MAGAZINE

BSD

FOR NOVICE AND ADVANCED USERS

HAST ON FREEBSD

HOW TO MAKE STORAGE HIGHLY AVAILABLE BY USING HAST

HOW SECURE CAN SECURE SHELL BE?

PFSENSE + SNORT

MAXIMISING WEBSITE RUNTIME
ON HOST SERVERS RUNNING FREEBSD

HOW TO APPLY A STYLE USING
CASCADING STYLE SHEETS

HOW TO IMPROVE THE LOGIN PROCESS
& ADD MORE SECURITY

VOL.7 NO.11
ISSUE 11/2013(52)
1898-9144

E5-2600

High Performance, High Density Servers for Data Center, Virtualization, & HPC

MODEL: iXR-22X4IB

http://www.iXsystems.com/e5

768GB of RAM in 1U

KEY FEATURES

iXR-22X4IB

- Dual Intel® Xeon® Processors E5-2600 Family per node
- Intel® C600 series chipset
- Four server nodes in 2U of rack space
- Up to 256GB main memory per server node
- One Mellanox® ConnectX QDR 40Gbp/s Infiniband w/QSFP Connector per node
- 12 SAS/SATA drive bays, 3 per node
- Hardware RAID via LSI2108 controller
- Shared 1620W redundant high-efficiency Platinum level (91%+) power supplies

iXR-1204+10G

- Dual Intel® Xeon® Processors E5-2600 Family
- Intel® C600 series chipset
- Intel® X540 Dual-Port 10 Gigabit Ethernet Controllers
- Up to 16 Cores and 32 process threads
- Up to 768GB main memory
- Four SAS/SATA drive bays
- Onboard SATA RAID 0, 1, 5, and 10
- 700W high-efficiency redundant power supply with FC and PMBus (80%+ Gold Certified)

Call iXsystems toll free or visit our website today! **1-855-GREP-4-IX | www.iXsystems.com**

High-Density iXsystems Servers powered by the Intel® Xeon® Processor E5-2600 Family and Intel® C600 series chipset can pack up to 768GB of RAM into 1U of rack space or up to 8 processors - with up to 128 threads - in 2U.

On-board 10 Gigabit Ethernet and Infiniband for Greater Throughput in less Rack Space.

Servers from iXsystems based on the Intel® Xeon® Processor E5-2600 Family feature high-throughput connections on the motherboard, saving critical expansion space. The Intel® C600 Series chipset supports up to 384GB of RAM per processor, allowing performance in a single server to reach new heights. This ensures that you're not paying for more than you need to achieve the performance you want.

The iXR-1204 +10G features dual onboard 10GigE + dual onboard 1GigE network controllers, up to 768GB of RAM and dual Intel® Xeon® Processors E5-2600 Family, freeing up critical expansion card space for application-specific hardware. The uncompromised performance and flexibility of the iXR-1204 +10G makes it suitable for clustering, high-traffic webservers, virtualization, and cloud computing applications - anywhere you need the most resources available.

For even greater performance density, the iXR-22X4IB squeezes four server nodes into two units of rack space, each with dual Intel® Xeon® Processors E5-2600 Family, up to 256GB of RAM, and an on-board Mellanox® ConnectX QDR 40Gbp/s Infiniband w/QSFP Connector. The iXR-22X4IB is perfect for high-powered computing, virtualization, or business intelligence applications that require the computing power of the Intel® Xeon® Processor E5-2600 Family and the high throughput of Infiniband.

HIGH Throughput & INCREDIBLE Performance Density

IXR-1204+10G: **10GbE On-Board**

4 Server Nodes in 2U

IXR-22X4IB

Call iXsystems toll free or visit our website today! **1-855-GREP-4-IX | www.iXsystems.com**

Dear Readers,

2013 is nearly over. Some of you are new to our magazine and some are with us for months, even years. The end of the year approaches rapidly and we decided that in exchange for your unmeasurable support, we will tell you our story that should give you an insight into what we have been through this year.

Let's discuss the statistics. This should help you visualize our work with the magazine and understand the process we have to undergo in order to meet your expectations.

This year, we have published 12 BSD issues – around 600 pages. 600 pages equals 35,68 m2 that our articles could cover. All the issues published in 2013, when put on the scale, would weight 34,28 pounds (12,79 kilos). Throughout the year, our readership base escalated 2 times – from slightly over 21459 to 49890 readers.

As you know, we are the number 1 BSD publication in the world. We would like to thank iXsystems company and all the team who has supported us from the very beginning. I would like to thank Denise Ebery and Annie A. Zhang for their patience, professionalism and a great work on all issues of BSD magazine.

In order to give you the materials you had a chance to read this year, we were working over 250 days, which equals more than 2000 hours for each employee. These few pages you go through in a couple of hours on a monthly basis, cost our experts almost 1000 weeks to prepare. Our beta testers and proofreaders have spent a similar amount of time making sure that you will enjoy your reading. Finally, our graphic devoted 3000 hours designing the layout to appeal to your eyes.

During our fight for your right to admin better, we have also suffered losses. As you may guess, our main weapon is the computer. Just like in every war, the equipment is exploited heavily and put through extreme situations. You may be sure that we have pushed our PCs to their absolute limits. We have overheated our processors, filled the hard drives, overused internet connection transfers, etc. Most of our inventory have survived, although we cannot deny there were casualties – 10 computer mice have passed away during the harsh battles for knowledge.

However, As long as we have our precious readers, we have a purpose. We owe you a huge THANK YOU. Everything we do, we do with you on our minds. We are grateful for every comment and opinion, either positive or negative. Every word from you lets us improve BSD magazine and brings us closer to the ideal shape of our publication, or, we should say – your publication.

Thank you BSD fans for your invaluable support and contribution.
Ewa & BSD team

MAGAZINE BSD

Editor in Chief:
Ewa Dudzic
ewa.dudzic@software.com.pl

Contributing:
Michael Shirk, Andrey Vedikhin, Petr Topiarz, Charles Rapenne, Anton Borisov, Jeroen van Nieuwenhuizen, José B. Alós, Luke Marsden, Salih Khan, Arkadiusz Majewski, BEng

Top Betatesters & Proofreaders:
Annie Zhang, Denise Ebery, Eric Geissinger, Luca Ferrari, Imad Soltani, Olaoluwa Omokanwaye, Radjis Mahangoe, Mani Kanth, Ben Milman

Special Thanks:
Annie Zhang
Denise Ebery

Art Director:
Ireneusz Pogroszewski

DTP:
Ireneusz Pogroszewski
ireneusz.pogroszewski@software.com.pl

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

Publisher:
Hakin9 Media SK
02-676 Warsaw, Poland
Postepu 17D
Poland
worldwide publishing
editors@bsdmag.org
www.bsdmag.org

Hakin9 Media 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.

Storage

NetBSD 6.0

Security

Column

Reports

Configuring a Highly Available Service on FreeBSD – Part 1: HAST

One of the problems a system administrator has when providing services like NFS on a network is that sometimes they are critical to the business and downtime needs to be kept to an absolute minimum. This problem can be solved with tools native to FreeBSD.

What you will learn…

- How to configure HAST
- How to control HAST
- How to recover from HAST failures

What you should know…

- How to install FreeBSD
- How to login to FreeBSD
- How to edit files on FreeBSD

In this series of articles, we will introduce and learn how to use these building blocks to make a service and the underlying storage highly available. As an example we will build a highly available NFS server running on two FreeBSD 9.2 machines called nfs-01 and nfs-02. The underlying principles can be applied to other services as well.

In this first part of the series we will learn how to make storage highly available by using HAST. We will take a look at what HAST is, how to configure it, how to control it and how to recover from failures like a splitbrain situation.

What is HAST?

HAST stands for Highly Available Storage. The main component of HAST is the hastd daemon, which allows the user to transparently store data on two physically separated machines which are connected over TCP/IP. HAST supports both the IPv4 and IPv6 connections. The creation of these connections is always initiated by the primary node. In this active/passive setup, the redundant storage can be accessed only on the active node where a disk-like device is presented under `/dev/hast/<resourcename>`. This `<resourcename>` is a GEOM provider. An important thing to know is that HAST does not configure or change the active (primary) or passive (secondary) role by itself. To automate role switching other tools, like for example CARP, have to be configured to handle the failover.

How to configure HAST

The main configuration of the hastd daemon is done in the `/etc/hast.conf` file. This file can consist of a global section, a node specific section and a resource specific section. Let's explore the basic configuration for our setup as described in Listing 1.

In the global section we see the *timeout 20* line, which sets the default timeout for the connection between the hastd daemon on the nodes. This global timeout could be overridden in the node specific and resource specific sections if we wanted to.

If we look at the node specific section for nfs-01 (see Listing 2), we first note the *on nfs-01 {* line. Which specifies that this part is valid for the machine called nfs-01. One advantage of using this construction is that it is possible to use the same configuration file on all nodes, because a node will only pick up the global parts and the

parts for itself. It will thus ignore the on `<othernodename>` parts of the configuration file.

The second line of the node specific section for nfs-01 says:

```
pidfile /var/run/hastd.pid
```

Which indicates that the pidfile used by hastd on nfs-01 should be placed in /`var/run/hastd.pid` (which is the default). The last line closes the node specific section.

Let us now continue by looking at the resource specific section (Listing 3). We first see the *resource* keyword followed by the name of the resource *sharedbynfs.* This means that the resource is called *sharedbynfs* and will become available under `/dev/hast/sharedbynfs` on the primary node when hastd has been started and initiated.

If we look at the node specific part of the resource section, we see two configuration options for every node. First the *local* directive which specifies the local device used on this node to use as a backing device for hast. In this example we will use the `/dev/da1` disk. The second line (*remote*) specifies the name of the other node. So for nfs-01 it is nfs-02 and for nfs-02 it is nfs-01.

Of course there are more options than specified in the configuration above. You can find the description of them in the hast.conf manual page (man `hast.conf`).

Starting hastd and controlling HAST

Now that the configuration is in place, we have to initialise our resource on both nodes with the `hastctl` command (Listing 4).

This initialisation creates the metadata that hast needs to be able to determine which data still needs to be synchronised between the nodes.

Now that we have our metadata initialised we can start using hast. To start hast, we have to add the line

Listing 1. *Our hast.conf*

```
timeout 20

on nfs-01 {
    pidfile /var/run/hastd.pid
}
on nfs-02 {
    pidfile /var/run/hastd.pid
}

resource sharedbynfs {
    on nfs-01 {
         local /dev/da1
         remote nfs-02
    }
    on nfs-02 {
         local /dev/da1
         remote nfs-01
    }
}
```

Listing 2. *Node nfs-01 specific section from hast.conf*

```
on nfs-01 {
    pidfile /var/run/hastd.pid
}
```

Listing 3. *Resource specific section from hast.conf*

```
resource sharedbynfs {
    on nfs-01 {
         local /dev/da1
         remote nfs-02
    }
    on nfs-02 {
         local /dev/da1
         remote nfs-01
    }
}
```

Listing 4. *Initializing our resource*

```
hastctl create sharedbynfs
```

Listing 5. *Starting hastd*

```
nfs-01# echo 'hastd_enable="YES"' >> /etc/rc.conf
nfs-01# service hastd start
nfs-01# hastctl role primary sharedbynfs
nfs-01# hastctl status

nfs-02# echo 'hastd_enable="YES"' >> /etc/rc.conf
nfs-02# service hastd start
nfs-02# hastctl role secondary sharedbynfs
nfs-02# hastctl status
```

Listing 6. *Putting a filesystem on the sharedbynfs hast resource*

```
nfs-01# newfs -U /dev/hast/sharedbynfs
nfs-01# mkdir /export
nfs-01# mount -o noatime /dev/hast/sharedbynfs /export
```

`hastd_enable="YES" to /etc/rc.conf`. Then we have to start hastd and set a role on each node (see listing 5 for the commands). In this example we make nfs-01 the primary node and nfs-02 the secondary node. Also note the use of the *hastctl status* command to check the current status of our hast configuration.

