
.PP
And of course,
.B
we have
.I
some formatting
.R
.BI capabilities.
Even
.BX boxes
or simple
.UL underlinings.
.LG
LARGE
.NL
or
.SM
small
.NL
fontsizes and everything.
.IP 1. 12
lists are always handy

even unnumbered ones or
.IP glossary
style.
.LP
But what's a technical document without footnotes\**?
.FS
You know, footnotes!
.FE
```

Listing 2. *pictest*

```
.PS
lineht = lineht / 2

box "\fIletter.tr\fP"
arrow
circle "tbl"
arrow
Eqn: circle "eqn"
arrow
Troff: circle "troff"
arc cw
line down
arc cw
left
arrow
circle "grops"
arrow
box "\fIletter.ps\fP"

up
line <- from Troff.n
arc
line
box "\fItmac.m\fP"
.PE
```

the incredible intricansies on it, get the free Hayden Book on troff at *http://oreilly.com/openbook/utp/*. A group of enthusiasts found Unix Text Processing such a great book, that they've come together and transcribed the whole book so that there's a true troff source available to build the book yourself (*http://home.windstream.net/kollar/utp/*).

XeTeX, LaTeX, TeTeX are all names for what is in the basis; TeX. TeX was created by Don Knuth, who we all love of course. I consider TeX to be troff's big brother, as it costs roughly about anything from half a gigabyte to multiple gigabytes of storage space on your machine, but it does produce extremely beautiful documents. The TeX Showcase (*http://www.tug.org/texshowcase/*) is a wonderful place to see what TeX is capable of and have a look at Knuth's TeXbook (ISBN 0-201-13448-9) for which the source is also available to learn from.

In time you will get crafty at producing PostScript documents, which you'll probably just convert to PDF. But there are times where you can tinker around with the PostScript output itself. Remember, PostScript is a programming language so after producing something with troff or TeX, take a look at the PostScript, tinker around a bit and see what it does directly with `gs(1)` or convert the modified PS to PDF and then check it out.

Music

But wait! You'll need some background music to do your writing with, right? I'm very fond of `mpg123(1)` because of it's small size and very simplistic interface. But take a look at the others out there. For example in the OpenBSD ports tree; mpg321, mp3blaster, herrie (very nice one!) or shell-fm for last.fm support. And there's lots more out there.

Downloading

And you need something to write a funny article about. So you can start up the download for the newest Ubuntu with the well-known `wget(1)` or `curl(1)` but you could also be more *community-friendly* and use the wonderful console-based rtorrent to pull the latest iso image using the bittorrent protocol and share back to your Linux friends of course.

Weblog

Then you're finished with your article in PDF form and you want to publish it. You could go hardcore with `vi` in-hand and write the site yourself, but there's also something called blazeblogger (*http://blaze.blackened.cz*) which is a set of perl scripts which work out a wonderful CMS without the need for a whole database backend and/or PHP. Very simple interface though nicely intuitive for the console user. The end results are nice when using the defaults, but everything is customizable.

You want to keep your friends on twitter in the loop as well, but `lynx(1)` or `links(1)` doesn't do a very good job at opening that site in all its interactive form. In comes ttyter (*http://www.floodgap.com/software/ttytter/*), a console-based twitter client, based on perl.

Chat

Even more of your friends are reknowned idlers on IRC, so you fire up `irssi(1)` and connect to the IRC server to blab about your latest article. A plugin for irssi (*http://cybione.org/~irssi-xmpp/*) can connect you to XMPP-style chatservices, like GoogleTalk. If you use bitlbee (*http://www.bitlbee.org/main.php/news.r.html*) with that, you can open up even more chat protocols at the same time.

A stand-alone jabber/xmpp console client is GNU's Freetalk which is in the ports tree. If you're more of a true MSN-type, then tmsnc is a nice console-based MSN-only client (*http://tmsnc.sourceforge.net/*).

Mail

After a couple of days, you want to check if anyone reacted to your article by emailing you. If you're running your own or utilizing your ISP's mailserver, you can probably just use the normal ways of fetchmail to get to your mail. GMail usually presents somewhat of a challenge. Here's a couple of links to get the mail client of your choice working for GMail:

- POP/SSL GMail with fetchmail *http://www.axllent.org/docs/networking/gmail_pop3_with_fetchmail*
- IMAP/SSL GMail with fetchmail *http://www.daemonforums.org/showthread.php?t=5590*
- IMAP/SSL GMail with mutt *http://shreevatsa.wordpress.com/2007/07/31/using-gmail-with-mutt-the-minimal-way/*
- SMTP/SSL GMail with heirloom-mailx(nail) and msmtp *http://ubuntuforums.org/showthread.php?t=780509*

And don't forget Michael Hernandez' awesome articles on *mutt on OS X* in BSDMag!

So there you have it. A small introduction on what effectively still can be done, purely on a 80x24 terminal interface. I think it's awesome and I rarely use X11 or something else window-y for my work and I think you could do your best to try the same as well!

SANDER REICHE

Sander Reiche is a PDP-11 fanatic and BSD/UNIX lover in his spare time, and a UNIX Systems Engineer on his day-job. Founder of the Veritable UNIX Systems Group. His web page is located at http://ls-al.eu/~reiche.

PostgreSQL

From Installation to PITR

"The most advanced open-source database available anywhere", this is the comment for the PostgreSQL package.

What you will learn…

- basic concepts about PostgreSQL
- how to install and start a PostgreSQL instance
- how to create users and database and how to backup your data

What you should know…

- basic SQL concepts
- basic shell commands and how to use the ports tree

PostgreSQL is an enterprise-level ORDBMS very stable and reliable, with a rich set of features that make it competing with well known commercial databases. PostgreSQL is released under a BSD license, has a very comprehensive documentation and is maintained by a set of database experts around the planet. In this article you will learn how to install and set up a PostgreSQL cluster, how to interact with the system and how to make backups of your databases.

Concepts

PostgreSQL manages a *cluster* of databases: a single PostgreSQL instance can run and control a set of databases, all kept isolated, and all served through the same TCP/IP socket address (Unix domain sockets are also allowed). There is no limitation about how many PostgreSQL instances you can run, except of course for host system resource limits and TCP/IP sockets.

The cluster is based on a *process* schema: the service starts a main process (historically called *postmaster*) which waits for incoming client connections. Once a connection is going to be accepted, the postmaster forks itself and the new process begins serving the client; such process is named *backend*. The choice of using a dedicated process to serve each client connection is due to the security and portability that the process API offers with regard to the thread one.

The cluster exploits the file system to keep all the databases and their data on mass storage: there is a main directory (per-cluster) called PGDATA which is organized into sub-directories, one per database, which in turn contain all the database objects (tables, sequences, and so on). It is also possible to *escape* from PGDATA keeping some stuff in another directory tree; this is done using the *tablespace* feature. PostgreSQL configuration is mainly done via a few text files that are usually contained withing PGDATA.

From the above readers can see that in order to run different clusters on the same machine there must be for each instance:

- a different TCP/IP socket on which the postmaster can wait for client connections;
- a different PGDATA directory tree;
- a different set of configuration files (usually kept under PGDATA).

Basic Installation

It is possibleto install PostgreSQL on a FreeBSD machine using the ports tree: it is required to install the *-server* port that will install also the *-client* port used to access the cluster (i.e., connecting to backend processes). Optionally it is worth installing also the *-contrib* module that provides useful tools to better manage the PostgreSQL instance.

Before begin installation you have to choose the right version. PostgreSQL version number is in the form `mainstream.number.minor` (e.g., 9.1.2).

The combination of `mainstream.number` makes the major release number and versions with different major numbers usually requires a full database dump and re-initialization. Versions with different minor numbers do not break compatibility and therefore do not require a re-initialization.

Listing 1 shows the steps required to install a 9.1 database server (and client tools); as readers can see at the end of the installation both the client and server packages will be installed, as well as a *pgsql* user will be added to the system. Such user is used to run the server processes in an unprivileged mode.

Once the database has been installed a new cluster must be created. In order to do that a new directory must be assigned as PostgreSQL PGDATA; creating

Listing 1. *Installing PostgreSQL from ports*

```
# cd /usr/ports/databases/postgresql91-server/
# make install clean
# cd /usr/ports/databases/postgresql91-contrib/
# make install clean
...
# id pgsql
uid=70(pgsql) gid=70(pgsql) groups=70(pgsql)
# pkg_info -cs 'postgresql*'
Information for postgresql-client-9.1.2:
Comment:
PostgreSQL database (client)
Package Size:
7914    (1K-blocks)
Information for postgresql-contrib-9.1.2:
Comment:
The contrib utilities from the PostgreSQL distribution
Package Size:
1647    (1K-blocks)
Information for postgresql-server-9.1.2:
Comment:
The most advanced open-source database available anywhere
Package Size:
14621   (1K-blocks)
```

Listing 2. *Initializing a cluster without PGDATA*

```
# echo 'postgresql_enable="YES"' >> /etc/rc.conf
# echo 'postgresql_data="/postgresql/cluster1"' >> /etc/rc.conf
# mkdir -p /postgresql/cluster1
# chown pgsql:pgsql /postgresql/cluster1/
# /usr/local/etc/rc.d/postgresql initdb
```

Listing 3. *Initializing a cluster with PGDATA*

```
# mkdir /postgresql/anotherCluster
# chown pgsql:pgsql /postgresql/anotherCluster
# su -l pgsql -c "exec /usr/local/bin/initdb -D /postgresql/anotherCluster"
```

a new directory does not suffice, the directory must be initialized with the `initdb(1)` command so that PostgreSQL can store into such directory basic data structures for the cluster to work. All the main cluster actions, including the initialization, start, stop and status of the cluster can be handled via the `/usr/local/etc/rc.d/postgresql` rc script, which in turn calls the correct command (usually `pg_ctl(1)` or `initdb(1)`) through the *pgsql* unprivileged user. As for many other services, it is important to have the service enabled in the `rc.conf` file as well as the main cluster directory so that all the PostgreSQL tools can be executed without the need of specifying the PGDATA directory. Listing 2 shows the set-up of a default PostgreSQL installation using `/etc/rc.conf` variables, while Listing 3 shows how the same result can be achieved specifying the PGDATA directory as command line argument.

With regard to the installation of the Listing 2 it is now time to start the service and to verify that it is running. Please note that no databases neither specific users have been created so far, so you will need to use the *pgsql* user to connect to the database (see Listing 4). The connection to the cluster (and to any of its databases) is performed through the `psql(1)` command, which can act as an interactive shell or can execute SQL statements as batch.

Anatomy of a fresh-installed instance

As shown in Listing 4 the cluster is not empty and two very important databases are ready: *template1* and *template0*.

Listing 4. *Starting PostgreSQL and getting the first information*

```
# service postgresql start
# /usr/local/bin/psql -l -U pgsql
                            List of databases
   Name    | Owner | Encoding | Collate |    Ctype    | Access privileges
-----------+-------+----------+---------+-------------+-------------------
 postgres  | pgsql | UTF8     | C       | en_US.UTF-8 |
 template0 | pgsql | UTF8     | C       | en_US.UTF-8 | =c/pgsql          +
           |       |          |         |             | pgsql=CTc/pgsql
 template1 | pgsql | UTF8     | C       | en_US.UTF-8 | =c/pgsql          +
```

Listing 5. *Inspecting a few PostgreSQL settings*