Creating a filesystem

Now that we have a working hast setup, it is time to put a filesystem (*newfs)* on it and make that filesystem available under /export on the primary node (nfs-01) with the mount command. The exact commands to do this are described in Listing 6. Please note that we are using the *noatime* mount option to reduce the number of I/O requests, which in turn reduces the number of synchronisation actions that hastd has to execute.

```
nfs-01# newfs -U /dev/hast/sharedbynfs
nfs-01# mkdir /export
nfs-01# mount -o noatime /dev/hast/sharedbynfs /export
```

Failover

Of course it is nice to have a setup like this, but to be able to put it to good use we must know how to do a manual failover. Assuming both nodes are still up and running this is relatively straight forward. We use our example setup with nfs-01 and nfs-02 to move the primary node from nfs-01 to nfs-02. First we umount the filesystem on nfs-01 and mark nfs-01 as secondary. When nfs-01 has become a secondary node, we can make nfs-02 the primary node, check the filesystem and mount the filesystem on nfs-02 (See listing 7 for the exact commands). It is a good practice to always check the filesystem after a failover, but before mounting. The reason for this is that in case of a failover due to a failing node, we can not be sure that every bit of data has been synchronised to the other side. This means that we can not be sure that the filesystem is in a clean and consistent state.

Listing 7. *Failover from nfs-01 to nfs-02*

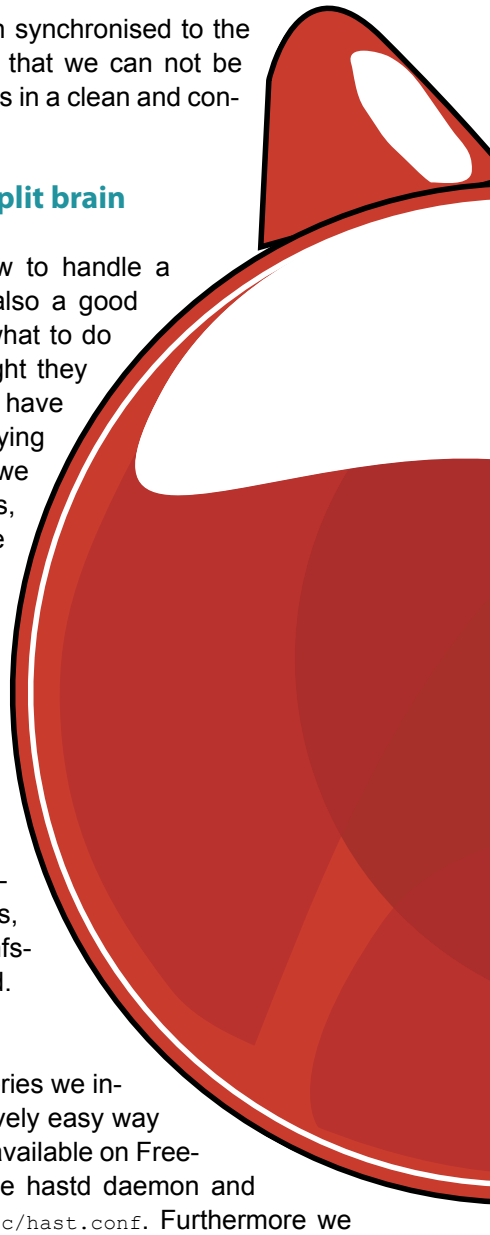
```
nfs-01# umount /export
nfs-01# hastctl role secondary sharedbynfs

nfs-02# hastctl role primary sharedbynfs
nfs-02# fsck -t ufs /dev/hast/sharedbynfs
nfs-02# mount -o noatime /dev/hast/sharedbynfs /export
```

Listing 8. *Recovering from a splitbrain situation*

```
nfs-02# hastctl role init sharedbynfs
nfs-02# hastctl create sharedbynfs
nfs-02# hastctl role secondary sharedbynfs
```

Recovering from a split brain situation

Now that we know how to handle a failover situation, it is also a good idea to take a look at what to do when both nodes thought they were the primary and have written to the underlying storage. In this case we can not avoid data loss, so a decision has to be made which node will resynchronise its data from the other node. That node will have to be disconnected, reinitialised and put in the secondary role after which full data synchronisation will take place. See Listing 8 for the exact commands to do this, where we assume that nfs-02 has to be reinitialised.



Conclusion

In this first part of the series we introduced HAST, a relatively easy way to make storage highly available on FreeBSD. We introduced the hastd daemon and its configuration file `/etc/hast.conf`. Furthermore we learned how to control HAST with the hastctl command and how to recover from a splitbrain situation. Now that we have configured HAST and therefore have created a highly available storage pool for our service, we will learn how to automate failover with CARP and devd in the next part of this series.

JEROEN VAN NIEUWENHUIZEN

Jeroen van Nieuwenhuizen works as a unix consultant for Snow. His free time activities beside playing with FreeBSD include cycling, chess and ice skating.


Are you in RED?
Meet iBLISS and turn into blue.
Professional services and solutions - Imperva, McAfee, HP, Tenable.
Penetration tests, Application Security, Managed Security Services (MSS).
Do as largest companies in Brazil, contact us!
www.ibliss.com.br info@ibliss.com.br +55 11 3255-3926
iBLISS
SEGURANÇA & INTELIGÊNCIA

IT Inventory & Asset Management Automation

The main aim of this article is to provide details of implementation of an Assets Management system based on a NetBSD 6.0 running computer to handle large IT platform sites, capable of acting as a gateway to collect all information from IT programmable devices with a minimal effort to administrator and using the benefits of Open Source solutions provided by OCS Inventory and GLPI Assets management projects. Moreover, the reader can extrapolate the procedures explained hereinafter to other Unix OS flavors with no major changes to achieve his own purposes.

What you will learn…

- Deployment of a combined inventory/asset management solution
- Inventory agents configuration for Unix/MS Windows platforms
- Basic operations on IT inventory and common features
- MySQL database installation, configuration and population

What you should know…

- Basic knowledge of Apache Web Server configuration and management.
- Perl modules installation procedures
- MySQL database installation, configuration and population
- User-level background on NetBSD OS (also Unix-like OS)

The class of supported hardware and software devices are:

- Computers
- Network adapters
- BIOS/OBP PROM
- Storage/Removable Media
- Video
- Printers
- Virtual machines, including Solaris zones
- Miscellaneous Hardware

The mechanisms for monitoring supported uses cover the following operating systems:

- MS Windows
- MacOS X
- Un*x (BSD-based OS, GNU/Linux, …)
- OpenVMS by means of SNMP agent/trap daemon collector.

In reference to software inventory, the items collected are:

- Operating System
- Installed Software
- Custom-specified registry queries for MS Windows equipment

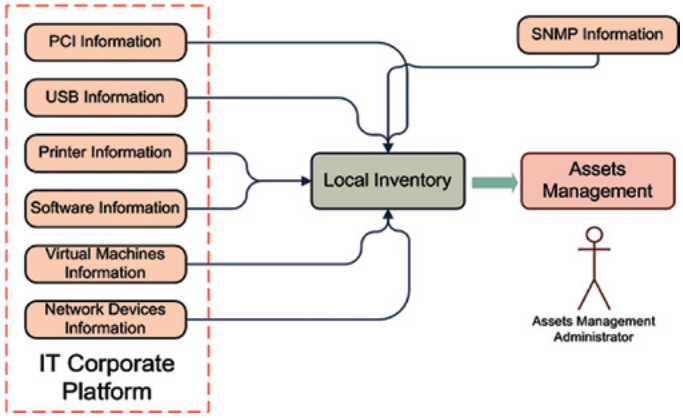


Figure 1. *Overview of IT Inventory/Asset management platform*

The Asset Management System itself, described in Figure 1, allows building up a database with an inventory for large companies to perform an accurate contract and licensing management maintenance and support, including a job-tracking system with mail-enabled alerts to provide online information. The most relevant feature of this platform is the fact that it has been developed entirely by using FOSS in order to avoid recurrent costs and to achieve long-term support, thus, the required investment is just for specialized personnel to deploy the low-level infrastructure.

IT Inventory and Asset Management

The main purpose of this document is to introduce a detailed view of a complete integrated platform to control all inventory and assets for complex organisations according with the architecture depicted in Figure 2.

- OCS is used to make the automated inventory generation for all IT hardware present in large organisations or companies as efficient as possible.
- GLPI is used for asset management and ticketing system for all IT-related items provided by OCS. In this way, there exists an interface between GLPI and OCS which allows automatic capture of data by means of an OCS Agent, available for MS Windows, Unix and MacOS platforms.
- OTRS is used as an ITIL-compliant ticketing system for all custom processes defined for large organisations or companies.

However, all OTRS functions distinct from processes definition and management can be carried out by the tandem OCS/GLPI, as happens with Help Desk services widely used in large companies to handle an automated management of hardware/software items as well as some assets associated to them.

For large IT platforms, it doesn't make sense and, besides, it has a propensity for mistakes when filling out all available data by hand. Hence, the most logical approach is to gather all available data concerning hardware and software in an automatic way. Furthermore, if we are able to develop a no-cost Open Source platform to manage inventory and related assets for IT equipment included in heterogeneous environments like Un*x, MS Windows and OpenVMS, it should be a suitable solution for future company asset management.

Inventory Management Platform

Once the architecture of our proposed inventory platform has been introduced, it is possible to distinguish three different categories of entities which are shown by Figure 3.

- Inventory Server, in charge of gathering all information provided by IT equipment to be included in inventory.
- Inventory Agents, which are each of the IT equipment to be included in inventory.
- Inventory Console, used for inventory/assets administration to get a full, updated and accurate picture of the current inventory.

Generic Asset Management Platform

The deployment of such architecture by using an IA32 computer running NetBSD 6.0 as the operating system, will eventually be followed by the installation of some processes running in the background. These processes, termed agents, will be in charge of providing gathered data about the hardware and software running on them by using a unidirectional link so that these pieces of IT equipment, which are now part of the inventory, are the items for associated assets.

OCS Server Installation Procedure

Before starting with the installation process, be aware that our recommendation is to use the NetBSD 6.0 Release on your target computer. If you try it by using the latest NetBSD 6.1.2 release, you will get into trouble due to some unavailable packages. In order to avoid further problems, we recommend the use of NetBSD 6.0 release with the following settings to properly use the pkgsrc system in order to get the necessary pack-

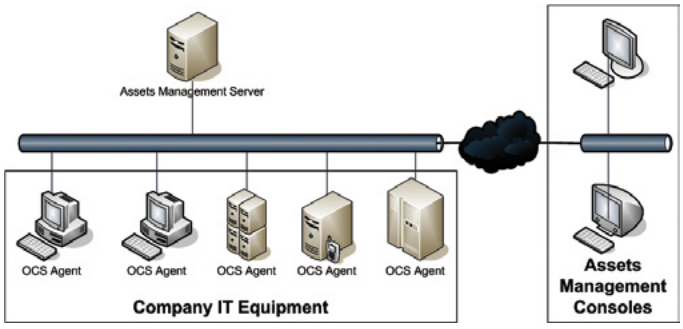


Figure 2. *Generic Asset Management Platform*

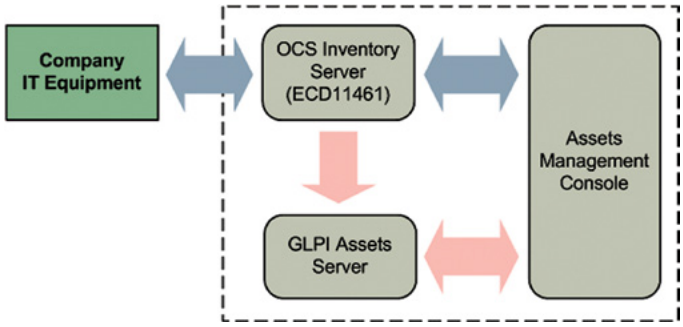


Figure 3. *Inventory and Asset management functional architecture*

ages by means of the pkg_add(1m) command. For this reason, before starting the installation process, you must add the following entries in your .profile if you are using a Bourne-like shell:

```
PKG_PATH="http://ftp.netbsd.org/pub/pkgsrc/packages/
   NetBSD/i386/6.1.2/All"
export PKG_PATH
```

Thus, the aim of this section is to provide detailed guidance on installing OCS for Inventory generation and management and GLPI for asset management with unidirectional interface between OCS and GLPI so that manual intervention is minimised.