```
template1=# SELECT name, setting, reset_val, context, sourcefile,sourceline
FROM pg_settings WHERE sourcefile IS NOT NULL;
           name           |      setting       |     reset_val      |  context   |              sourcefile              | sourceline
--------------------------+--------------------+--------------------+------------+--------------------------------------+------------
 DateStyle                | ISO, MDY           | ISO, MDY           | user       | /postgresql/cluster1/postgresql.conf |        485
 default_text_search_config | pg_catalog.english | pg_catalog.english | user       | /postgresql/cluster1/postgresql.conf |        507
 lc_messages              | en_US.UTF-8        | en_US.UTF-8        | superuser  | /postgresql/cluster1/postgresql.conf |        500
 lc_monetary              | en_US.UTF-8        | en_US.UTF-8        | user       | /postgresql/cluster1/postgresql.conf |        502
 lc_numeric               | en_US.UTF-8        | en_US.UTF-8        | user       | /postgresql/cluster1/postgresql.conf |        503
 lc_time                  | en_US.UTF-8        | en_US.UTF-8        | user       | /postgresql/cluster1/postgresql.conf |        504
 log_destination          | syslog             | syslog             | sighup     | /postgresql/cluster1/postgresql.conf |        276
 max_connections          | 40                 | 40                 | postmaster | /postgresql/cluster1/postgresql.conf |         64
 shared_buffers           | 3584               | 3584               | postmaster | /postgresql/cluster1/postgresql.conf |        109
 silent_mode              | on                 | on                 | postmaster | /postgresql/cluster1/postgresql.conf |        313
 update_process_title     | off                | off                | superuser  | /postgresql/cluster1/postgresql.conf |        419
template1=# SHOW shared_buffers;
 shared_buffers
----------------
 28MB
```

Table 1. *Main PGDATA entries and their meaning*

PGDATA entry	Type	Meaning
PG_VERSION	text file	Contains the major number version of the cluster that owns PGDATA.
postgresql.conf	text file	Contains all the main options and configuration settings for the cluster.
pg_hba.conf	text file	Host Based Access (HBA): specifies which hosts can connect to which databases.
base	directory	Contains cluster databases (one per directory).
pg_tblspc	directory	Contains links to external directories used to store database objects (tablespaces).
pg_clog	directory	Contains the transaction commits logs.
pg_xlog	directory	Contains the Write Ahead Logs (WAL) required for proper cluster functioning and survival.
global	directory	Contains the cluster catalog.
pg_ident.conf	text file	Allows a mapping of Operating System usernames to database usernames.

The two template databases act as a skeleton for other database that will be created in the future by the DBA; both the database are created during the initialization of the cluster (i.e., when `initdb(1)` runs). In particular each newly created database will be cloned starting from *template1*; *template0* act as a backup copy of *template1* just in case the latter is compromised. Please note also that both templates can be used as regular databases to which users can connect to.

The PGDATA directory is owned exclusively by the *psql* user and contains configuration files for the whole cluster as well as data files; main entries are detailed in Table 1 and shown in the blue part of Figure 1. Please consider that doing a plain file system level backup of the PGDATA directory is not enough to restore or migrate a cluster; much complex procedures like Point In Time Recovery (more on this later) need to be applied.

Each database object is identified by an unique number called OID (*Object IDentifier*); each object is then stored on the file system (under PGDATA) into a file named after the object OID. Each data file can grow to a maximum of 1 GB, after that the file is split into segments each numbered with the OID and a progressive counter.

Basic configuration

PostgreSQL configuration is based on *key-value* textual parameters. By default the parameters are stored in the `postgresql.conf` file, but can be accessed even through the database catalogues. Each parameter is tight to a specific context that define when the cluster will apply the setting change as follows:

- *postmaster*: the cluster must be restarted;
- *sighup*: the cluster must get a SIGHUP signal;
- *backend*: applied to all new client connections;
- *[super]user*: immediate if done by a [super]user.

Settings can be inspected and changed (unless in the *postmaster* context) from a connection to the database using respectively the commands `SHOW`, `SET` or querying the *pg_settings* catalogue view, which shows also the context to which the parameter is applied and from which

Table 2. *Main postgresql.conf configuration parameters*

Parameter	Meaning
listen_address	The IP addresses on which the daemon accepts client connections.
port	TCP/IP port on which the daemon listen for incoming connections.
max_connections	The max number of client connections that can be created.
shared_buffers	The overall memory used by all the PostgreSQL processes to keep data in memory and client connections (consider at least 400kB per connection).
work_mem	Memory used temporarly for re-ordering of data.
maintanance_work_mem	Memory used by maintanance processes (vacuum, reindexing, etc.).
log_destination log_directory log_filename	Define where daemon logs should be sent and stored (not WAL or commit logs).
log_min_duration_statement	Sets a threshold that will cause the backend process to log a query that is executing for more seconds than the value of the settings (useful to log slow queries).

Listing 6. *Creating a database from the shell*

```
~> createuser -P bsdmag
Enter password for new role:
Enter it again:
Shall the new role be a superuser? (y/n) y
~> createdb -O bsdmag bsdmagdb
```

Listing 7. *Creating a database from the database itself*

```
template1=# CREATE USER bsdmag WITH SUPERUSER LOGIN ENCRYPTED PASSWORD 'bsdmag';
CREATE ROLE
template1=# CREATE DATABASE bsdmagdb WITH OWNER bsdmag;
CREATE DATABASE
```

Listing 8. *HBA rules (pg_hba.conf)*

```
# TYPE  DATABASE        USER            ADDRESS                 METHOD

local   all             pgsql                                   trust
host    bsdmagdb        bsdmag          192.168.200.0/24        md5
```

Listing 9. *Populating the database*

```
~> psql bsdmag -U bsdmag -h 192.168.200.2
bsdmagdb=# CREATE TABLE magazine(
pk serial,
id text,
month int,
issuedon date,
title text,
PRIMARY KEY(pk),
UNIQUE(id)
);
NOTICE:  CREATE TABLE will create implicit sequence "magazine_pk_seq" for serial column "magazine.pk"
NOTICE:  CREATE TABLE / PRIMARY KEY will create implicit index "magazine_pkey" for table "magazine"
NOTICE:  CREATE TABLE / UNIQUE will create implicit index "magazine_id_key" for table "magazine"
CREATE TABLE
bsdmagdb=# INSERT INTO magazine(id, month, issuedon, title)
VALUES('2012-01', 1, '01/01/2012'::date, 'FreeBSD: Get Up To Date');
INSERT 0 1
bsdmagdb=# INSERT INTO magazine(id, month, issuedon, title)
VALUES('2011-12', 12, '01/12/2011'::date, 'Rolling Your Own Kernel');
INSERT 0 1
bsdmagdb=# INSERT INTO magazine(id, month, issuedon, title)
VALUES('2011-11', 11, '01/11/2011'::date, 'Speed Daemons');
INSERT 0 1
```

configuration file it comes from (see Listing 5 for an example).

An example minimal set of configuration parameters that should be review before starting the PostgreSQL instance is listed in Table 2. Beside those parameters the Host Based Access control list (contained into `pg_hba.conf`) should be reviewed for every new created user or database.

Creating a new database

For the example of this article a simple database for your favourite magazine will be created, so the *bsdmagdb* database and the *bsdmag* superuser will be the starting point. PostgreSQL allows you to create (and destroy) databases and users from either the database itself (e.g., when you are connected to the *template1*) or from the shell. Listing 6 shows how to create the superuser and the database from the shell prompt, while Listing 7 how to do the same from a database connection.

Once the database and its user(s) are in place it is worth reviewing the Host Based Access rules contained in the `pg_hba.conf`. Such rules specifies which host can connect to the cluster and which users can connect to which database. Covering the syntax of the `pg_hba.conf` file is out of the scope of this article, it suffice to say that the rules in Listing 8 specifies that connections to database *bsdmagdb* are allowed only for user *bsdmag* from an host within the network 192.168.200.0/24. The *md5* options force the user to be prompted for a password, while the keyword trust allows the user to connect even without providing a password. Therefore while user bsdmag is allowed to connect only to the

Listing 10. *File information*

```
~> oid2name
All databases:
    Oid  Database Name  Tablespace
----------------------------------
  16387       bsdmagdb  pg_default
  11912       postgres  pg_default
  11904      template0  pg_default
      1      template1  pg_default
~> oid2name -H 192.168.200.2 -d bsdmagdb -U bsdmag -t magazine
From database "bsdmagdb":
  Filenode  Table Name
----------------------
     16390    magazine
~> ls -l /postgresql/cluster1/base/16387/16390
-rw-------  1 pgsql  pgsql  8192 Jan 19 14:26 /postgresql/cluster1/base/16387/16390
```

Listing 11. *Increasing the size of a table and seeing the space required on disk*

```
bsdmagdb=# INSERT INTO magazine(id, month,issuedon, title)
bsdmagdb-# VALUES( generate_series(1,100000), 0, '01/01/2009'::date, 'TEST');
INSERT 0 100000
bsdmagdb=# SELECT relname,relfilenode,relpages,reltuples
bsdmagdb-# FROM pg_class WHERE relname = 'magazine';
 relname  | relfilenode | relpages | reltuples
----------+-------------+----------+-----------
 magazine |       16390 |      690 |    100003

~> ls -l /postgresql/cluster1/base/16387/16390
-rw-------  1 pgsql  pgsql  5652480 Jan 19 15:44 /postgresql/cluster1/base/16387/16390
```

Listing 12. *Single database backup on plain text file using pg_dump*

```
~> pg_dump -h 192.168.200.2 -U bsdmag -f bsdmag.backup.sql bsdmagdb
```

bsdmagdb database from the local network and will be prompted for a password, the user *pgsql* can connect to any database without a password prompt through a local domain socket.

It is time now to connect to the above *bsdmagdb* database and populate it with at least one table and a few tuples: Listing 9 shows the creation of a table to store heading information about magazine issues. Please consider that the `psql` command will prompt for the user password upon each new connection, but it is possible to avoid the password request. It does suffice to create the `~/.pgpass` file (user readable) and store in such file an entry in the form

```
server-address:port:dbname:username:password
```

as for instance

```
192.168.200.2:5432:bsdmagdb:bsdmag:mypassword
```

so that before prompting for a password the psql command will search for a match in the `~/.pgpass` file and, if found, will use such password. This can also be very convenient for scripts scheduled via `cron(8)` or `periodic(8)`.

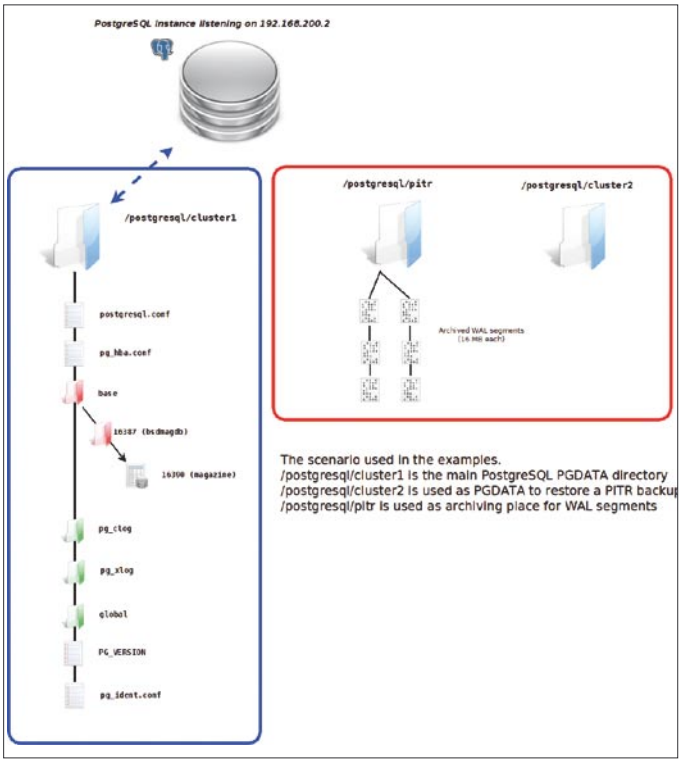


Figure 1. *PostgreSQL layout used in the article examples*

The Path to the Data

As stated before PostgreSQL stores database objects, including tables (and their tuples) as files into the PGDATA directory (under *base* if no tablespace has been defined). Such files are named after the OID assigned to the object itself, even if a few maintenance commands can break this rule. The system administrator can find exactly which file correspond to which database object and vice versa using `oid2name` command that is available into the *contrib* module or querying the system catalogue.

As Listing 10 shows the `oid2name` command can be used to see all the OIDs of the available databases (if invoked without arguments) and can provide also the OID of a specific table into a specific database. In the example of Listing 10 `oid2name` provides 16387 as OID for the database *bsdmagdb* and 16390 as OID for the table *magazine*. This means that under *PGDATA/base* there must be a directory named 16387 (OID for *bsdmagdb*) which contains, among other files, a file named 16390 that contains the tuples of the *magazine* table. As readers can see, the `ls(1)` of such file shows a size of 8 KB even if only a few tuples have been stored into the table. This is the default size for PostgreSQL data page and means that each table is aligned to a 8 KB size. In order to see how the required size changes let generate ten thousand fake tuples using the special function `generate_series()` as shown in Listing 11. The size of the file on disk is now about 5.4 MB. Considering that each page is 8 KB in size, it means that the table contains

```
5652480 / ( 8 * 1024 ) = 690 pages
```

as confirmed by the query on the system catalogue in Listing 11.

Backup your data

While PostgreSQL is very reliable and stable the DBA has to backup the cluster content to avoid data loss due to hardware failure, misconfiguration, application or human errors, and so on. There are several ways of doing backups which can be grouped mainly into *logical*

Listing 13. *Preparing the system for PITR*

```
~> psql -U bsdmag -c "ALTER TABLE magazine ADD COLUMN ts timestamp default 'now'::text::timestamp;" bsdmagdb
ALTER TABLE
~> psql -U bsdmag -c "SELECT pg_start_backup('MY FIRST PITR');" bsdmagdb
 pg_start_backup
-----------------
 0/3C000020
(1 row)

~> cp -Rf /postgresql/cluster1/* /postgresql/cluster2/
~> rm /postgresql/cluster2/pg_xlog/*
~> rm /postgresql/cluster2/postmaster.pid
~> psql -U bsdmag -c "SELECT pg_stop_backup();" bsdmagdb
NOTICE:  pg_stop_backup complete, all required WAL segments have been archived
 pg_stop_backup
----------------
 0/3C000094
(1 row)


bsdmagdb=# INSERT INTO magazine(id, title)
VALUES( generate_series(1, 200000), 'BATCH-1');
INSERT 0 200000
bsdmagdb=# INSERT INTO magazine(id, title)
VALUES( generate_series(200001, 400000), 'BATCH-2');
INSERT 0 200000
bsdmagdb=# INSERT INTO magazine(id, title)
VALUES( generate_series(400001, 600000), 'BATCH-3');
INSERT 0 200000
bsdmagdb=# INSERT INTO magazine(id, title)
VALUES( generate_series(600001, 1000000), 'BATCH-4');
INSERT 0 400000
bsdmagdb=# SELECT ts::time, title FROM magazine GROUP BY ts,title ORDER BY title;

-----------------+---------
 08:22:42.982515 | BATCH-1
 08:22:51.401579 | BATCH-2
 08:23:03.243077 | BATCH-3
 08:23:17.011491 | BATCH-4
(4 rows)
bsdmagdb=# TRUNCATE TABLE magazine;
TRUNCATE TABLE
```

backup and *physical backup*. Logical backup is a way of *one-shot* backup that is transaction consistent: the system extracts the data from the cluster and dumps it to a backup media respecting transaction boundaries; the DBA does not need to access the PGDATA directory. The physical backup requires the DBA to access the PGDATA directory and to archive each single data file. This of course does not respect transaction boundaries, and therefore the system will need to do a transaction-replay once started. To perform such transaction replay the database will need also the WALs, the logs its already uses to survive crashes. How the WALs are archived defines also the level of physical backup: it is possible to do an archiving only between two full logical backups, as well as do continuos archiving, as well as send logs to another machine that will replay them to act as a *clone* (this is in short the way replication works).

Logical Backup: pg_dump

The standard way of doing logical backups using the `pg_dump(1)` command which dumps a database content as an SQL text file (plain or compressed). The command `pg_dump(1)` has many options and can be used to dump only the data, only the database structure, a single table or object, or the whole content. There is also another tool named `pg_dumpall(1)` which is used to backup a whole cluster instead of a single database. The usage of both tools is similar and Listing 12 shows how to backup the example database. The installation from the ports also creates a script under `/usr/local/etc/periodic/daily` which, when enabled, performs a `pg_dump` of all the databases available in the cluster (excluding *template0*).

Physical Backup and Point in Time Recovery (PITR)

PITR is an interesting backup technique developed in the 8 series that is based on physical backup and allows restoration at a specific time in the past. The idea behind PITR can be summarized as follows: the system keeps a track of the WALs that contain, at any point in time, the *image* of the database status. When the restore is required the daemon is stopped and restarted simulating a crash (i.e., a dirty status). The database then starts rolling the WALs in order to redo all the transactions for internal consistency. Specifying how many WALs must be rolled it is possible to control at which time in the past the instance must be restored.

In order to see PITR in action we need a place to store WALs, so we create a directory `/postgresql/pitr` (see the red box in Figure 1) and configure the following options into `postgresql.conf`:

```
wal_level = archive
archive_mode = on
archive_command = 'cp -i %p /postgresql/pitr/%f'
```

Listing 14. *Testing PITR*

```
~> cat /etc/rc.conf | grep postgres
postgresql_enable="YES"
postgresql_data="/postgresql/cluster2"
~> cat /postgresql/cluster2/recovery.done
restore_command = 'cp /postgresql/pitr/%f "%p"'
recovery_target_time = '2012-01-20 08:23:18'
~> service postgresql start
~> psql -U bsdmag -c "SELECT ts::time, title, count(title) FROM magazine GROUP BY ts,title ORDER BY title;" bsdmagdb
       ts        |  title  | count
-----------------+---------+--------
 08:22:42.982515 | BATCH-1 | 200000
 08:22:51.401579 | BATCH-2 | 200000
 08:23:03.243077 | BATCH-3 | 200000
~> cat /postgresql/cluster2/recovery.done
restore_command = 'cp /postgresql/pitr/%f "%p"'
recovery_target_time = '2012-01-20 08:23:18'
```

which inform the cluster when and how to archive the WALs. After having restarted the instance we can connect and simulate a workload. In order to better understand the example we modify the *magazine* table to keep timestamps associated to the tuples (see Listing 13).

The first step to enable PITR is to have a physical copy of the PGDATA directory, excluding the *pg_xlog* content; please note that it is not important where and when the backup is done, as well as how much does it take to copy PGDATA, but the cluster must be informed that the copy is in progress, so we have to surround the copy process using the `pg_start_backup()` and `pg_stop_backup()` functions. As Listing 13 shows, to keep things simple, our copy is done onto a mirrored directory of PGDATA called `/postgresql/cluster2` (see the red box in Figure 1). While the copy is in progress the cluster can continue to work (i.e., user connections are allowed). At the end of the backup we insert, using four batch statements, a million tuples with four different timestamps, and then we erase of all them (see Listing 13). This is the simulation of an application error, but other (more complex) disasters can be simulated.

Now imagine we want to get back in time at 08:23:18 (the whole third batch done, the fourth not commited yet – timestamps in Listing 13 are evaluated when the INSERT begins). In order to restore the system at that time we have to:

- stop the instance;
- change the PGDATA directory into `/etc/rc.conf` to point to our physical copy;
- create a `recovery.conf` file in the new PGDATA directory that will contain the position of the archived WALs as well as the time at which we want to restore the instance;
- start the instance and wait for the recovery to complete.

As Listing 14 shows, once the instance is restarted the content of the *magazine* table is switched back in time to only the third batch and all its tuples.

It is worth noting that once the backup is completed the system moves the recovery.conf file to recovery.done and that another text file, `backup_label`, is moved to `backup_label.old`. The latter file contains information about when the physical backup started, the label of the backup (as passed to `pg_start_backup()`) and the WAL information. Both files can be used as additional information to understand the status of the cluster.

As readers can see, PITR is a cluster-wide backup technique and is very powerful. It is worth noting that there is no limitation to the amount of time PITR can work, as well as the archiving method: it does suffice to have enough WAL archiving space and to use the right command (e.g., *cp*, *scp*, *tar*, etc) to be able to archive a whole database history and to recovery it at any time in the past.

On The Web

- PostgreSQL official Web Site: *http://www.postgresql.org*
- ITPUG official Web Site: *http://www.itpug.org*
- PostgreSQL 9.1 Documentation: *http://www.postgresql.org/docs/9.1/interactive/index.html*

Which backup strategy?

The choice among the right backup strategy depends on the cluster workload. Logical backups are the simplest, and should be used each time taking one of it does not require too much time or space (or when, of course, you need a full backup dump). When it is required too much time (e.g., the database is huge) or keeping a set of logical backups requires too much space, physical backup should be take into account.

You can even mix the logical and physical backup archiving WALs for PITR only in the time frame between two logical backups. This will give you an incremental backup up to the next full (logical) backup. Be sure to test that the backup strategy fits your needs and that it is working properly.

Summary and Coming Next

This article briefly covered the basis of PostgreSQL installation and usage. In the next article we continue exploring PostgreSQL glancing at MVCC and vacuum.

LUCA FERRARI

Luca Ferrari lives in Italy with his wife and son. He is an Adjunct Professor at Nipissing University, Canada, a co-founder and the vice-president of the Italian PostgreSQL Users' Group (ITPUG). He simply loves the Open Source culture and refuses to log-in to non-Unix systems. He can be reached on line at http://fluca1978.blogspot.com.

OpenBSD with SUN JAVA & Netbeans

This article is designed for Java developers who wants to have a good operating system for their works.

What you will learn…

- You will learn how to setup SUN Java system and Netbeans on your OpenBSD desktop.

What you should know…

- How to install OpenBSD. See official website.
- You should have OpenBSD with X11 already installed and an Internet connection.

This article is based on OpenBSD 5.0 version. Let's go!

Installing Ports

We need to install port in first time. With it, we will be able to build JAVA jre.

Downloading:

```
ftp ftp://ftp.fr.openbsd.org/pub/OpenBSD/5.0/ports.tar.gz
```

Extraction:

```
tar zxvf ports.tar.gz -C /usr
```

Accept SUN Lisence

We will add that we accept the SUN JAVA license in `mk.conf`. If we don't, we can't be able to build JAVA jre.

```
echo „ACCEPT_JRL_LICENSE=Yes" >> /etc/mk.conf
```

Dependencies

Install wget to get dependencies

Firt we will export a varible, it contain the way to get OpenBSD packages.

```
export PKG_PATH=ftp://ftp.fr.openbsd.org/pub/OpenBSD/5.0/
                    packages/i386/
```

Now, we can install software, let's install `wget`:

```
pkg_add -iv wget
```

Stock dependencies

Now we ready to download all dependencies needed to build SUN JAVA.

But we need to stock it in the right place:

```
mkdir /usr/ports/distfiles
cd /usr/ports/distfiles
```

Get dependencies

Ready? Downloading...