Previous Requirements

Four points should be reviewed before starting with the installation of OCS NG Server in Unix platforms:

1. Apache 2.x Web Server running
   - Apache daemon binary [–/usr/sbin/httpd–]
   - Apache main configuration file [–/etc/httpd/httpd.conf–]
   - user account is running Apache web server [apache]
2. PHP Settings (php.ini) in –/etc/httpd/– directory:

```
post_max_size = 200M
upload_max_filesize = 200M
```

3. MySQL 5.5 running an instance as TCP service in localhost/3306.
4. Required Perl modules installed available at *www.cpan.org*.

Hence to install the binary packages required, issue the following commands:

```
# pkg_add apache-2.4.6
# pkg_add php-5.5.4
```

The same steps shall be done for the packages required to install MySQL DB Server and Client utilities, *http://www.glpi-project.org/*.

```
mysql-client-5.6.13nb1 MySQL 5, a free SQL database (client)
mysql-server-5.6.13 MySQL 5, a free SQL database (server)
```

and then check whether these two packages have been successfully installed. In the case of Apache HTTP server:

```
laertes# pkg_info apache-2.4.6
```

Information for apache-2.4.6:

Comment

Apache HTTP (Web) server, version 2.4

Requires

```
apr>=1.4.5nb3
apr-util>=1.4.1nb4
pcre>=8.30nb1
```

Description

The Apache HTTP Server Project is an effort to develop and maintain an open-source HTTP server for various modern desktop and server operating systems, such as UNIX and Windows NT. The goal of this project is to provide a secure, efficient and extensible server which provides HTTP services in sync with the current HTTP standards.

This package tracks 2.4.x release.

Homepage: *http://httpd.apache.org/*.

Install notice

```
$NetBSD: MESSAGE,v 1.1 2012/08/26 12:37:34 ryoon Exp $
```

After apache-2.4.3, `--enable-mpms-shared='event prefork worker'` is passed to configure script, then a three multi-process model is built and you can select the model in the configuration file.
The mod_cgi.so module conflicts with the non-prefork multi-process model, and mod_cgi.so module is not built anymore. You can use mod_cgid.so module instead.

Also, if you are using this procedure, remember that the main configuration file http.conf is placed in the `/usr/pkg/etc/httpd/httpd.conf` directory.

Once these packages have been installed, it is time to download a copy of the source code for both server-client addons; OCS NG Server for UNIX and GLPI, which are available at *http://www.ocsinventory-ng.org* and *http://www.glpi-project.org/*, respectively. Notice that you should download OCS Inventory NG Version 2.1RC1, released on April 2013 and GLPI Version 0.84, released in May 2013.

The suggested procedure to install OCS NG Server is summarised in the following sequential steps:

1. Run the –setup.sh– script (Listing 1).
2. Ensure that all Perl required modules are available in the future platform for OCS NG Server (Listing 2). Notice that –SOAP::Lite– is also rrequired together with the features to enable OCS Inventory NG SOAP Web Service: Listing 3.

This is undoubtedly the most difficult step, as it requires a lot of patience to get all required Perl modules from *http://www.cpan.org* site to compile and install them before continuing with the installation process. Thus, the process of building and installing a new Perl module consists of the following steps:
a) Download and uncompress Perl module tarball
b) Build and install the module in our NetBSD host:

```
# perl Makefile.PL
# make
# make test
# make install
```

3. Ensure that Apache recognizes the following configuration:

```
# OCS NG Inventory
Include /usr/pkg/etc/httpd/extra/ocsinventory-reports.conf
```

4. Point a browser to *http://localhost/ocsreports* to get the main HTML page of your Apache Web Server under NetBSD and fill up the following fields:

Listing 1. *Run the —setup.sh— script*

```
root@ecl51991:/home/c20395/OCS/OCSNG_UNIX_SERVER-2.0.5# ./setup.sh

+------------------------------------------------------------+
|                                                            |
| Welcome to OCS Inventory NG Management server setup !      |
|                                                            |
+------------------------------------------------------------+

Trying to determine which OS or Linux distribution you use
+------------------------------------------------------------+
| Checking for Apache web server binaries !                  |
+------------------------------------------------------------+

CAUTION: If upgrading Communication server from OCS Inventory NG 1.0 RC2 and
previous, please remove any Apache configuration for Communication Server!

Do you wish to continue ([y]/n)?
...

+------------------------------------------------------------+
| OK, Administration server installation finished ;-)        |
|                                                            |
| Please, review /etc/httpd/ocsinventory-reports.conf
| to ensure all is good and restart Apache daemon.           |
|                                                            |
| Then, point your browser to http://server//ocsreports
| to configure database server and create/update schema.     |
+------------------------------------------------------------+

Setup has created a log file /home/c20395/OCS/OCSNG_UNIX_SERVER-2.0.5/ocs_server_setup.log. Please, save this file.
If you encounter an error while running OCS Inventory NG Management server,
we can ask you to show us this content !

DON'T FORGET TO RESTART APACHE DAEMON !

Enjoy OCS Inventory NG ;-)
```

Listing 2. *Ensure that all Perl required modules are available in the future platform for OCS NG Serve*

```
Checking for DBI PERL module...
Found that PERL module DBI is available.
Checking for Apache::DBI PERL module...
*** ERROR: PERL module Apache::DBI is not installed !
Checking for DBD::mysql PERL module...
Found that PERL module DBD::mysql is available.
Checking for Compress::Zlib PERL module...
Found that PERL module Compress::Zlib is available.
Checking for XML::Simple PERL module...
Found that PERL module XML::Simple is available.
Checking for Net::IP PERL module...
*** ERROR: PERL module Net::IP is not installed !
*** ERROR: There are one or more required PERL modules missing on your computer !
Please, install missing PERL modules first.
```

Listing 3. *The —SOAP::Lite— features*

```
Feature                      Prerequisites               Install?
---------------------------- --------------------------- --------
Core Package                 [*] Scalar::Util            always
                             [*] URI
                             [*] constant
                             [*] Test::More
                             [*] MIME::Base64
                             [ ] Class::Inspector
                             [*] XML::Parser (v2.23)
                             [ ] Task::Weaken
Client HTTP support          [ ] LWP::UserAgent          always
Client HTTPS support         [ ] Crypt::SSLeay           [ no ]
Client SMTP/sendmail support [ ] MIME::Lite              [ no ]
Client FTP support           [ ] SOAP::Transport::FTP (v0.711) [ no ]
Client TCP support           [ ] SOAP::Transport::TCP (v0.714) [ no ]
Standalone HTTP server       [ ] HTTP::Daemon            [ no ]
Apache/mod_perl server       [ ] Apache                  [ no ]
FastCGI server               [ ] FCGI                    [ no ]
POP3 server                  [ ] MIME::Parser            [ no ]
                             [*] Net::POP3
IO server                    [*] IO::File                [ yes ]
MQ transport support         [ ] SOAP::Transport::MQ (v0.712) [ no ]
JABBER transport support     [ ] SOAP::Transport::JABBER (v0.712) [ no ]
MIME messages                [ ] MIME::Parser            [ no ]
DIME messages                [ ] IO::Scalar (v2.105)     [ no ]
                             [ ] DIME::Tools (v0.03)
                             [ ] Data::UUID (v0.11)
SSL Support for TCP Transport [ ] IO::Socket::SSL        [ no ]
Compression support for HTTP [*] Compress::Zlib          [ yes ]
MIME interoperability w/ Axis [ ] MIME::Parser (v6.106)  [ no ]
--- An asterix '[*]' indicates if the module is currently installed.
```

MySQL Login:	ocs
MySQL Password:	ocs
Name of Database:	ocsweb
MySQL Hostname:	localhost

This script creates 94 new tables in MySQL ocsweb database.

5. MySQL configuration requires to deal with at least 2 MB for `–max_allowed_packet parameter–`. So, modify the following entry in `/usr/pkg/share/mysql/my.cnf` configuration file:

```
max_allowed_packet = 2M
```

OCS Agent Installation Guide

Once the installation of OCS Server has been successfully undertaken, it is time to install OCS Agents for all IT equipment you wish to control. The role played by these agents is to provide a way to gather all items to be stored as a part of our IT inventory in an automatic way.

For our purposes, it will be sufficient to illustrate the case for Unix and MS Windows platform in order to get a working prototype. This prototype can be easily extended to a real, more complex IT platform using the procedures described hereinafter.

Hardening the OCS NG Server

In order to avoid the proliferation of permissions and redundant grantings, it is convenient to follow the sequence of steps below to strengthen security in our OCS NG server:

1. Set up permissions.

```
# chown -R root:apache /usr/share/ocsinventory-reports/
   ocsreports
# chmod -R g+w /usr/share/ocsinventory-reports/ocsreports
```

2. Create old_conf directory with write permissions for Apache user.

```
root@ecl51991:/usr/share/ocsinventory-reports/ocsreports/
   plugins/main_sections/conf/old_conf
```

3. Ensure that the user/password of MySQL in `–z-ocsinventory-server.conf–` is OK.

```
# Name of database
PerlSetEnv OCS_DB_NAME ocsweb
PerlSetEnv OCS_DB_LOCAL ocsweb
# User allowed to connect to database
PerlSetEnv OCS_DB_USER ocs
# Password for user
PerlSetVar OCS_DB_PWD ocs
```

A working prototype is installed and available at *http://ecd11461/ocsreports* for free. The access to *http://ecd11461/ocsreports* is granted by default to user 'admin'. This fact shall be kept in mind to configure OCS NG Agent for other equipment.

GLPI Installation Process

In contrast to OCS NG server, the installation of GLPI is easier to perform for Unix platforms and may be summarised in the following stages:

1. Create database. The default login/passwords once the MySQL database has been initialized are:
   - glpi/glpi for the administrator account
   - tech/tech for the technician account
   - normal/normal for the normal account
   - post-only/postonly for the postonly account

   and that is all and it is ready for use at the URL associated to OCS Inventory/Assets Server at *http://ecd11461/glpi/install/install.php*
2. Install OCS Import plugin (1.6.1) and uncompress in plugins directory. Select setup->plugins in GLPI web console.
3. Enable OCS NG mode in GLPI web console.

```
../../../glpi/files/_log/ocsng_fullsync.log
../../../glpi/files/_log/sql-errors.log
../../../glpi/files/_log/php-errors.log
../../../glpi/files/_log/cron.log
```

OCS NG / GLPI Plugin interface

Log into the GLPI web console interface at *http://ecd11461/glpi/* by using "glpi" administrator user as shown in Figure 4, and select the following options:

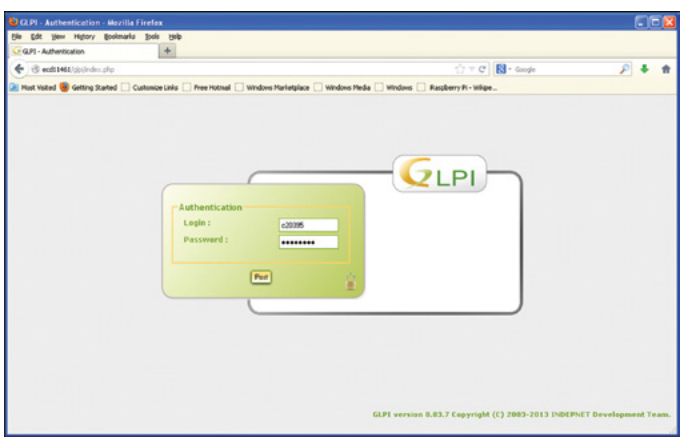


Figure 4. *Inventory and Asset management functional architecture*

1. Setup->General Setup and activate
2. Setup->OCSNG mode

and select “localhost” by setting up the following parameters:

OCSNG database	ocsweb
OCSNG database user	ocs
OCSNG database password	pass
OCSNG database in UTF8	Yes
Active	Yes

Also ensure that Web address of the OCSNG console points to *http://ecd11461/ocsreports*, which is the OCS NG Console URL address.