```
wget http://download.java.net/jdk6/6u3/promoted/b05/jdk-
6u3-fcs-src-b05-jrl-24_sep_2007.jar

wget http://download.java.net/jdk6/6u3/promoted/b05/jdk-
6u3-fcs-bin-b05-jrl-24_sep_2007.jar

wget http://download.java.net/jdk6/6u3/promoted/b05/jdk-
6u3-fcs-mozilla_headers-b05-unix-24_sep_2007.jar
```

There are some dependencies that can not be downloaded by `wget` because of licence accepting. So get it by hand:

Get the file

`bsd-jdk16-patches-4.tar.bz2`

via *http://www.eyesbeyond.com/freebsddom/java/jdk16.html*

Get the file

`jdk-1_5_0_16-fcs-src-b02-jrl-28_may_2008.jar (JRL license)`

Get the file

`jdk-1_5_0_16-fcs-bin-b02-jrl-28_may_2008.jar`

via *http://download.java.net/tiger/tiger_u16/*

Get the file

`bsd-jdk15-patches-9.tar.bz2`

via *http://www.eyesbeyond.com/freebsddom/java/jdk15.html*

Get the file

`jdk-1_5_0_16-solaris-i586.tar.Z`

via *http://www.java.sun.com/products/archive/j2se/5.0_16/index.html* (Need and account on Oracle website)

Get the file

`xalan-j_2_7_0-bin.tar.gz`

via *http://archive.apache.org/dist/xml/xalan-j/*

And put all in `/usr/ports/distfiles/`.

Make JAVA

Yes! We are now ready to build SUN JAVA system

```
cd /usr/ports/devel/jdk/1.6
make install
```

Compilation take a lot of time, take a coffee.

Install Netbeans

The installation of Netbeans is more easier than JAVA because Netbeans is packaged in OpenBSD.

So install Netbeans easily with the classic:

```
pkg_add -iv netbeans
```

Product Version: NetBeans IDE 6.9.1 (Build 201007282301)
Java: 1.6.0_03-p4; Java HotSpot(TM) Client VM 1.6.0_03-p4-root_20_dec_2011_14_37-b00
System: OpenBSD version 5.0 running on i386; ISO8859-1; en (nb)
Userdir: /home/gui/.netbeans/6.9

Figure 1. *Launch Netbeans*

Conclusion

Congratulation you can now code code and code again with your favourite language on your favourite OS.

GUILLAUME DUALÉ

System Administrator; OpenBSD addict! And pfSense contributor. g.duale@otasc.org

Data Classification Policy

Dear readers, I have been struggling with ideas for what to write about this month. I was toying with the idea of writing a "newbie" guide to PC-BSD 9.0, but the PC-BSD team has done such a wonderful job with the product. It's incredibly easy to install and use, so there's not much content there.

What you will learn…

- How to put forth a data classification policy that even lay-people can understand.

What you should know…

- There are no technical prerequisites for this article.

As I poked around PC-BSD 9.0 in Virtualbox, I started to think about what I feel passionate about. What would I like to share with the readers of *BSD Magazine*? What would interest my audience? I realized that like me, the lion's share of you are probably system administrators.

A good sysadmin realizes that security is more than firewalls, encryption, patching, and other technical considerations. One common saying is: *The only secure computer is one that's not plugged into the network*. Humbug! A clever intruder will easily trick the user into plugging that Ethernet cable back into its socket. The weakest point in any network is the human element.

I want to share with you my data classification policy. While it appears much different in this current form, I developed this policy by starting with the sample data classification policy in *The Art of Deception: Controlling the Human Element of Security* by Kevin D. Mitnick (Author), William L. Simon (Author), and Steve Wozniak (Foreword). I then spent many weeks refining and simplifying that policy into what you see here.

I highly recommend *The Art of Deception: Controlling the Human Element of Security*. It is an outstanding resource. My opinion: The book should not only be required reading for technical professionals; the book ought to be required reading for everyone. For a more in depth look at the technology scams that an intruder may use to compromise security you will want to take a look at *Stealing the Network: The Complete Series Collector's Edition, Final Chapter, and DVD* by Johnny Long (author), Timothy Russel (author), and Timothy Mullen (author).

Introduction

Data classification is fundamental to the security of the Organization's information. Without explicit data classification, any decision about the sensitivity of information is left in the hands of individuals. This policy provides a framework for answering the questions: *How should I handle my data*? and *Who can receive my data*? Data classification is an important defense against social engineering.

Data Classification

Data shall be classified according to three categories: public, controlled, and secret. Managers of individual departments have complete authority and obligation to classify departmental specific data into one of these categories.

Applicable Rules

Depending on which category your data is classified as, there can be two rules that apply to how the data is handled. These rules are:

The Three Questions (3Q)

The 3Q rule means that you may not share it or transmit it to anyone else, including other Organization employees, until you establish three attributes of the recipient:

- Identity: Is the recipient really who he says he is?
- Authority: Is the recipient authorized to have the data?
- Need to know: Does the recipient need the data?

Encryption

The *encryption* rule means that you must encrypt the data if it leaves the protection of the Organization's production network and if the data is in a digital format. You may decrypt the data only while you are working on it. You may not decrypt the data while you are in any public place, such as a café or airport.

Example 1

You do not have to keep the data in a locked briefcase as you take it from your car to your home.

Example 2

You may not bring the data into a café unless you keep it in a locked briefcase. You may not read or work with the data while you are in the café.

Additionally, if the encryption rule applies to your data, then it must be destroyed if and when you dispose of it. Physical data must be shredded. Files can simply be deleted. CDROM's must be broken. Laptops, computers, and external drives must be securely erased by the IT department.

Possible Classifications

Public: No rules apply to this data.
Controlled: Only the 3Q rule applies to this data.
Secret: Both the 3Q and the encryption rule apply to this data.

Cryptography Standards

Acceptable cryptography standards include SSL or TLS for secured web sites: AES, Blowfish (or derivatives such as Twofish or Threefish) or 3DES encryption algorithms for the transmission of encrypted files. Certificate/key-pair based e-mail encryption is acceptable. The IT department can recommend and provide training for encryption procedures.

Passwords are always classified as *secret* and should never be transmitted electronically. If you are sending someone encrypted data, then the decryption password is classified as *secret* and should be communicated in person or over the telephone only.

Inheritance Of Classifications

Any set of data shall inherit the strictest classification assigned to any subset of that data. For example, if social security numbers are classified as secret, and applications for employment contain social security numbers, then applications for employment must be classified as secret.

Default Classification

Any data not explicitly classified will be classified as *Controlled*.

Appendix A: Classification Matrix

	3Q Rule Applies	3Q Rule Does Not Apply
Encryption Rule Applies	Secret Data	Not Applicable
Encryption Rule Does not apply	Controlled Data	Public Data

Appendix B: Suggestions for Establishing Identity

Individual departments must establish procedures for following the 3Q rule that have the best combination of maximizing security while minimizing the affect on workflow. The most difficult part of the 3Q rule is establishing identity. Remember that Caller ID is NOT a good method for establishing identity. Caller ID's can easily be spoofed. Here are some examples of methods for establishing identity:

Callback: Look up internal requester in the Organization directory. Use 4-1-1, the Yellow Pages, or the White Pages to look up external requester. Call the requester back using the listed number to verify identity. If you use this practice, it will be helpful to have directories of external organizations you deal with frequently.

Shared Secret: Use a shared secret that you have established with another person or organization.

Digital Certificates: Request a digitally signed e-mail or certificate that is verified through a third party, such as VeriSign or Comodo. Comodo offers free personal SSL certificates for e-mail here: *http://www.instantssl.com/ssl-certificate-products/free-email-certificate.html*.

Personal Voice Recognition: This is one of the most secure methods for establishing identity.

In Person with ID: Only government issued ID should be accepted. Student ID should not be accepted. Employer ID should not be accepted unless the employer is employed by this Organization, a partner, or well trusted vendor of this Organization.

Appendix C: Matrix of Encryption Solutions

Solution	Compliant Algorithm Choices	Does this solution comply with the Data Classification Policy?	
		Yes	No
Office Password Protection	N/A – Not Encryption		X
Zip Genius Encryption	No Compliancy		X
Telnet Protocol	None		X
File Transfer Protocol	None		X
Secure Sockets Layer (SSL, TLS, or HTTPS)	3DES	Y	
WinZip 9 or later, 7zip	AES	Y	
TrueCrypt Encryption	AES	Y	
dsCrypt Encryption	AES	Y	
PGP or GPG	AES or 3DES	Y	
E-Mail Certificate Based Encryption	AES or 3DES	Y	
Windows Vista Bitlocker Encryption	AES	Y	
Mac OSX 10.5+ Secure Disk Image	AES	Y	
Secure Shell (SSH) or SFTP	Blowfish	Y	

Appendix D: Avoiding Confusion

The Organization strongly recommends that every controlled and secret document incorporates a footer to remind employees of the document's classification. Recommended color coding is green for public (classification footers for public documents is optional), blue for controlled, and red for secret. Here are examples:

- Classification: Public
- Classification: Controlled
- Classification: Secret

Another option is to print information of varying classifications on different colored paper.

Example 1

In Iraq, the U.S. Marine Corps prints all information classified as secret on yellow paper. This is not an enforced rule, but a convenient practice for quickly identifying the classification of printed materials.

Example 2

A work group might choose to print controlled information on light blue paper and secret information on pink paper.

Work groups can set their own procedures for this.

Conclusion

The IT Department takes every available technical measure to protect the Organization's confidential data, but technical measures cannot protect the Organization from human mistakes. It is up to each department and work group to follow this guide. It is up to individual managers to decide how various files and data ought to be classified.

TOBY RICHARDS

Toby Richards has been a network administrator since 1997. Each article comes straight from the notes that he takes when doing a new project with *BSD. Toby recommends bsdvm.com for your hosting needs because they provide console access to your virtual machine.

Load Balancers

Enterprise Load & Service Availability

The world is a complex place. A term that means one thing to one person may mean something completely different to someone else.

What you will learn…

- Basic Load balancing
- Load balancing security considerations
- How to find out if a website is using a Load Balancer
- Implementing a Load Balancing in an OpenBSD environment

What you should know…

- Internetworking
- Web Hosting
- Routing & Address Pools (Bitmask, Random, Source-Hash, Round-Robin)
- Unix Administration (pfsync, rsync)

Take Load Balancers for example. How many different Load Balancers can you think of? Types of Load Balancers:

- Layer-2 Load Balancing (bonding)
- Layer-4 Load Balancing
- Layer-7 Load Balancing (reverse proxy)
- Hardware SSL acceleration or offload
- DNS Load Balancing
- Link Load Balancing
- Load Balancing Optimization / Compression,
- WAN Load Balancing Optimization
- SIP Load Balancing

Load Balancers handle *load* and people have been thinking about how to handle *load* as it relates to resource availability for a very long time. Frequently when people talk about Load Balancers today they are specifically thinking about how to handle web server traffic.

Fact

As web traffic patterns have become more unpredictable, the industry has demanded a way of scaling capacity to meet that demand (load).

The single box solution has proven itself to be unacceptable regardless of how *beefy* the box is. Let's

Definitions

- OpenBSD = A Unix-like OS descended from BSD
- LOAD = A measurement of the # of simultaneous requests for a service at a given point in time
- LOAD BALANCER = A software or hardware solution that helps you manage load
- CARP = Common Address Redundancy Protocol
- VRRP = Virtual Router Redundancy Protocol
- HSRP = Hot Standby Router Protocol
- VNIC- Virtual network interface
- PenTester = A security professional who evaluates corporate security and makes expert recommendations

Tools

You should become familiar with some of the tools mentioned in this article:

- ldb.sh
- Halberd 0.2.4

not even mention what happens when you add concepts like *virality* to the mix- you start dealing with tsunami level load.

Current Issue

These days you have a bunch of smart business people who are spending a significant amount of time trying to „design for virality" with minimal consideration of the impact on network services/operations. Why do they do this? They do this because Virality = Money. What does this mean for you? Virality = DEPLOY LOAD BALANCERS.

Problem

A large number of service requests to a web server at one time will knock over a typical server.

Load

When we talk about web hosting and load, load in this sense means the number of concurrent requests for services received by a host or cluster of hosts at a particular point in time.

Enter The Load Balancer

A Load Balancer is a device that decides where to send traffic requests. When you point your web browser to Microsoft.com or CNN.com, contrary to popular belief you are not going to a single server. You are hitting one of many servers. Organizations of any substantial size typically have thousands of web servers ready to handle your request.

TIP

You have a lot of options when it comes to deciding how you will distribute load across different servers.

Frequently a Load Balancer is deployed as a dedicated piece of hardware with multiple interface cards that can connect to multiple hosts.

How It Works

A Load Balancer receives the request from your computer (most commonly in the form of a request from your web browser). It evenly distributes the load across a group of servers. The Load Balancer is actually distributing the load across what is called a *cluster* of servers. The concept is that you don't care which server you are sent to because all the servers are the same.

Up Time And Availability

The goal is to answer the request. You don't care which server answers the request. The largest types of environments will need this type of equipment to support hundreds of thousands of users trying to connect at the same time.

Fact

A Load Balancer is critical in large environments.

Max Flexability

You can decide how the request is distributed. You can distribute based on load or on content. Maybe you want one of the servers to provide video, one server to provide the web page, and maybe even another server to provide images? You can decide exactly how to separate the load across all the servers.

Security

This creates a huge security problem. You have all of these connections coming into the Load Balancer and being distributed across all of these servers so you will want to be sure that all of your servers have the very latest `builds/patches/updates/security` software running on them.

TIP

Make sure that all hosts are running the very latest security software and that all programs are current and up to date.

Know Thyself

Make sure that you know the full extent of the Load Balancers security capability and associated vulnerabilities. Specifically make sure that there is no hijacking of cookies.

TIP

Determine what the known exploits are for the Load Balancer itself.

Fact

When the Load Balancer is compromised an attacker can send traffic to another destination.

Proxy

Folks frequently deploy proxy servers for extra protection. You can offer additional protection to your end users by introducing a Proxy server.

This is a type of server or a series of servers that are designed to sit between the users and the Internet. Its job is to take any request that an in-house user sends out to a web server and stop it. Once it stops the request- it takes over the request and re-sends it on the user's behalf.

Fact

Reliability and compatibility are the most frequently overlooked elements of any Load Balancer / Proxy server deployment.

The proxy server receives the response. The proxy server inspects the response and confirms that it does not contain any malicious malware or viruses. The proxy server can even check to see if the user has permission to go to that website. If everything checks out OK the proxy server will send the answer (results) back to the end user.

Fact

A proxy server represents an extra step between you and the internet.

Clearly a Load Balancer and Proxy server become the new bottle necks so you will want to make sure that the hardware is scaled properly to handle the load going through the wire.

How To Check If A Website Uses A Load Balancer

Frequently PenTesters (Penetration Testers) are very interested in knowing if a Load Balancer or Proxy is in place since they can be responsible for the incorrect results that today's security tools return.

1. First enable your browser to show you *live HTTP header requests*.
2. Retrieve public information about a domain/host by searching for the site name on *searchdns.netcraft.com* or *DomainName.com*
3. Take note of the operating system that is being used as it will be listed in the results from *searchdns.netcraft.com*
4. Learn how to use DIG to find additional information.

Tool: Dig

NAME: dig – DNS lookup utility.

```
SYNOPSIS : dig [ @server ] [ -b address ] [ -c class ]
[ -f filename ] [ -k filename ] [ -p port# ] [ -t type ]
[ -x addr ] [ -y name:key ] [ name ] [ type ] [ class ]
[ queryopt... ]
```

Dig is a tool for finding out additional information from DNS servers. Dig does a SND lookup and displays the answers that are returned from the name server. Dig is primarily a troubleshooting tool.

5. Find and use Halberd 0.2.4 (14-Aug-2010) (*http://pydoc.net/halberd/0.2.4/Halberd.version*)
6. Run `./Halberd -v DomainName.com`
7. You will want to use LBD. The LBD Load Balancing Detector 0.1 will check to see if a given domain is using load balancing. The program was written by Stefan Behte http://ge.mine.nu and is currently in a proof on concept state.

TIP

LBD checks for DNS-Load Balancing and then checks for HTTP Load Balancing. Returns the state.

8. Run `.ldb.sh www.DomainName.com`

Fact

Load Balancers are expensive and while we all love f5, sometimes the budget just won't support that type of capital investment.

Build Your Own Load Balancer

You can build your own basic Load Balancer with OpenBSD. Before you start, read up a bit on address pools and review the four methods for using address pools:

- Bitmask
- Random
- Source-Hash
- Round-Robin.

Open BSD will let you Load Balance incoming connections and Outgoing Traffic.

Load Balancing Incoming Connections

Achieving Load Balancing by using address pools is very straight forward. You distribute all inbound web server connection requests across a web server farm.

The command will look something like this:

```
web_server = „{10.0.0.10, 10.0.0.11, 10.0.0.13}”
match in on $ext_if_proto tcp to port 80 rdr-to
$web_servers round-robin sticky-address
```

Every connection requests will be sent to a new web server in the group following- round-robin style. Additionally, in this scenario, a connection from the source server will continue to be sent to the same destination server.

CARP

CARP is also very popular with the *Load Balancer Do It Yourself* (LBDIY) community. CARP gives you the

ability to achieve system level redundancy which can be an important part of any enterprise Load Balancing plan.

CARP stands for Common Address Redundancy Protocol and lets you have a bunch of hosts and share an IP address so that if one server fails, another server in the CARP group can answer the request. This provides a very basic level of redundancy at the server/host level. Additionally, you can share load between members of a CARP group.

Fact

Everyone thinks CARP is a firewall-specific protocol. Not true.

TIP

Use CARP to guarantee service availability and load sharing of/and between web servers.

Historically speaking the OpenBSD folks wanted to distribute a free implementation of VRRP and HSRP but were unable to do so for patent violations. The OpenBSD team immediately went to work on a VRRP variant.

Today, to deploy CARP the first thing you need to do is group several physical computers together under one or more virtual addresses. Since CARP is a multi-cast protocol, one of the systems will need to be the master and will be required to respond to all packets destined for the group. The other systems (backups) will just be in standby mode waiting for any abends `/crashes/K.O.` situations to take place.

When the master gets knocked over, the other hosts in the CARP group begin to advertise. The host that is able to advertise the most frequently becomes the new master. Even if the original master comes back on-line right away it will only be allowed back into the group as a backup server.

TIP

CARP only manages the VNIC- Virtual network interface. You will need to use pfsync or rsync to synchronize data at the application level.

CARP Setup

This is done via the sysctl and ifconfig commands. The syntax information is freely available on the Internet.

CARP Load Balancing

CARP provides two different methods for load balancing incoming network traffic among a set of CARP-enabled hosts: ARP balancing and IP balancing. Both methods require that you build a load balancing group.

Resources

- Search *www.SearchDNS.NetCraft.COM*
- Halberd Load Balancer Detector *www.pydoc.net/halberd/0.2.4/Halberd.version*
- LBD by Stefan Behte *http://ge.mine.nu*

Again, the syntax information is freely available on the Internet.

Summary

When it comes to managing load the only thing you can count on is nothing. Work to be prepared for all types of unpredictable traffic events. Evaluate affordable hardware solutions that can give you best of breed functionality without sacrificing manageability and security. When confronted with price constraints evaluate Load Balancers that are software based. Be aware that the Load Balancer is a visible part of the network and will serve as a big bottleneck and an even bigger security target.

Thank you for taking the time to read this article.

RICHARD C. BATKA, CONSULTING CIO/CTO

Richard C. Batka is a business & technology executive based out of New York who currently provides consulting services at the Chief Information Officer (CIO) and Chief Technology Officer (CTO) level to select Fortune 500 companies. Mr. Batka has worked for global leaders Microsoft, PricewaterhouseCoopers, Symantec, Verizon, Thomson Reuters and JPMorgan Chase. A graduate of New York University, he can be reached at rbusa1@gmail.com or followed on Twitter at http://twitter.com/richardbatka.

Anatomy

of a FreeBSD Compromise (Part 3)

Continuing in our security series, we will look at the tools essential to securing and exploiting systems.

What you will learn…
- How to use network and O/S security tools

What you should know…
- BSD and network administration skills

In the previous articles, the author looked at the culture and processes behind hacking exploits, as well as some possible real-life examples. In this article we will look at some of the tools used to penetrate, test and secure devices as well performing analysis and discovering vulnerabilities. While the examples here are non-destructive, it is recommended that these tests are carried out on a private test network and definitely not on the Internet or without your employer's approval. To do so may well be in breach of your ISP's or employers Acceptable Use Policy and could lead to legal action against you.

Ethics

One of the long standing arguments on the Internet is what to do with traffic from hackers, spammers etc when it arrives on your network. As it is relatively easy to identify where your attack is coming from (once you have identified your incident and provided multiple attackers are not involved) there is great temptation to *fire back* at the lowlife that has abused your system. This is a very natural reaction, yet in the authors opinion this is not the best route to take. First of all, you are raising a flag to say that there is intelligent life at the end of the wire, so possibly opening yourself up to more attacks. Secondly, it has not been clarified in law (in the UK at least) that your response could not be construed as an aggressive attack. This in effect means if you "fire back" in the eyes of the law you are just as guilty as the original offender. The situation is complicated by the fact that often while the device at the other end of the wire is compromised, the owner is blissfully unaware and the last thing on thing on their mind is to attack your web-server. The safest route is to drop suspect packets on the floor, rather than launching a full scale attack on the origin in the hope that you will bring fairness to the situation. If repeated incidents occur, pass the incident to the originators ISP or worst case to law enforcement. That said, I have included both sticky and non-sticky honeypots in the tools list as defenses – I will leave the lawyers to work out whether or not tar-pitting an offensive packet can be considered an offensive act or not.

Requirements

For these series of tests the author will be using a combination of bridged Virtual Machine running on a desktop host connected to an internal LAN. Two FreeBSD boxes (One configured as a virtual machine and the other a live server) are available as well as other devices on the LAN, Border.merville.intranet and hacker.merville.intranet are the victim and attacker respectively. `Border.merville.intranet` is a copy of FreeBSD 6.1, with Apache, PHP, and MySql installed which was originally compromised but repaired,

`Hacker.merville.intranet` is FreeBSD 9.0 Release running TWM and our tools added via `pkg_add` as required. Various other servers and desktop machines will be used as targets in later exercises, but to cover the initial fundamentals 2 machines are sufficient. I am using a Cisco switch on my LAN, but a hub (or virtual LAN) will work as well if not better. Results will vary between a switch and a hub, depending on the configuration of the switch, as not all packets will be present on all ports whereas they will be on a hub. If a virtual LAN is used you may need to tweak the security settings of the virtual machines accordingly. The results from your network will vary considerably from the test LAN here depending on what devices etc. are present.

Tools

While there are a lot of command line tools available under *BSD that will help the administrator discover vulnerabilities, there are a number of bootable ISO's available that include collections of tools that are useful across software platforms. These are useful in that all the utilities are rolled together, and can be run discretely from a CDROM or bootable USB stick. The majority of these utilities are available in one form or another under *BSD, with the exception of the O/S specific tools (e.g. those that are are available on the Hirens BootCD) but will require further tweaking (installation of additional scripts/templates etc.) to tun under *BSD. Some require a commercial license or will benefit from commercial support. It is up to the system administrator to evaluate the best tool for the job, and sometimes it is prudent to enlist the help of specialists especially in the fast moving arena of systems security. Please refer to *Table 1 – common Security and Network Tools* which covers the majority of popular security tools across the *BSD/ Windows platform. In the world of security, there is no

Table 1. *Common security and network tools*

Name	Description	Category	Commercial License / Support	Website
netstat, telnet, ping, dig, ps, netcat, top, tcpdump etc.	General purpose administration and networking tools. Should be in every sys-admins armoury.	Application		N/A
Nmap / Nmapsi4	Port scanner. Checks for tcp enabled devices on networks and open ports	Application		*http://nmap.org*
Ntop	Network traffic probe	Application	Yes	*http://www.ntop.org*
Nessus	Vulnerability scanner. Using the professional feed will identify open ports, applications and their corresponding vulnerabilities	Application	Yes	*http://www.nessus.org*
Wireshark	Network protocol analyzer and packet sniffer	Application		*http://www.wireshark.org*
Metasploit	Penetration testing software. Performs network discovery and vulnerability verification	Application	Yes	*http://metasploit.com*
Snort	Intrusion detection system (IDS)	Application	Yes	*http://snort.org*
Honeyd	Non-sticky honeypot – used to lure attackers	Application		*http://honeyd.org*
Labrea	Sticky honeypot – As above but sends them to a tar pit and slows down attacks	Application		*http://labrea.sourceforge.net*
Trinity rescue kit	General purpose tookit. Allows cloning of media over network, MS Windows password reset, deleted file recovery etc.	Bootable CD		*http://trinityhome.org*
BackTrack Linux	Premier Linux distro tuned to the needs of the security professional. Just about every tool you will require in one ISO. The ninja hacker's weapon of choice.	Bootable CD		*http://www.backtrack-linux.org*
Knoppix S-T-D	Live distro with a large collection of security tools	Bootable CD		*http://s-t-d.org*
Hiren's BootCD	Live CD aimed at virus removal etc. mainly aimed at Microsoft desktops	Bootable CD		*http://www.hiren.info*

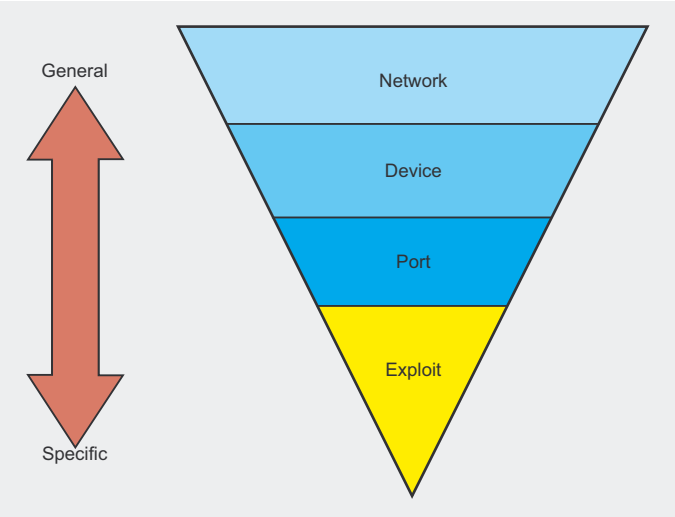


Figure 1. *Attack strategy*

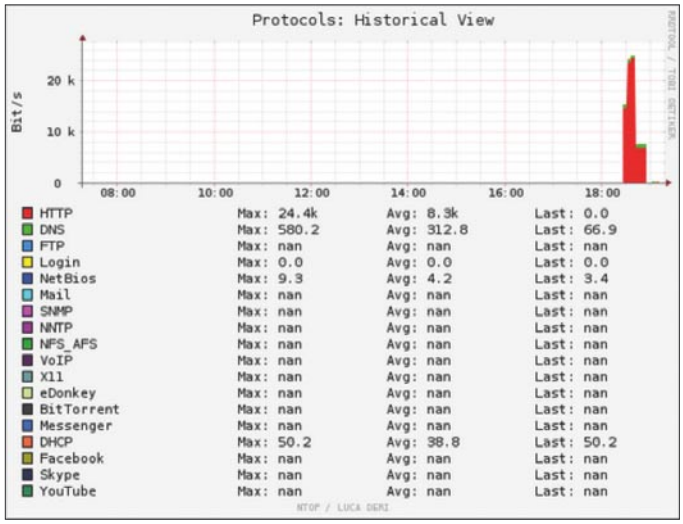
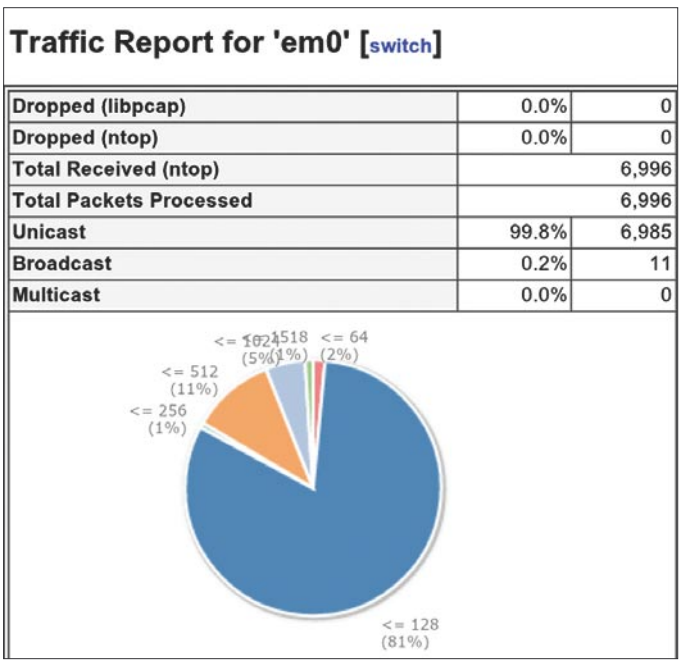


Figure 4. *NTOP showing protocols*



Traffic Report for 'em0' [switch]

Dropped (libpcap)	0.0%	0
Dropped (ntop)	0.0%	0
Total Received (ntop)		6,996
Total Packets Processed		6,996
Unicast	99.8%	6,985
Broadcast	0.2%	11
Multicast	0.0%	0

Figure 2. *NTOP showing network traffic*

Host	Location	IP Address	MAC Address	Community	Other Name(s)
hacker		192.168.0.131	08:00:27:20:AA:9C		
intel		192.168.0.132	00:16:E6:58:A0:75		
cisco (vlan 3)			00:0D:65:1C:1A:C2		
bridge sp. tree/osi route:00:00:00			01:80:C2:00:00:00		
border.merville.intranet		192.168.0.254	00:11:D8:0C:BD:8F		
cisco cdpd/vtp:cc:cc:cc (vlan 3)			01:00:0C:CC:CC:CC		

Inbound vs Outbound	Nw Board Vendor	Hops Distance	Host Contacts	Age/Inactivity	AS	Fingerprint
	CADMUS COMPUTER SYSTEMS		8	4:28 24 sec		
	GIGA-BYTE TECHNOLOGY CO.,LTD.		10	4:31 21 sec		
	Cisco Systems		2	4:48 3 sec		
	Bridge Sp. Tree/OSI Route		1	4:48 3 sec		
	ASUSTek Computer Inc.		8	4:49 2 sec		
	Cisco CDPD/VTP		1	4:34 15 sec		

Figure 3. *NTOP showing clients – columns wrapped for clarity*

single application that covers all bases, so using a wide range of tools allows the system admin to examine the environment from many different angles. Anti-virus and firewall software has been omitted from the list – these are platform specific and it is taken as read that the system administrator will have these systems in place.

Strategy

One of the reasons that bot-nets are highly effective is the analysis and gathering of repetitive data. In human terms, we need to drill down from the macro (general) to the the micro (specific). This is simplified in *Figure 1 – moving from the general to the specific*. What networks have we access to? What devices are on this network? What ports are open on these devices? And finally, what exploits are available on these ports? If we do the math, there are 27 different types of web-server software alone currently available according to Wikipedia, and that is just one application running on port 80. Taking into account the 65536 TCP/IP ports and the possibility that the software has not been configured securely, patched or is just plain broken, this leaves a lot of ground to cover and that is just one server. Adopting the hacker mentality for a moment, a form of triage is required – we need to identify the most vulnerable system. Our first task is therefore is to scan our network to see what potential targets are available.

NTOP, NMAP and other tools

NTOP is a great web based tool for monitoring traffic on the network, and

```
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on em0, link-type EN10MB (Ethernet), capture size 65535 bytes
20:00:35.019848 IP 192.168.0.132.40625 > hacker.merville.intranet.ssh: Flags [.], ack 974678183, wir
20:00:35.021509 IP hacker.merville.intranet.ssh > 192.168.0.132.40625: Flags [P.], seq 4294967185:1,
20:00:35.023281 IP 192.168.0.132.40625 > hacker.merville.intranet.ssh: Flags [.], ack 113, win 5951,
20:00:35.023393 IP hacker.merville.intranet.ssh > 192.168.0.132.40625: Flags [P.], seq 1:113, ack 0,
20:00:36.025242 IP hacker.merville.intranet.17779 > border.domain: 26732+ PTR? 132.0.168.192.in-addr
20:00:36.025643 IP border.domain > hacker.merville.intranet.17779: 26732 NXDomain* 0/1/0 (109)
20:00:36.026828 IP 192.168.0.132.40625 > hacker.merville.intranet.ssh: Flags [.], ack 833, win 5921,
20:00:36.026853 IP hacker.merville.intranet.ssh > 192.168.0.132.40625: Flags [P.], seq 113:833, ack
20:00:36.681247 STP 802.1d, Config, Flags [none], bridge-id 8001.00:0d:65:1c:1a:c0.8002, length 43
20:00:37.024585 IP hacker.merville.intranet.17773 > border.domain: 26733+ PTR? 254.0.168.192.in-addr
20:00:37.025058 IP border.domain > hacker.merville.intranet.17773: 26733* 2/1/1 PTR border., PTR bor
```

Figure 5. *TCPDUMP showing network traffic in real time*

the command line equivalent TCPDUMP. in combination with GREP is a good combination for quickly identifying packets on the wire. But what of that forgotten server that nobody uses? Unless it is requesting some service (e.g. ARP or DHCP etc.) it may well remain silent when the probe takes place and remain undetected. NMAP on the other hand can actively scan the whole network so there is little chance anything will remain hidden. NTOP will perform network asset discovery as well, and will provide traffic analysis beyond what NMAP is designed for. We will use all 3 methods to discover what devices are on our network. WIRESHARK is useful as well, especially if we want to examine what is on the wire, but without active (intrusive) tools such as NMAP, the hackers task would be much more convoluted. Whereas NTOP, WIRESHARK and TCPDUMP are passive out of the box, the best results are achieved through using a combination of both sniffing (passive) and probing (intrusive) strategies.

Preparing hacker.merville.intranet (Attack server)

As root, install NMAP, NMAPSI4 and NTOP via the package system:

```
pkg_add -r nmap nmapsi4 ntop
```

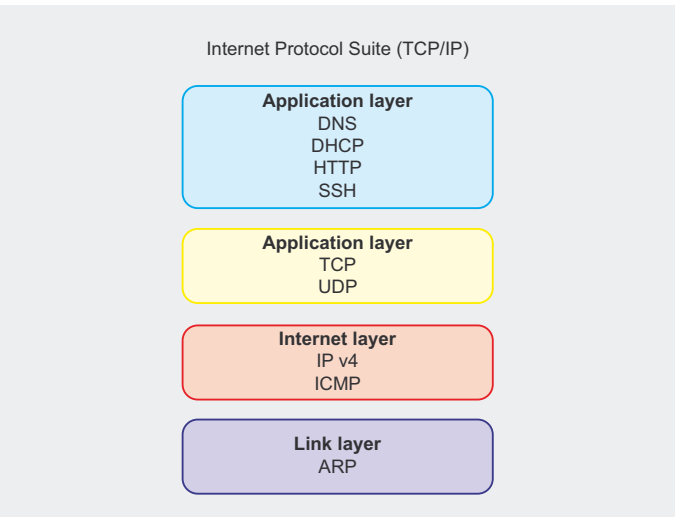


Figure 6. *ARP – from http://en.wikipedia.org/wiki/Address_Resolution_Protocol*