Ensure that Plugins -> OCS Import has been setup for localhost OCS server.

OCS Agent Installation Guide

Each candidate platform to be included in the inventory must count with a running OCS Agent to deliver copies of the items hardware and software installed periodically.

UNIX OCS NG Agent Installation

OCS Inventory Agent for UNIX is nothing else than a Perl module whose compilation and linking process is already known by professionals. The point is the final configuration steps for such an agent. To show the whole process, we are going to use a SunOS/SPARC server in which we can install it and get the first results. To begin with, we need to set up the following Perl module dependencies for `Net::SMTP`, which requires the following modules available at CPAN site:

- Crypt::DES 2.03
- Digest::HMAC 1.00
- Digest::SHA1 1.02

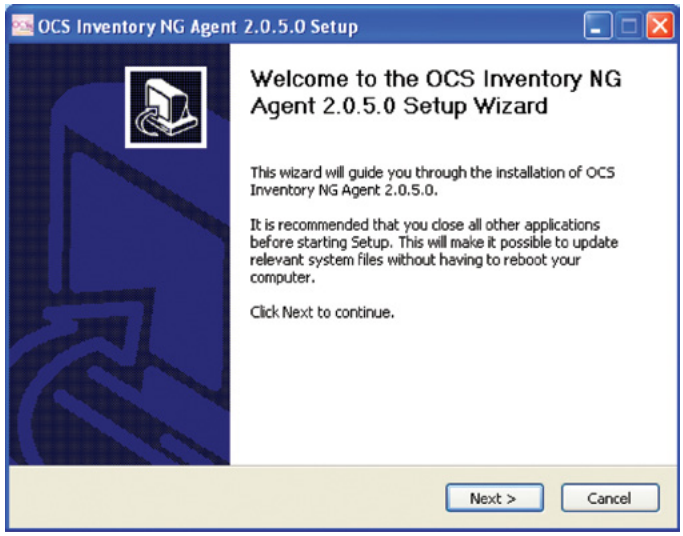


Figure 5. *OCS Agent for MS Windows installation process (I)*

Listing 4. *Questions*

```
Where do you want to write the configuration file?
 0 -> /etc/ocsinventory
 1 -> /usr/local/etc/ocsinventory
 2 -> /etc/ocsinventory-agent
?>  2
Do you want to create the directory /etc/ocsinventory-agent?
Please enter 'y' or 'n'?> [y] y
[info] The config file will be written in /etc/ocsinventory/ocsinventory-agent.cfg,
What is the address of your ocs server?> [ocsinventory-ng] localhost
Do you need credential for the server? (You probably don't)
Please enter 'y' or 'n'?> [n]
Do you want to apply an administrative tag on this machine
Please enter 'y' or 'n'?> [y]
tag?>  ecl51991
ocsinventory agent presents: /usr/local/bin/ocsinventory-agent
Do you want to install the cron task in /etc/cron.d
Please enter 'y' or 'n'?> [y]
Where do you want the agent to store its files? (You probably don't need to change it)?> [/var/lib/ocsinventory-
   agent]
Where do you want the agent to store its files? (You probably don't need to change it)?> [/var/lib/ocsinventory-
   agent]
```

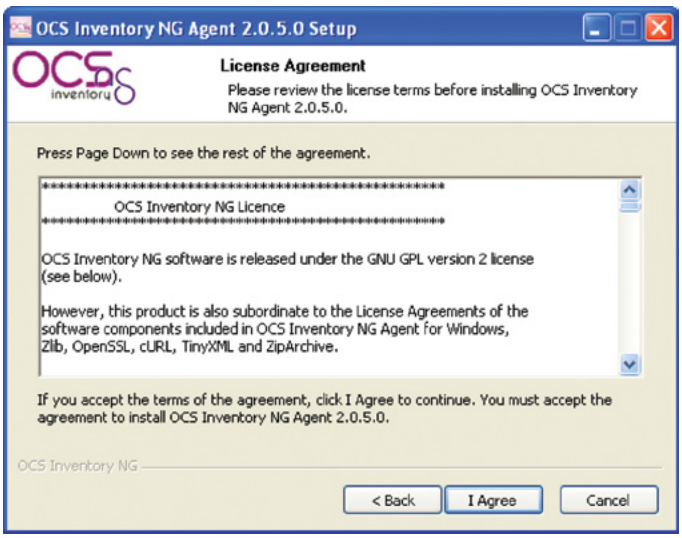


Figure 6. *OCS Agent for MS Windows installation process (II)*

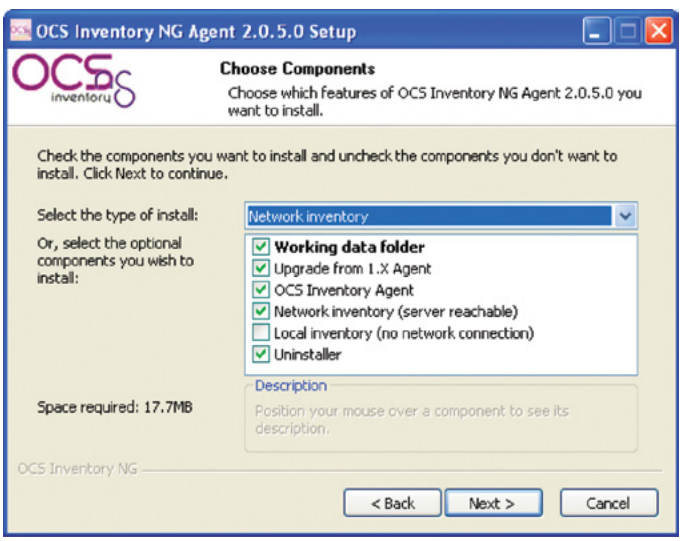


Figure 7. *OCS Agent for MS Windows installation process (III)*

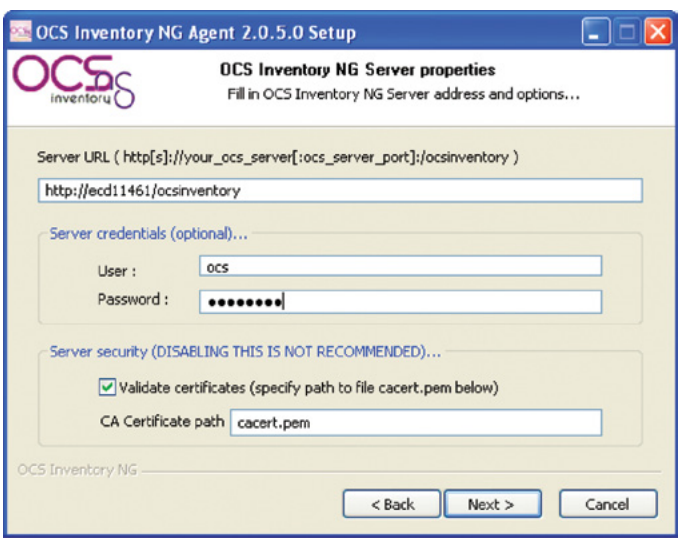


Figure 8. *OCS Agent for MS Windows installation process (IV)*

Once the —setup.sh— script is running, A few questions must be answered (See Listing 4). Hence, it is possible to start the process of periodically gathering all available information provided by this Solaris host. Notice that this info will be stored in the MySQL database defined for OCS, which acts as a backend of our IT inventory/assets platform.

MS Windows OCS NG Agent

Installing OCS NG Agent for MS Windows platforms is a really straightforward process as an installer, shown by Figures 5 to 8, which is provided to simplify the process. As a result, the configuration file is placed in —ocsinventory.ini— whose contents are shown in Listing 5.

Hardening the OCS NG Server

In order to avoid the proliferation of permissions and redundant granting, it is convenient to follow the sequence of steps below to strengthen security in our OCS NG server:

Listing 5. *The configuration file is placed in —ocsinventory*

```
[OCS Inventory Agent]
ComProvider=ComHTTP.dll
Debug=1
Local=
NoSoftware=0

HKCU=0
NoTAG=0
IpDisc=
[HTTP]
Server=http://ecd11461/ocsinventory
SSL=1
CaBundle=cacert.pem
AuthRequired=1
User=53kCz99Idagoh1maatjMVA==|||4ZqRoYC8QEbwlfa8iP2NCg==
Pwd=nh5fWTdG44Nb+x8OxXrZXg==|||BsGQu3tuUhNiLQy+AjV4Ng==
ProxyType=0
Proxy=
ProxyPort=0
ProxyAuthRequired=0
ProxyUser=
ProxyPwd=
[OCS Inventory Service]
TTO_WAIT=15540
PROLOG_FREQ=24
OLD_PROLOG_FREQ=24
Now the installed MS Windows agent is ready to gather
all data concerning the hardware and software in order
to populate the MySQL database created during the OCS
Inventory NG installation process.
```

1. Set up permissions.

```
# chown -R root:apache /usr/share/ocsinventory-reports/
    ocsreports
# chmod -R g+w /usr/share/ocsinventory-reports/ocsreports
```

2. Create old _ conf— directory with writing permissions for Apache user.

```
root@ecl51991:/usr/share/ocsinventory-reports/ocsreports/
    plugins/main_sections/conf/old_conf
```

3. Ensure that the user/password of MySQL in —z-ocsinventory-server.conf— is OK.

```
  # Name of database
  PerlSetEnv OCS_DB_NAME ocsweb
  PerlSetEnv OCS_DB_LOCAL ocsweb
  # User allowed to connect to database
  PerlSetEnv OCS_DB_USER ocs
  # Password for user
  PerlSetVar OCS_DB_PWD ocs
```

A working prototype is installed and available at *http://ecd11461/ocsreports* for free.

The access to *http://ecd11461/ocsreports* is granted by default to user 'admin'. This fact shall be kept in mind to configure OCS NG Agent for other equipment.

Asset Management Platform

Asset Management Installation Process

In contrast to OCS NG server, the installation of GLPI is easier to perform for Unix platforms and may be summarized in the following stages:

1. Create database. The default login/passwords once the MySQL database has been initialized are:
   - glpi/glpi for the administrator account
   - tech/tech for the technician account
   - normal/normal for the normal account
   - post-only/postonly for the postonly account

   and that is all and it is ready for use at the following URL: *http://localhost/glpi*
2. Install OCS Import plugin (1.6.1) and uncompress in plugins directory. Select setup->plugins in GLPI web console.
3. Enable OCS NG mode in GLPI web console

```
../../../glpi/files/_log/ocsng_fullsync.log
../../../glpi/files/_log/sql-errors.log
../../../glpi/files/_log/php-errors.log
../../../glpi/files/_log/cron.log
```

Inventory-Assets Interface

GLPI/OCS NG Plug-in Installation Process

Log into the GLPI web console interface at *http://ecd11461/glpi/* by using "glpi" administrator user and select the following options:

- Setup->General Setup and activate
- Setup->OCSNG mode

and select "localhost" by setting up the following parameters:

OCSNG database	ocsweb
OCSNG database user	ocs
OCSNG database password	pass
OCSNG database in UTF8	Yes
Active	Yes

Also ensure that the Web address of the OCSNG console points to *http://ecd11461/ocsreports* which is the OCS NG Console URL address.

Ensure that Plugins -> OCS Import has been setup for localhost OCS server. In this way, we are ready to import all data gathered by the inventory management platform based on OCS NG to GLPI as the platform to manage all assets associated to these items, which are part of the inventory.

GLPI Data Import from OCS NG Inventory

A plugin for GLPI named OCS NG provides a unidirectional interface from OCS NG data to GLPI in order to develop the required asset management.

Once all inventory data has been gathered by OCS NG Inventory database, this data can be transferred into the GLPI database by using OCS NG plugin for GLPI, according to the picture given by Figure 9.

As a result of the latter import, the GLPI database contains all new inventory data as shown in Error: Reference source not found, which will be the basis to define some company assets, such as:

- Maintenance Contracts
- Support Contracts
- Licensing Renewals
- Incidences and Problems

Notice that the data collected automatically by OCS NG Agents is periodically updated, so that this data can be synchronized with GLPI in order to get an updated view of all assets in the company.

The final picture achieved at the end of the process described above is depicted by Figures 10 and 11.

Main Features

Rather than giving a list of the main features provided by this asset management platform, it is much more useful for end-users to give answers to a set of common questions to illustrate some practical applications:

Dynamical groups

Let me use a real case in our working environment in which we have a running OCS/GLPI installation to handle inventory and asset management. Suppose you are asked for the number of personal computers having IBM DOORS(TM) 9.1 installed and their hostnames. The answer is logically provided by the dynamic groups feature.

Dynamic groups allow categorization of systems by different criteria, i. e. those computers having DOORS 9.1 installed or other software as it is shown by Figure 12.

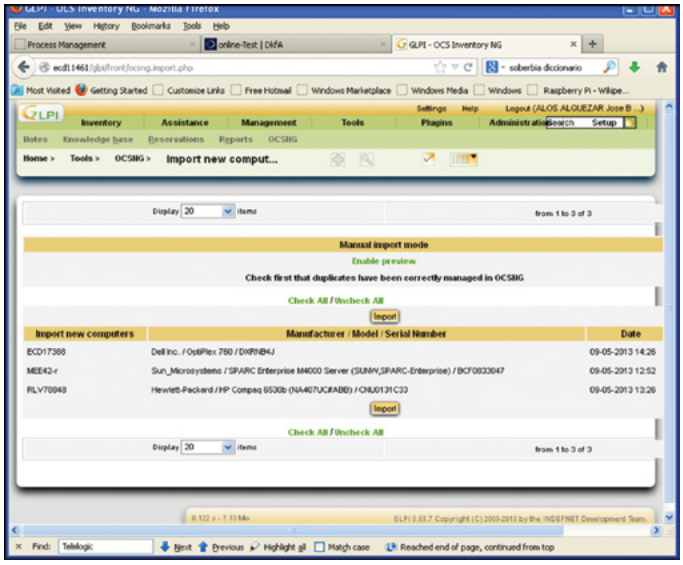


Figure 9. *Data import from OCS NG Inventory server to GLPI Asset Management server*

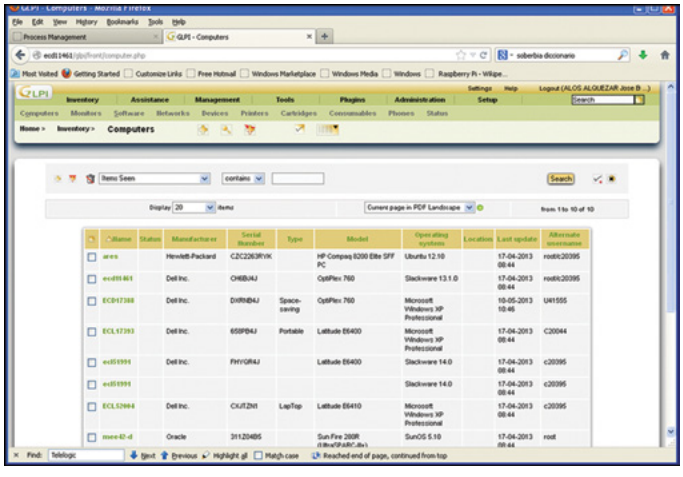


Figure 10. *GLPI imported data from OCS NG Inventory server*

To conclude, the process of creating one of such dynamic groups consists of choosing in our OCS Inventory server URL the option 'Search with various criteria' and then selecting all items matching the software name we are looking for.

Conclusions and Remarks

One of the most common issues facing heterogeneous IT platform management for large companies is the lack of suitable tools to tackle daily activities in order to control and trace changes in hardware and software installed, as well as the tasks related to maintenance support and license renewals that can affect some IT equipment. Hence, to shed some light as well as to avoid the use of manual inventories, which constitute a big source of mistakes and are difficult to update and control, we have described the detailed steps to be followed in order to set up a seed platform to deal with big IT platforms.

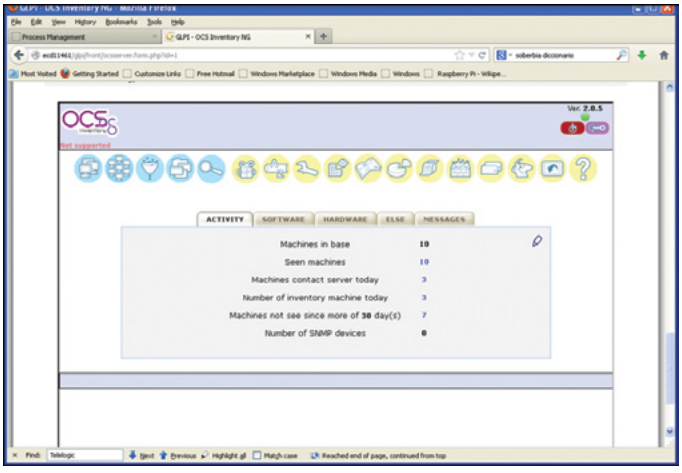


Figure 11. *Data import process from OCS NG Inventory server*

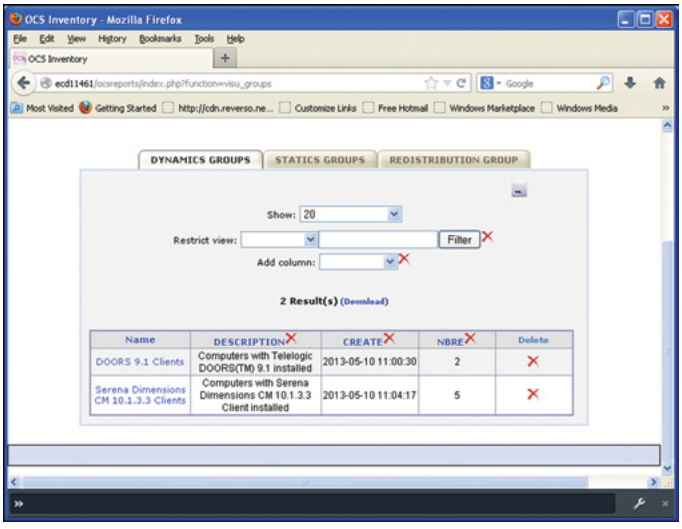


Figure 12. *Dynamical groups usage*

References

The two main references to get all source code and related documentation for OCS Inventory NG and GLPI software are:

- OCS Inventory NG – *www.ocsinventory-ng.org*
- GLPI Assets Management – *www.glpi.org*

Also recommended is the main site for Comprehensive Perl Archive Network in which you can find all necessary modules to deploy SOAP-based features required by the tandem OCS/GLPI.

- CPAN Main site – *www.cpan.org*

There are, however, a lot of commercial products that could provide you the necessary help and support, but they can be expensive and what is more, have a degree of complexity that is not required in many cases. In such a case the choice of OCS/GLPI tandem suits well to serve your needs, which is especially important if you cannot afford to pay the fees for such commercial solutions.

Although the purpose of this article is not to show all advanced features available, we recommend strongly to read a copy of the book "IT Inventory and Resource Management with OCS Inventory NG", written by Barzan Antal, which constitutes a great support for advanced topics on this question and allows to get a taste of the possibilities offered for IT Administrators.

Acronyms and Abbreviations

BIOS	Basic Input-Output System
COTS	Commercial-Off The Shell
CPAN	Comprehensive Perl Archive Network
FOSS	Free Open Source Software
GLPI	Gestionnaire Libre de Parc Informatique
IT	Information Technology
NG	Next Generation
OBP	OpenBoot PROM
OCS	Open Computers and Software
PROM	Programmable Read-Only Memory
SNMP	Simple Network Management Protocol

JOSÉ B. ALÓS

José B. Alós began his professional career in 1999 with EDS, as a UNIX System Administrator mainly focused on SunOS/Solaris, BSD and GNU/Linux. Five years ago, he joined EADS Defense and Security. Nowadays he works for CASSIDIAN where he is responsible for providing end-user support in aircraft engineering departments for long-term projects. He was also an Assistant Professor in the Universidad de Zaragoza (Spain), specializing in the design of High Availability solutions and his academic background includes a PhD in Nuclear Engineering and three MsC degrees in Electrical and Mechanical Engineering, Theoretical Physics, and Applied Mathematics.

UPDATE NOW WITH STIG AUDITING

"IN SOME CASES **nipper studio** HAS VIRTUALLY **REMOVED** the NEED FOR a **MANUAL AUDIT**"

CISCO SYSTEMS INC.

Titania's award winning Nipper Studio configuration auditing tool is helping security consultants and end-user organizations worldwide improve their network security. Its reports are more detailed than those typically produced by scanners, enabling you to maintain a higher level of vulnerability analysis in the intervals between penetration tests.

Now used in over 45 countries, Nipper Studio provides a thorough, fast & cost effective way to securely audit over 100 different types of network device. The NSA, FBI, DoD & U.S. Treasury already use it, so why not try it for free at www.titania.com

2012 Computing Security Awards WINNER Enterprise Security Solution of the Year

Security Products' GOVIES 2013 Government Security Awards

2012 Computing Security Awards WINNER Network Security Solution of the Year

www.titania.com

FreeBSD Programming Primer – Part 10

In the tenth part of our series on programming, we will improve the login process, add more security, and keep spam robots under control.

What you will learn…

- How to to configure a development environment and write HTML, CSS, PHP and SQL code

What you should know…

- BSD and general PC administration skills

In the previous article we put in place a very crude login system that allowed anyone to login to our CMS and add content. We assume that the user has been correctly authenticated by comparing their password against a hashed password stored in the CMS database, then writing a cookie at the client side. It is then a simple matter of checking that authorization has been granted prior to carrying out sensitive actions (e.g. adding a user or amending content).

Listing 1. *Logout function*

```
function logout(){

  setcookie(KEYNAME, LOGINKEY, time()-3600, "/");
  echo "You have been logged out";

}
```

Listing 2. *Adding the logout logic*

```
   }elseif($action == "appendnewlogin"){

      $username = $_POST["username"];
      $password = $_POST["password"];
      $auth = $_POST["auth"];

      appendnewlogin($username,$password,$auth,$sql);

   }elseif($action == "logout"){

      // Logout the user

      logout();

   }else{

      // Invalid action - request login details
      requestlogindetails();
   }
```

Listing 3. *logoutform*

```
function logoutform(){

// Check if user is logged in, if so display the logout
   button.

require_once 'includes/cms.inc';
require INCLUDES . 'login.inc';

  if(isset($_COOKIE[KEYNAME])){

    echo '<div id="logout">';
    echo '<form action="login.php" method="post">';
    echo '<input type="submit" value="logout">';
    echo '<input type="hidden" name="action"
   value="logout">';
    echo '</form>';
    echo '</div>';

  }

}
```

Faster.
Better.
Reliable.

Trusted by over 500 ISPs worldwide.

Hyper is the first multimedia cache fully developed in Brazil, by Taghos.
With Hyper, ISPs can save on network bandwidth while increasing content-delivery speeds, resulting in end-customer satisfaction.