```
20:35:42.333952 ARP, Request who-has border (Broadcast) tell 192.168.0.141, length 46
20:35:43.332698 ARP, Request who-has 192.168.0.141 (Broadcast) tell 192.168.0.141, length 46
20:35:44.333172 ARP, Request who-has 192.168.0.141 (Broadcast) tell 192.168.0.141, length 46
20:35:45.334734 ARP, Request who-has 192.168.0.141 tell 192.168.0.141, length 46
20:35:47.794886 ARP, Request who-has border tell 192.168.0.132, length 46
20:35:47.794991 ARP, Reply border is-at 00:11:d8:0c:bd:8f (oui Unknown), length 46
20:36:14.906547 ARP, Request who-has 192.168.0.132 tell hacker.merville.intranet, length 28
20:36:14.908076 ARP, Reply 192.168.0.132 is-at 00:16:e6:58:a0:75 (oui Unknown), length 46
20:37:50.323320 ARP, Request who-has border tell 192.168.0.132, length 46
20:37:50.323430 ARP, Reply border is-at 00:11:d8:0c:bd:8f (oui Unknown), length 46
```

Figure 7. *TCPDUMP ARP traffic*

Add the following lines to /etc/rc.conf:

```
ntop_enable="YES"
ntop_flags="-d --use-syslog=daemon -u nobody -A -4"
```

This forces NTOP to start in IPv4 rather than IPv6 mode. Reboot:

```
reboot
```

If you experience difficulties running NTOP, see man ntop for further details.

Network and device discovery

NTOP

On restarting the attack server, point your browser to port 3000 and you should see the NTOP interface. Enable all the plugins and after a short delay you can view a web-page detailing the network traffic (Figure 2) discovered hosts (Figure 3) and Protocol history (Figure 4). Over time, the analysis will become more detailed as additional traffic flows through the network. Examining the output, we discover that most of the traffic is unicast (e.g. traffic between a single target and a single destination), that we have 3 hosts (Hacker, Intel and Border) on a single 192.168.x.x network and HTTP, Netbios, DNS and DHCP traffic is present of which HTTP is the majority. This is a good start, as we have part of the jigsaw of network and devices. But I know there is more on my LAN than this, for instance, I have routers, and I use SSH to access Border. These devices and protocols have not been revealed by NTOP – yet.

TCPDUMP

If you do not have access to the Internet to install packages, a great tool for analyzing network traffic is

```
# Nmap 5.61TEST2 scan initiated Sun Jan 15 21:45:10 2012 as: nmap -sn -oN SN_192.168.0.0.TXT 192.168.0.0/24
Nmap scan report for 192.168.0.0
Host is up (0.0035s latency).
Nmap scan report for hacker.merville.intranet (192.168.0.131)
Host is up.
Nmap scan report for 192.168.0.132
Host is up (0.00023s latency).
Nmap scan report for 192.168.0.250
Host is up (0.0065s latency).
Nmap scan report for border (192.168.0.254)
Host is up (0.00065s latency).
MAC Address: 00:11:D8:0C:BD:8F (Asustek Computer)
Nmap scan report for 192.168.0.255
Host is up (0.0025s latency).
# Nmap done at Sun Jan 15 21:50:32 2012 -- 256 IP addresses (6 hosts up) scanned in 321.43 seconds
```

Figure 8. *NMAP network scan results*

TCPDUMP. Used in conjunction with GREP and TELNET. On the attack machine, run the following command (You may need to throw an IFCONFIG to identify your network card type):

```
tcpdump -i em0
```

You will be presented with a real time view of the traffic passing by the ethernet device `em0` on the attack machine (Figure 5). Depending on the size of the network you are probing, the amount of traffic displayed may be impossible to read, so either filter the output through MORE (`tcpdump -i em0|more`) or GREP for the specific IP address, protocol, or as in the example below, search for *Address Resolution Packets* (ARP):

```
tcpdump -i em0 | grep ARP
```

This will result in the output similar to *Figure 7 – TCPDUMP ARP TRAFFIC*. Press Ctrl C to kill the display.

This qualifies the words of wisdom carried on the Backtrack Linux website *The quieter you become the more you are able to hear*. One of the key techniques of the hacker is stealth – hammering away at a network will quickly raise the suspicion of the firewall and any *Intrusion Detection Software* (IDS). Patience and just sitting listening over an extended period of time is a far better strategy, especially if you are cloaked or have taken measures to obfuscate your presence.

Table 2. *Potential Targets*

No	Hostname	IP Address	Discovered by
1	Hacker	192.168.0.131	NTOP, TCPDUMP, NMAP
2	Intel	192.168.0.132	NTOP, TCPDUMP, NMAP
3	?	192.168.0.141	TCPDUMP
4	?	192.168.0.250	NMAP
5	Border	192.168.0.254	NTOP, TCPDUMP, NMAP

TCPDUMP has now discovered our first anomaly, the device at 192.168.0.141 doesn't know who or where it is on the network! More interesting still, there was no indication of 192.168.0.141 on the network when we initially ran NTOP. This can easily be explained by the nature of ARP, which works at the link layer (Figure 6 – *ARP*). ARP is used to translate between IP and MAC addresses, and is often cached locally on a device. The gratuitous arp packet from 192.168.0.141 would suggest some form of router, as this method is used to force a refresh of the local ARP cache on network hosts.

To see what your arp cache contains run:

```
arp -a
```

You will see something similar to Figure 9.

```
? (192.168.0.141) at 00:16:b6:3a:18:a2 on em0 expires in 1087 seconds [ethernet]
hacker.merville.intranet (192.168.0.131) at 08:00:27:20:aa:9c on em0 permanent [ethernet]
? (192.168.0.132) at 00:16:e6:58:a0:75 on em0 expires in 453 seconds [ethernet]
border (192.168.0.254) at 00:11:d8:0c:bd:8f on em0 expires in 454 seconds [ethernet]
```

Figure 9. *ARP Cache*

```
Starting Nmap 5.61TEST2 ( http://nmap.org ) at 2012-01-15 22:48 GMT
Nmap scan report for 192.168.0.141
Host is up.
All 1000 scanned ports on 192.168.0.141 are filtered

Nmap done: 1 IP address (1 host up) scanned in 203.33 seconds
```

Figure 10. *192.168.0.141 stealth scan*

```
Starting Nmap 5.61TEST2 ( http://nmap.org ) at 2012-01-15 22:53 GMT
sendto in send_ip_packet_sd: sendto(4, packet, 44, 0, 192.168.0.140, 16) => Host is down
Offending packet: TCP 192.168.0.131:38559 > 192.168.0.140:110 S ttl=45 id=50372 iplen=11264  seq=623347669 win=1024 <mss 1460>
Sleeping 15 seconds then retrying
```

Figure 11. *192.168.0.140 – Nothing is there*

```
Nmap scan report for border.merville.intranet (192.168.0.254)
Host is up (0.00040s latency).
Not shown: 992 closed ports
PORT     STATE SERVICE
25/tcp   open  smtp
53/tcp   open  domain
110/tcp  open  pop3
139/tcp  open  netbios-ssn
143/tcp  open  imap
587/tcp  open  submission
631/tcp  open  ipp
3128/tcp open  squid-http
MAC Address: 00:11:D8:0C:BD:8F (Asustek Computer)

Nmap done: 1 IP address (1 host up) scanned in 265.18 seconds
```

Figure 12. *192.168.0.254 – Lots of potential here*

NMAP

NMAP comes in two flavors, a command line utility and a GUI version which will run under any X11 window

manager. For this exercise, we will use the CLI version as we want to capture the output to a file. As root, run the following in a directory that has sufficient disk space (root generally has a very small partition size):

```
nmap -sn 192.168.0.0/24 -oN SN_192.168.0.0.TXT
```

This will generate a text file `SN_192.168.0.0.TXT` which contains a list of discovered hosts in the range 192.168.0.0 – 192.168.0.255 (Figure 8). NMAP has many scan modes from brute force to stealth, and the scan we are using is SN – a simple ping scan without port discovery. This should flush out most of the devices on our network provided they will respond to an ICMP ping request.

Using our combination of NTOP, TCPDUMP and NMAP we now know of the following potential victims on our network (Table 2 potential targets).

Let us patiently see if we can detect what ports are open on 192.168.0.141 and 192.168.0.254 by running NMAP -Pn. This treat all hosts as online, and will fail if the device really doesn't exist. This scan will take some time. We can now tell that 192.168.0.141 really does exist but is filtering all of its ports (Figure 10). If the device did not exist, we would get a different response (Figure 11). Figure 12 shows what a 5 minute scan of 192.168.0.254 will reveal in the way of open TCP ports, Figure 13 shows the accumulated stats that NTOP managed to gather over a 4 hour 40 minute period,

Finally, going back to command line tools let us not forget the humble TENET client. Looking at Figure 12, we can see that SQUID is running on port 3128 of 192.168.0.254. By typing in some garbage, we can get the daemon to reveal the version number, SQUID2.5STABLE14 (Figure 14):