Features:

- 24x7x365 always-on support
- Active monitoring
- Automatic updates
- Appliance or license
- Easy deployment
- Configuration and reports via web interface

hyper

Remote Install

Using your hardware

Model	Traffic	RAM	Cache	SSD
T15	Up to 15 Mbps	8 GB	1x 1 TB	-
T50	Up to 50 Mbps	8 GB	2x 1 TB	-
T100	Up to 100 Mbps	8 GB	2x 1 TB	1x 160 GB
T150	Up to 150Mbps	16 GB	3x 2 TB	1x 160 GB
T300	Up to 300 Mbps	16 GB	5x 2 TB	1x 240 GB
T500	Up to 500 Mbps	32 GB	7x 2 TB	1x 480 GB
T1000	Up to 1 Gbps	64 GB	10x 1 TB	1x 480 GB
T2000	Up to 2 Gbps	96 GB	24x 1 TB	3x 480 GB
T3000	Up to 3 Gbps	128 GB	32x 1 TB	5x 480 GB

Visit us at **www.taghos.com** and start saving bandwidth today!

Unfortunately, exposing any login system on the World Wide Web leaves us open to undesirable elements. Brute force attacks (repeatedly attempting a login using dictionary attacks) and spambots that want to add advertising or phishing spam are commonplace, and our basic login system needs to defend against this. We also need to add logout functionality to every page that requires it.

The logout functionality

As the parameters passed to the cookie that is set when we are logged in, it makes sense to hold the logout function as part of the *login.php* page. We can then detect a logout post event to *login.php* and delete the cookie by setting the expire date to a time in the past. Add the following code at the end of *login.php* (Listing 1).

Now we need to check for a post event that carries the value logout. Add the following elseif branch between *append* and the closing *else* (Listing 2).

We now need a `logoutform()` function that will provide a logout button whenever a user is logged in to the system. If we check whether or not the user is logged in we can place this in the footer of all pages where login / logout

Figure 1. *Login – no cookie present*

Figure 2. *Cookie present but no logout button*

Listing 4. *Test to see if user is logged in*

```
function ifnotloggedin(){
  // Check if user is logged in, if not, redirect to
   login form

  require_once 'includes/login.inc';

  if(!isset($_COOKIE[KEYNAME])){

   header( 'Location: http://'.CMSDOMAIN.'/login.php' )
   ;

  }

}
```

Listing 5. *Set our domain*

```
// Our domain

define("CMSDOMAIN", '192.168.0.118');
```

Listing 6. *amendcontent.php*

```
// Check we are logged in
ifnotloggedin();
// Build the page up to the body tag
outfile(TEMPLATES . 'header.inc');
```

Listing 7. *phpinfo.php*

```
<?php

// Check we are logged in

require_once 'includes/cms.inc';
require INCLUDES . 'content.inc';
require INCLUDES . 'core.inc';
ifnotloggedin();
phpinfo();
logoutform();
```

Listing 8. *amendcontent.php and login.php*

```
echo BODY;
logoutform();
```

Listing 9. *global.css*

```
#logout {
  float: right;
  background-color: tomato;
  padding: 5px;
  border-radius: 10px;
}
```

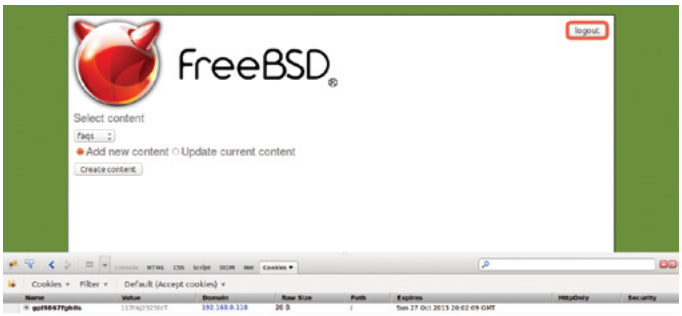


Figure 3. *Cookie present – logout button visible*

Figure 4. *Logout button visible on new faq's page*

Figure 5. *Logout message*

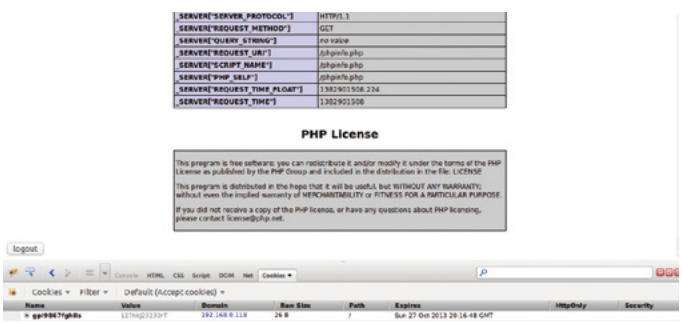


Figure 6. *Logout button on phpinfo.php*

Listing 10. *validatelogin()*

```
setcookie(KEYNAME, LOGINKEY, time()+3600, "/");

// Display options

$title = 'Welcome ' . $username;

buildheader($title);
echo wraptag("h1",$title);

echo ahref('Add or amend content', '/amendcontent.
   php');

buildfooter();
```

Listing 11. *Replacement buildheader();*

```
function buildheader($title, $forcelogout = 0){

   // As cookies need to be set before any output is
   sent to the browser
   // use a function call to build the page header

   // Build the page up to the body tag

   outfile(TEMPLATES . 'header.inc');

   echo wraptag('title', $title);
   echo HEAD;
   echo BODY;
   logoutform($forcelogout);

   echo '<div id="content">';
   echo '<div id="php">';

}
```

Listing 12. *Amended validatelogin();*

```
setcookie(KEYNAME, LOGINKEY, time()+3600, "/");

// Display options

$title = 'Welcome ' . $username;

buildheader($title,1);
echo wraptag("h1",$title);
```

functionality is required, and the button will be displayed only if the user is logged in. Add this to the end of core.inc (Listing 3).

We need to add a function call to check if the user is logged in or not, and redirect them to the login page if they are not. Add this at the end of core.inc (Listing 4).

As we cannot guarantee that the user does not spoof HTTP headers for the redirect, define our CMS Domain in *cms.inc*. Replace 192.168.0.118 with either the IP address or domain name of your server (if accessible via DNS). (Listing 5)

Add the `ifnotloggedin()` function call to the beginning of *amendcontent.php* and replace *phpinfo.php* with the content in Listing 7 (Listing 6 & 7).

Figure 7. *The fixed welcome page*

Listing 13. *Amended logoutform();*

```
function logoutform($forcelogout = 0){

  // Check if user is logged in, if so display the
   logout button.

  require_once 'includes/login.inc';

  if(isset($_COOKIE[KEYNAME]) || $forcelogout == 1){

    echo '<div id="logout">';
```

Listing 14. *Add spambot field to requestlogindetals() in login.php*

```
echo 'Username' . div('<input type="text"
   name="username">',$class);
echo 'Password' . div('<input type="password"
   name="password">',$class);
echo 'Email' . div('<input type="text"
   name="email">','loginemail');
echo '<input type="submit" value="Submit">';
```

Listing 15. *Remove the comment out from createnewlogin, suffix with // to revert to normal login action*

```
if(!isset($_POST["action"])){

   createnewlogin();

}
```

Listing 16. *xx*

```
CREATE TABLE `access` (
  `id` int(10) unsigned zerofill NOT NULL AUTO_INCREMENT,
  `ipaddress` varchar(64) NOT NULL,
  `page` varchar(64) NOT NULL,
  `status` int(1) NOT NULL,
  `timestamp` timestamp NOT NULL DEFAULT CURRENT_
   TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=0 DEFAULT CHARSET=latin1;
```

Listing 17. *sqlstatements.inc*

```
<?php
/*
 *
 * sqlstatements.inc
 * Contains CMS SQL statements
 *
 */

$sql[0] = "INSERT INTO `access` (`ipaddress`, `page`,
   `status`, `timestamp`)
         VALUES ('---P0---', ('---P1---'), '---P2---',
   now());";

$sql[1] = "SELECT status FROM access
           WHERE ipaddress = '---P0---'
           AND status > 0
           LIMIT 1";

Add the following line to cms.inc [Listing ]

// Honeypot for bad traffic

define("HONEYPOT", 'www.google.com');
```


Dr.WEB®
since 1992
ERA Dr.Web
Emergency Response Anti-virus
Dr.Web 9.0
for Windows —
the rapid response anti-virus
1. Reliable protection against the threats of tomorrow
2. Reliable protection against data loss
3. Secure communication, data transfer
and Internet search
www.drweb.com
Free 30-day trial: https://download.drweb.com
New features in Dr.Web 9.0 for Windows: http://products.drweb.com/9
FREE bonus — Dr.Web Mobile Security: https://download.drweb.com/android
© Doctor Web
2003 — 2013

Listing 18. *mysql_select()*

```
function mysql_select($sql) {

    $db = new mysqli(DBSERVER, DBUSER, DBPASSWORD,
   CMSDB);

    if ($db->connect_errno > 0) {
        die('Unable to connect to database [' .
   $db->connect_error . ']');
    }

    if (!$result = $db->query($sql)) {
        if (DEBUG) {
            die('There was an error running the query ['
   . $db->error . ']');
        } else {
            die('');
        }
    }

// Pass our results to an array to be returned

    if(isset($result->num_rows)){

    $r = array();

    $r[] = $result->num_rows;    // No of rows returned
    $r[] = $db->field_count;     // No of columns in
   table
    $r[] = $db->affected_rows;   // No of rows affected
   e.g. update / delete

// Append the results to our result count

    if ($result->num_rows != 0) {

        $r = array_merge($r, $result->fetch_
   array(MYSQLI_ASSOC));
    }

    // Free the result

    $result->free();


   }else{

      $r = NULL;

    }

// Close the connection

    $db->close();

    return $r;
}
```

Listing 19. *Additions to core.inc*

```
function loginsecurity(){

  require INCLUDES . 'sqlstatements.inc';

  // Get client IP address

  $ip =  $_SERVER["REMOTE_ADDR"];

  if(isset($_POST["email"])){

   // email will always be set, check if it is populated

   if($_POST["email"] !== ''){

      // Ban 'em

      banip($ip, 'login.php');

   }

  }else{

    // Check that they have not been flagged as
   suspicious

   $s = $sql[1];
   $s = str_replace ( '---P0---' , $ip , $s );

   $result = mysql_fetchrows($s);

   if($result){

     foreach($result as $row){

       $status = $row[0];

     }

   }else{
```

```
        $status = 0;

    }

    // Redirect to our honeypost if status is set

    if($status !== 0){

        header( 'Location: http://' . HONEYPOT ) ;

    }

  }

}

function banip($ip, $page){

  require INCLUDES . 'sqlstatements.inc';

      // Add to our banlist

      $s = $sql[0];
      $s = str_replace ( '---P0---' , $ip , $s );
      $s = str_replace ( '---P1---' , $page , $s );
      $s = str_replace ( '---P2---' , 1 , $s );

      mysql_select($s);

      // Redirect to our honeypot

      header( 'Location: http://' . HONEYPOT ) ;


}

function logip($page){

      require INCLUDES . 'sqlstatements.inc';

      // Just log a visit

      $ip =  $_SERVER["REMOTE_ADDR"];

      $s = $sql[0];
      $s = str_replace ( '---P0---' , $ip , $s );
      $s = str_replace ( '---P1---' , $page , $s );
      $s = str_replace ( '---P2---' , 0 , $s );

      mysql_select($s);

}
```

Listing 20. *Replacement validatelogin()*

```
function validatelogin($username, $password, $sql){

    // Create a session to keep track of our login
    attempts

    session_start();

    // As the password is hashed and hopefully cannot be
    decrypted,
    // We need to send the encrypted password

    $hashed_password = hash('whirlpool', $password);

    // Fetch credentials from DB, if match create a login
    cookie

    $s = $sql[0];
    $s = str_replace ( '---P0---' , $username , $s );
    $s = str_replace ( '---P1---' , $hashed_password , $s
    );

    $result = mysql_fetchrows($s);

    if($result){

      foreach($result as $row){

       $auth = $row[1];

      }

     }else{

      $auth = 0;

    }

    if ($auth == 1) {

       // Log our sucessful login

       logip('login.php');
```

```
    // Reset our attempt count in case they login
again

    unset($_SESSION['loginattempts']);

    // Create auth cookie

    setcookie(KEYNAME, LOGINKEY, time()+3600, "/");

    // Display options

    $title = 'Welcome ' . $username;

    buildheader($title,1);
    echo wraptag("h1",$title);

    echo ahref('Add or amend content', '/amendcontent.
php');

    buildfooter();


}else{

    // Keep a track of the number of attempts we have
made at logging in

    if(isset($_SESSION['loginattempts'])){

      $_SESSION['loginattempts'] = $_
SESSION['loginattempts'] + 1;

    }else{

      $_SESSION['loginattempts'] = 1;

    }

    // If they have exceeded our limit, ban 'em

    if($_SESSION['loginattempts'] > 3){

      $ip = $_SERVER["REMOTE_ADDR"];

      banip($ip, 'login.php');

    }

    // Try again

        requestlogindetails();

    }

}
```

Listing 21. *Modified buildheader()*

```
// As cookies need to be set before any output is sent
   to the browser
   // use a function call to build the page header

   // Check we are not on the ban list and that we are
   not a spam robot

   loginsecurity();

   // Build the page up to the body tag

   outfile(TEMPLATES . 'header.inc');
```

Listing 22. *Hide the email address field*

```
.loginemail {
  visibility: hidden !important;
}
```

Figure 8. *The login page with the email "honeytrap"*

Add the `logoutform()`; after every occurrence after *echo BODY*; in *login.php* and *amendcontent.php* (Listing 8).

Add the following to global.css to highlight and position the logout button (Listing 9).

With Firebug enabled in Firefox, check that a cookie called `gpl9867fghlls` is created when a user is logged in. The logout button should appear on all pages except the second time login.php is called and we arrive at the welcome page (Figure 1 – 6).

Now this is a problem, as we should be able to logout immediately after we login. Subsequent calls to login.php will show the logout button. So what is happening here? The problem lies in the `validatelogin()` function (Listing 10).

We must set the cookie prior to creating the page header, but as the cookie data is generated at the client browser side when the page is loaded, as far as the PHP code running at the server side is concerned the cookie is not present yet. We can fool `buildheader()` by passing a parameter to force the display of the logout button (Listing 11 – 13). This will result in login.php working as desired (Figure 7).

Spambots and robots

While we could use the very effective Apache MOD_SECURITY module to trap bad behaviour, this can be tricky to set up. What we will do here is monitor behaviour in two ways. First, we will create a hidden field that a normal user will not see under normal circumstances, which most spam-robots will fill in assuming it is a genuine field. On completing the field, our CMS will automatically ban all connections from that IP address to login.php permanently.

We will also check that no more than 3 invalid attempts are made to the login.php page, and if that is exceeded, that IP address will be banned as well.

First create another testuser by changing login.php as follows and visit login.php anew to create another user (e.g. Test, Test, Auth = 1). Don't worry about the error messages – we will fix them later. Once you have created the new user, go back and comment out `createnewlogin()`; and check that you can login as the test user (Listing 14 and 15).

If you visit login.php you should be able to login as Test (Ignore the Email field), then Logout. (Figure 8). Now create our access table in MySQL to hold our banlist (Listing 16). Now create the file sqlstatements.inc in our includes directory (Listing 17). Replace the `mysql_select()` function call in mysql.inc with the following code (Listing 18). This fixes a bug where a PHP error is raised when no results are returned. Add the following function calls to core.inc (Listing 19). Replace `validatelogin()` in login.php with the following code (Listing 20). Modify `buildheader()` in *login.php* to call `loginsecurity()` (Listing 21).

Useful links

PHP manual – *http://php.net/manual*

Testing

It is recommended that you run Firebug to view the cookies and PHP sessions generated during this test. Clear all cookies etc. from you browser and visit login.php:

- Login as Test with the correct password. You should be able to login. Logout.
- Login as Test with the correct password and an email address. You should be redirected to google.com. Any visits to login.php will cause a redirect to *google.com*.
- Use Adminer to clear all the entries from the access table.
- Visit login.php and click on the submit button 3 times without making any input. You should be redirected on the 4th attempt.
- Use Adminer to clear all the entries from the access table.
- Visit login.php and login and logout as normal. Your access attempts should be logged correctly with IP address and date.
- Login with a mixture of bad username and good password, good username and bad password. You should be banned on your 4th login attempt.

CSS modification

Finally, add the following code to global.css and refresh your browser with Ctrl F5 a couple of times to clear the cache. The email field should now be invisible to human visitors, but available to robots etc. (Listing 22).

Next steps

It might be a good idea to add the banlist functionality to all pages on a failed login etc. and keep a tally of what pages are accessed etc. legitimately. We also need to add the facility to add a user rather than manually editing code each time. Our CMS is getting quite large, with over 2,100 lines of code (excluding the Jquery libraries) so we will look at refactoring some of this code in the next article.

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.

PfSense + Snort: Fast Approach

Pfsense is a FreeBSD-based distro specially oriented as a security appliance for firewall UTM with many modules ready for more functions. You can integrate things like squid, dansdnsguardian, varnish, mod_security, and... snort!

In-line with firewall! This means integration with firewall rules, you can pass the address from snort alerts sources to firewall to block attacks at wan interface. This is really cool.

Easy and fast, no scripting, no configuration files, just use the web interface to run everything. Also FreeBSD has reported one of the best performance results at TCP/IP stack benchmarks, paying attention to security, so I think this is a good starting point.

The software has a lot of possibilities. You can use it in embedded appliance systems with compact flash storage, on virtual appliances (you can even download a virtual hard disk with installed one), or on any standard machine, booting from a live CD or from HD and there are also isos for x64 and x86 architecture.

For practice, it will be useful if you write down the MAC addresses from your interfaces somewhere.

On this fast startup, we will install a virtual appliance from scratch for the example but the steps are the same for any physical install.

The first step is to download an iso from the official site an iso to boot on our machine. Just enter pfsense.org and go to Downloads. Click on “here on the mirrors” and select a mirror. You will get a list of possible sources, just get the one you need, typically pfSense-2.0.1-RELEASE-[arch].iso.gz.

Once you have downloaded and burned the CD, just start from it and it will show the boot menu with typical options: default, without acpi, safe start... Just let it start at default, and it will ask for Live Boot or Install. Press “i” for install and a GUI will prompt for character and keymap sets, you can choose your own or just change Keymap to the one of your country. Once selected “Accept these Settings” Select Quick/Easy Install and go, this option erases automatically the HD and repartitions it.

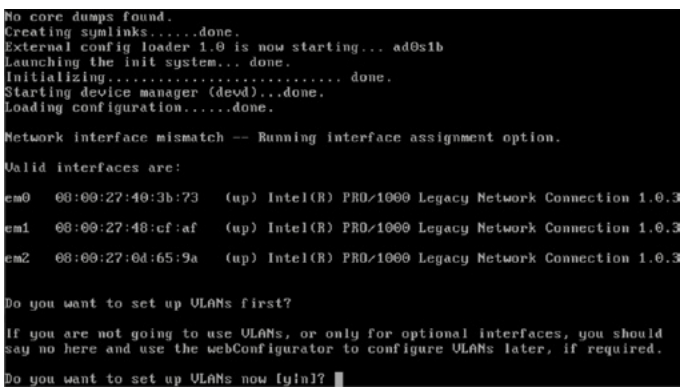


Figure 1. *Vlan Creation Assignment prompt*

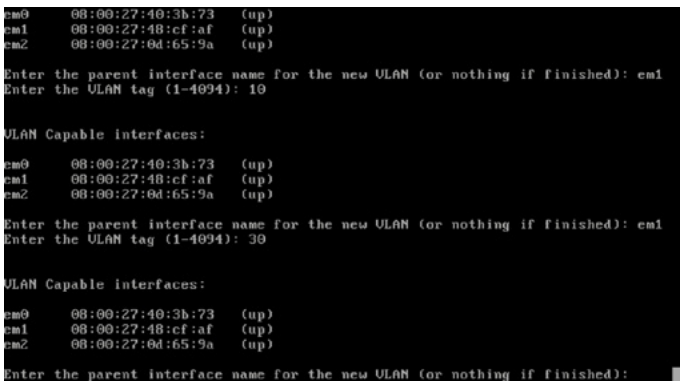


Figure 2. *Vlan Creation Assignment prompt*

The next step is to choose the Kernel type: Symmetric multiprocessor (in case you have more than one core), Embedded Kernel (without VGA and Keyboard for typical rack appliances) and the developers one. For our example, obviously choose the first one.

It copies all necessary files and asks for reboot. Remove the CD from the tray, let it restart and see how it boots.

After the reboot as it says the next default values are loaded:

- IP address at LAN Interface 192,168,1,1
- Username: admin
- Password: pfsense

Anyway an assistant will help us at boot to assign interfaces at startup, It shows a prompt asking for VLANs creation if needed. For this example we will create 2 VLANs at LAN interface, one for data and one for voice, so we answer y.

At this moment, it would be useful to know the Mac address of each interface and where are they connected, here comes the vlan assignment, see Figure 2.

After the Vlan info, you can simply press enter without entering any info and it will show a brief of vlan assignment, and start the interface network assignment, at this point you inform the system what interface will give access to wan network, to lan, dmz and others if you have them.

If you don't know the interface Mac address, the assistant offers an auto-detection method for selecting the correct interface, just disconnect all network wires from the machine and press "a". After that, it invites you to connect the wire to the NIC card selected for that kind of network. In this case, the first one it asks for is WAN, so if you plug a wire on the physical interface and establish the link, the system detects what interface is up and gets its Mac address.

Once you've finished, just press "enter" without the previous value and it will show you a brief and prompt for proceed assignment. Press "y" and "enter".

At this moment, you've reached the end of the basic install (Figure 5). Now you can modify the default IP address assigned for lan network to the one chosen by you, just press 2 and enter, interface, IP and netmask, it will ask you about enabling a DHCP service. Answer as you need for your case. It will then ask about using http protocol for admin. You can answer "y" now and change to https later, as you wish.

Now you have to access from a lan machine to the administration URL shown at brief (Figure 6).

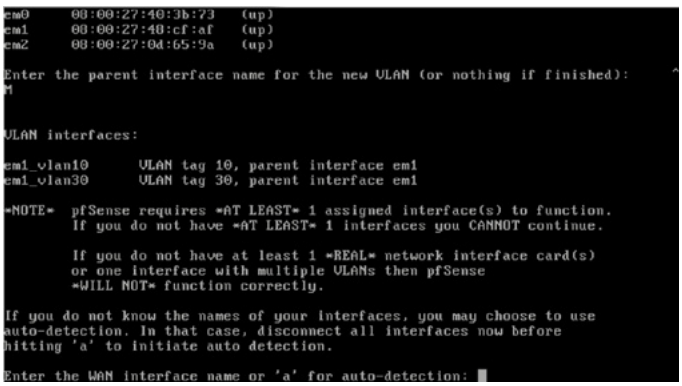


Figure 3. *Vlan Resume and Interface Asignent*

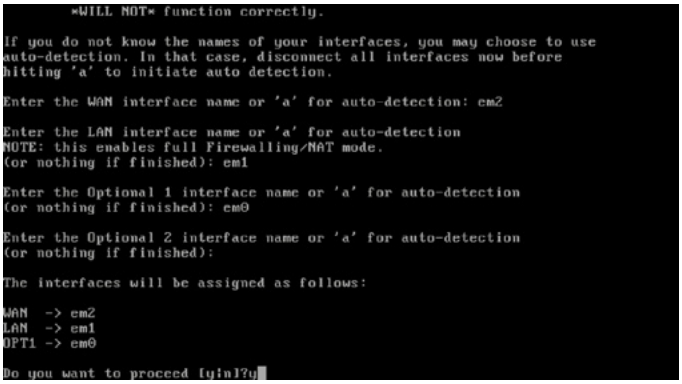


Figure 4. *Interface to network type assignment*

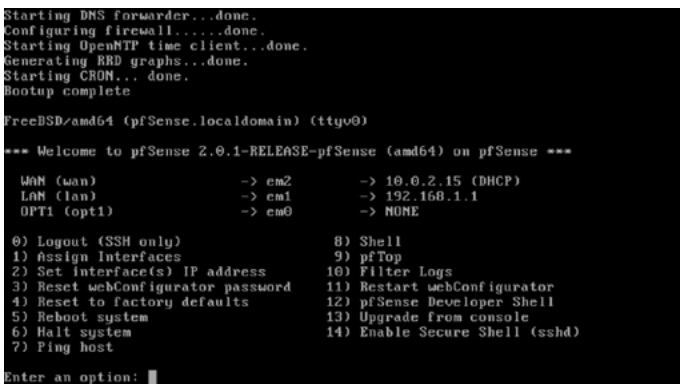


Figure 5. *Default shell startup menu*

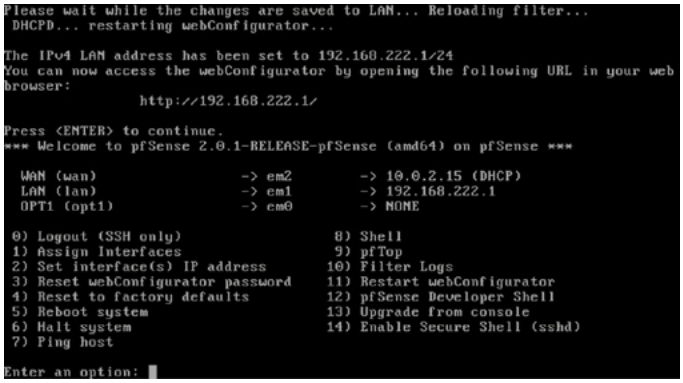


Figure 6. *LAN ip setup and url for web setup*

Once you've logged on to the web interface, you will see the dashboard. It is a customizable panel for an overview of the system where you can add or delete the info plugins you prefer. At the top you can see the drop-down bar that gives you access to all the functions configuration.