```
telnet 192.168.0.254 3128
[Press ENTER and type some garbage]
```

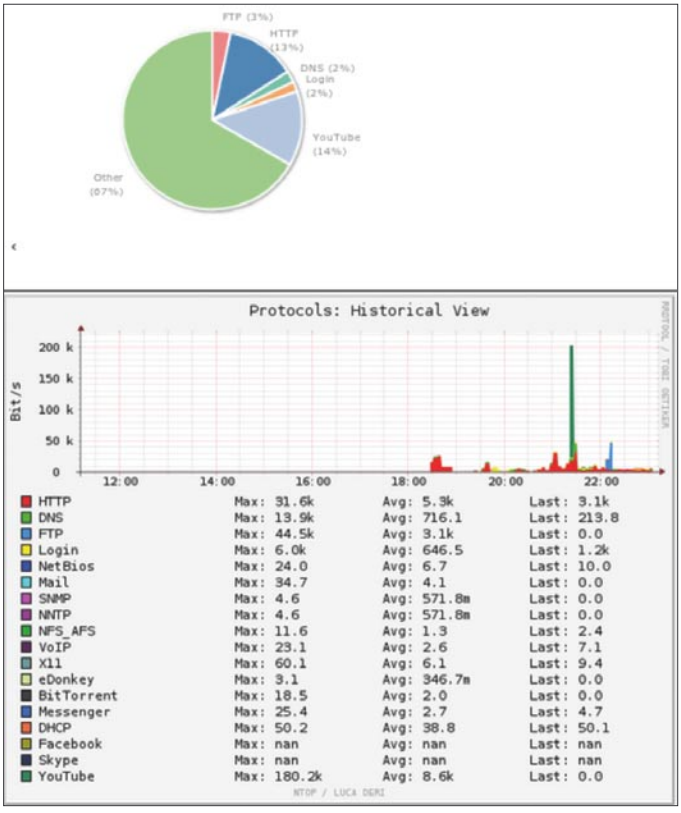


Figure 13. *NTOP updated after a few hours*

Figure 13. *Using TELNET to check open ports*

Conclusion

There are many ways of extracting valuable information from a network. We can listen over an extended period, scan devices using both brute force and stealth methods, and use tools like telnet to get daemons to reveal their identity.

ROB SOMERVILLE

Rob Somerville has been passionate about technology since his early teens. A keen advocate of open systems since the mid eighties, he has worked in many corporate sectors including finance, automotive, airlines, government and media in a variety of roles from technical support, system administrator, developer, systems integrator and IT manager. He has moved on from CP/M and nixie tubes but keeps a soldering iron handy just in case.

Counting Our Losses

2011, a hefty year as they all say. Even Discovery Channel has specials on the events of this year. The devastating earthquake in Japan, the war on terrorism finally claimed their hard sought-after victim and even the untimely death of Steve Jobs has a special. But what about the real heroes? The heroes behind the screens, outside of popular media?

Kenneth Harry Olsen co-founded DEC, Digital Equipment Corporation. DEC is best known for the PDP computer series, as well as the VAX and Alpha processors and accompanying machines. DEC also made a lot of bad decisions which eventually led them out of the control of computers in general. Next to computers, DEC was about innovation as well. Ethernet came, for a large part from DEC. Clustering machines together as one logical huge computer, or RAID 0 and 1 comparisons but using computers instead of disks, is DEC's invention. RSX, RT-11, Ultrix, VAX or OpenVMS, OSF/1, Digital UNIX and Tru64 were DEC's Operating Systems. The DLT tape standard also originated at DEC. VT100 has a recognizable ring to it, as well as VT220 maybe? Also DEC. Project Athena at MIT was largely funded by DEC, which propelled products like the X Window System, Kerberos, Zephyr which was the first instant messaging system and distributed computing, just to name a few. Apparently the first MP3 player was the so-called Personal Jukebox, which started at the DEC Systems Research Center during the merger with Compaq.

So a large portion of the internet almost raged at the low level of media attention the demise of 70 year old *Dennis MacAlistair Ritchie* on October 12th of 2011 got, compared to that of Steve Jobs. Ritchie is one of the greater pioneers of the computing age. Steve Jobs probably couldn't have achieved what he had, without UNIX and thus Dennis Ritchie. Then again, Dennis probably wouldn't have written the C programming language for UNIX, because there might not have been a DEC PDP-7 lying in the corner at Bell Labs for Ken Thompson to pick up and write UNIX on, if it wasn't for Ken Olson and his Digital Equipment Corporation.

Still UNIX and the C language had and still has a very pronounced presence in and on our current computing platforms. If you want close CPU programming for speed and accuracy, and don't want to fiddle with assembly, C is the way to go. And, as we all know, if you want something done the right way, you choose a UNIX variant. I think this is a very good ode to Dennis and his incredible work.

```
{
 printf("goodbye, dad.\n";
 return 0;
}
```

But *Robert Morris* made sure we all could login to a UNIX machine in a relatively secure manner. He wrote *crypt*, the *bc* programming language and the math library within UNIX. A cryptographer at heart, he probably has done a lot of stuff, which we will never learn. Next to a career at Bell Labs, he switched to the NSA for which you can read what led up to that switch, in the story from Dennis Ritchie *Dabbling in the Cryptographic World.* Robert's son, Robert Tappan wrote the infamous *Morris Worm* which almost destroyed 10 percent of the back then small Internet, consisting of about 60000 connected systems. I guess computer security ran in the family. Robert died 78 years of age on the 26th of June, 2011.

Steve Jobs of course is a person whom will be missed, for sure. He was a weird man, but also a true visionary in my book. He was responsible for the Apple Computer of course, but also Pixar and NeXT. This article isn't big enough to describe his life in short, there's been enough in the magazines, papers and even books about him to read up on his crazy life. He died 56 years of age on the 5th of October, 2011.

Jack, or *Jacob E., Goldman* died at the age of 90 on December the 20th, 2011. A true scientist, he did a lot of work for the Ford Scientific Lab. He was working on an electric car with a sodium-sulfur battery in the 1960s. But his main tribute to our world, was to hire Dr. George Pake to create Xerox' Palo Alto Research Centre which went on to house companies and labs, and create stuff like laser printing, Ethernet, the GUI (which Jobs *stole* for Apple) and object-oriented programming to name a few. Maybe not basis-like as the guys mentioned before, but nevertheless technologies we now use daily and take for granted.

Another Great Man which passed away this year is *John McCarthy*. He lived to see the age of 84 before passing away on October the 24th, 2011. This man thought up and nearly invented what we know as A.I. or Artificial Intelligence. He created the Lisp machine which most programmers will have seen, touched or loved at one point or another in their lives. He has done lots more, mostly anticipating future technologies way before they became reality as the one we live in today. Start off by reading into this marvellously interesting person on his Wikipedia page.

Paul Baran is credited with the invention of packet switching which is used in our networks and the internet today. While working on a report for the RAND Corporation, he had to come up with a network of some kind which could stand a nuclear attack. He used redundancy back in the early 60s to make that happen, which ended up through a series of distillations of the report into the packet switching basics which went on to be used in DARPA (The very beginning of the Internet). Later on he used the same principles to make new standards and products like ATM (*Asynch Transfer Mode*) and the discrete multi-tone modem which, in turn, is the basic for Orthogonal frequency-division muxing which is used in DSL modems. Paul passed away at the age of 84 on March the 26th of 2011.

On the 23rd of March, 2011, *Jean Jennings Bartik*, one of the six original ENIAC programmers, passed away. She also worked on the BINAC and the even more known UNIVAC 1 computer. Back in those days it was more of a woman's job to program computers. Now look at the state of it today. She didn't receive that much respect later on in her life being laid off out of her career in the computer industry, by McGraw-Hill in 1985 when she was 61 years of age. A museum in her name is at the Northwest Missouri State University in Maryville, Missouri featuring ENIAC, BINAC and UNIVAC exhibits. She's listed on the 'Women in Technology International Hall of Fame and was one of the three Fellow Award honorees 2008, along Bob Metcalfe and Linus Torvalds.

Jack Wolf was responsible for a lot of theory from the 60s till around 2000 and has won a large amount of awards for those theories. You could wade through all those papers he published, but it simply comes down to all the problems they had when trying to get data from one point to the other without loosing anything. It still is used in everything we use today, like the hard disk drives, tape drives and more from the last 20 years or so. He was 76 years of age when he passed on May 12, 2011.

Then on April the 21st, marked the end of *Max Mathews'* life. Even less known than the guys listed above, this man might have been the one which made it all possible for us to have complete orchestras inside our computers today. No later than 1957 Max had an IBM up running at Bell Labs, serving up 17 seconds of composition, purely made and synthesized on a computer! He was 84 years of age.

Anthony Edgar 'Tony' Sale was a British electronic engineer and loved computer history. He was the man who built George the robot out of Meccano back in 1949, which was a big thing back then. He restarted George in 2011 after being stored in Tony's garage, with some oil in the bearings, new lithium batteries and George was alive again. Tony was a member of the British Computer Society, the Computer Conservation Society and a big man to Bletchley Park, where he also built the famous Colossus computer replica which can still decipher encrypted messages. He died at the age of 80 on August 30, 2011.

Tom West is probably best described by reading the book *The Soul of a New Machine*, which was written by Tracy Kidder, but Tom was the source of information for the book. Tom also created the Eclipse MV/8000, a 32-bits computer in the late 70s, which moved Data General up to the level of the big players. He seemed a very interesting man, of which not much is found. The piece written up at The Boston (*http://www.boston.com/bostonglobe/obituaries/articles/2011/05/22/tom_west_engineer_was_the_soul_of_data_generals_new_machine/?page=full*) seems to just lift the tip of the blanket of Tom's life. He died on the 19th of May, 71 years of age.

Michael Stern Hart might not have been a true engineer at heart, but he was responsible for two things at least, which are pretty important for us today. The electronic book, or e-book and Project Gutenberg. He founded the Project and with that paved the way for a lot of electronic available books and texts. He was originally an author and his own books are all available from the Project. He had a knack for writing monospaced email messages of which each line was exactly the same length, pretty cool. He lived to experience the age of 64 when he passed on the 6th of September, 2011.

May they all rest in peace.

SANDER REICHE

Sander Reiche is a PDP-11 fanatic and BSD/UNIX lover in his spare time, and a UNIX Systems Engineer on his day-job. Founder of the Veritable UNIX Systems Group. His web page is located at http://ls-al.eu/~reiche.

In the next issue:

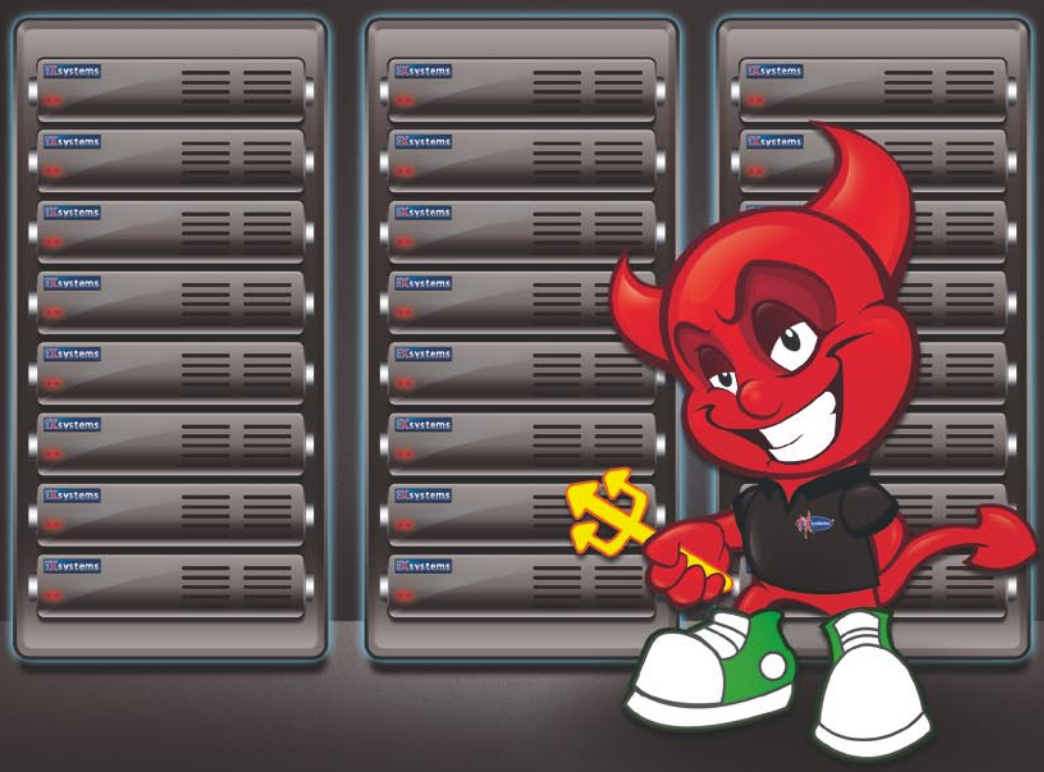
Continuation of series:

- **BSD Certification by Dru**
- **Anatomy of FreeBSD Compromise**
- **More about PostgreSQL**
- **and Other !**

Next issue is coming in March!


Looking for help, tip or advice?
Want to share your knowledge with others?
BSD MAGAZINE
Give us your opinion about the magazine's content
and help us create the most useful source for you!

What has your server vendor done for BSD lately? Probably, not much.



Work with a vendor that supports the operating system you love!

iX is the corporate sponsor of the PC-BSD® Project, a major corporate donor to the FreeBSD Foundation, and leads the FreeNAS™ development team -- all while employing some of the most brilliant minds in the FreeBSD® community. For BSD hardware and software expertise, look no further.

1-855-GREP-4-IX

http://www.iXsystems.com/community

iXsystems®