At System → Packages you can see a lot of additional packets to improve pfsense functionalities.

Personally, I tested and recommend the following:

Bandwidthd → monitor bandwidth on interfaces and store stats.

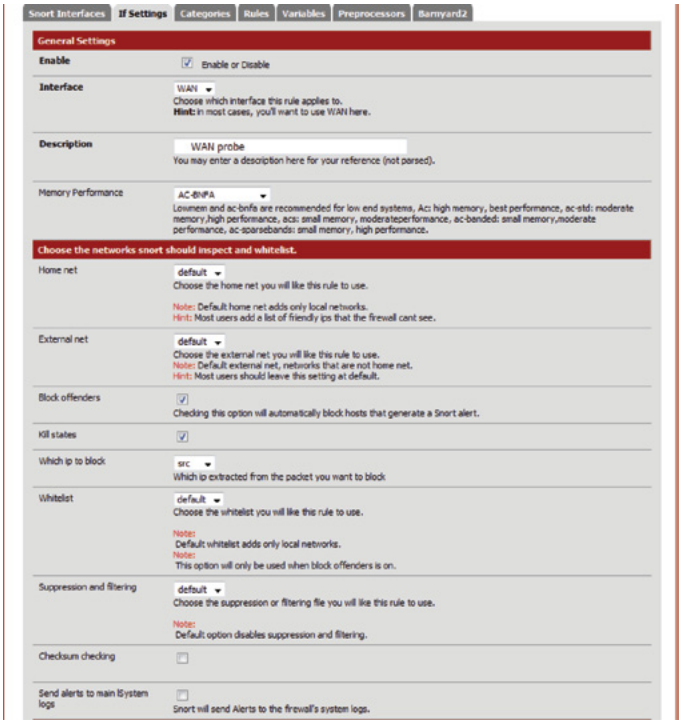


Figure 7. *Pfsense Main Dashboard*

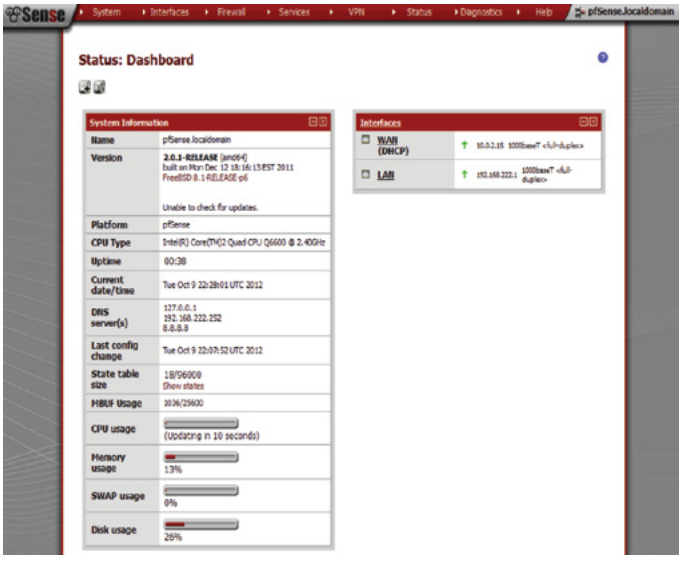


Figure 8. *Packages listst*

Dansguardian → Anti malware and contents access control based on DNS requests, not-free.

IP-Blocklist → For block entire country ip blocks.

squid ->Proxy server for caching and auditing.

Lightsquid->Reports from squid proxy server more pretty.

Mailreport → It sends to you a report from chosen info by email based on a schedule.

OpenVPN Client Export Utility → To create an automated installation package for OpenVpn clients.

Package Name	Category	Status	Package Info	Description
Asterisk	Services	Beta 1.8.8.1 pkg v 0.1 platform: 2.0	Package Info	Asterisk is an open source framework for building communications applications. Asterisk turns an ordinary computer into a communications server.
anyterm	Diagnostics	BETA 0.5 platform: 1.2.3	Package Info	Ajax Interactive Shell - Have you ever wanted SSH or telnet access to your system from an internet deser - from behind a strict firewall, from an internet cafe, or even from a mobile phone? Anyterm is a combination of a web page and a process that runs on your web server that provides this access. WARNING! We suggest using Stunnel in combination with this package!
Apache with mod_security-dev	Network Management	ALPHA 0.2 platform: 2.0	Package Info	ModSecurity is a web application firewall that can work either embedded or as a reverse proxy. It provides protection from a range of attacks against web applications and allows for HTTP traffic monitoring, logging and real-time analysis. In addition this package allows URL forwarding which can be convenient for hosting multiple websites behind pfSense using 1 IP address.
Avahi	Network Management	ALPHA 0.6.29 pkg v 1.02 platform: 1.2.3	Package Info	Avahi is a system which facilitates service discovery on a local network. This means that you can plug your laptop or computer into a network and instantly be able to view other people who you can chat with, find printers to print to or find files being shared. This kind of technology is already found in Apple MacOS X (branded Rendezvous, Bonjour and sometimes zeroconf) and is very convenient. Avahi is mainly based on Lennart Poettering's flexedns mDNS implementation for Linux which has been discontinued in favour of Avahi.
AutoConfigBackup	Services	Stable 1.20 platform: 1.2	Package Info	Automatically backs up your pfSense configuration. All contents are encrypted on the server. Requires pfSense Premium Support Portal Subscription from https://portal.pfsense.org
arping	Services	Stable 2.09.1 platform: 1.0.1	Package Info	Broadcasts a who-has ARP packet on the network and prints answers.
arpwatch	Security	ALPHA 2.1.a15_5 platform: 2.0	No info, check the forum	Arpwatch monitors ethernet/ip address pairings. It also logs certain changes to syslog.
Backup	System	Beta 0.1.5 platform: 1.2	No info, check the forum	Tool to Backup and Restore files and directories.
bandwidthd	System	BETA 2.0.1.3 platform: 1.2.1	No info, check the forum	BandwidthD tracks usage of TCP/IP network subnets and builds html files with graphs to display utilization. Charts are built by individual IPs, and by default display utilization over 2 day, 8 day, 40 day, and 400 day periods. Furthermore, each ip address's utilization can be logged out at intervals of 3.3 minutes, 10 minutes, 1 hour or 12 hours in cdf format, or to a backend database server. HTTP, TCP, UDP, ICMP, VPN, and P2P traffic are color coded.
blinkled	System	Beta 0.3 platform: 1.2.3	Package Info	Allows you to use LEDs for network activity on supported platforms (ALIX, WRAP, Soekris, etc)
bacula-client	Services	Stable 5.2.6 pkg v 1.0 platform: 2.0	Package Info	Bacula is a set of Open Source, computer programs that permit you (or the system administrator) to manage backup, recovery, and verification of computer data across a network of computers of different kinds.

Figure 9. *Snort General Setup*

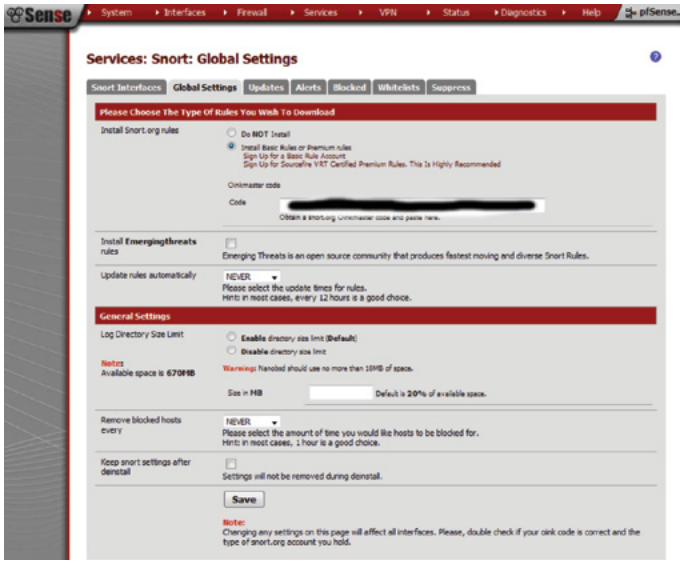


Figure 10. *Snort Instance on wan interface*

PfBlocker → to block host ranges recovered from internet black listings of spammers, malware...

snort → Our main target at this article, the snort IDS.

To install, simply click at the icon with plus symbol, it will download and install the snort package.

Now you need to register at the snort web site (snort.org) in order to get an OINKCODE. This is a code to validate against the snort users database and download the intrusion detection signatures.

Once registered and confirmed, log in with your account on the snort site and goto “My Account”. There you can find a folder called Subscriptions and Oinkcodes where you can get your oinkcode free.

Copy it and go to the pfsense web interface. Click on Service->Snort and go to “Global Settings”, here you can change radio button to install Basic rules from snort and paste your code.

You can also mark the install Emerging Threats option to receive more attacks signatures from Emerging Threats community, and choose and update period. I use it to get daily updates.

It’s also a good practice if you have a small disk to limit the growth of logs and data to keep everything working, so set the Enable size limits to a cypher that fits your disk size.

The last important option at this point is the time that a supposed attacker is being blocked at the firewall, snort can remove hosts from blocked lists periodically, if you wish to block them.

When you finish click on “save” and go to updates for signatures download. You only have to press the button shown to launch the manual download of the signatures. When finished, click “return”.

Now we must tell the systems where snort will listen. If we go to the snort interfaces folder, we can add an instance of snort listening on each interface, typically it only is being run on WAN / DMZ but for more complex scenarios it can be a good idea to put a probe on different layers of the network to see what alerts are being blocked on wan and if someone is being alerting too on LAN or intermediate networks if you think of network security layers as an onion.

The important fields are:

- Enable: cheked to enable interface
- Interface: WAN to select wich one is going to be enabled
- Memory performance: ac-bnfa is enough for starting.
- Block offenders: checked this is the magic, to block who raises an alert at snort IDS.
- Kill States: checked To also kill current connections of attackers.*
- Which ip to block:src, we want to block typically the source.
- Checksum setting: checked to check packet cheksum

*At the beginning, it is not recommended to block offenders. First of all, you must see how it works and which kinds of alerts are raised.

Once you finish, just save and the interface will reload itself showing the interface instance of snort with a play icon.

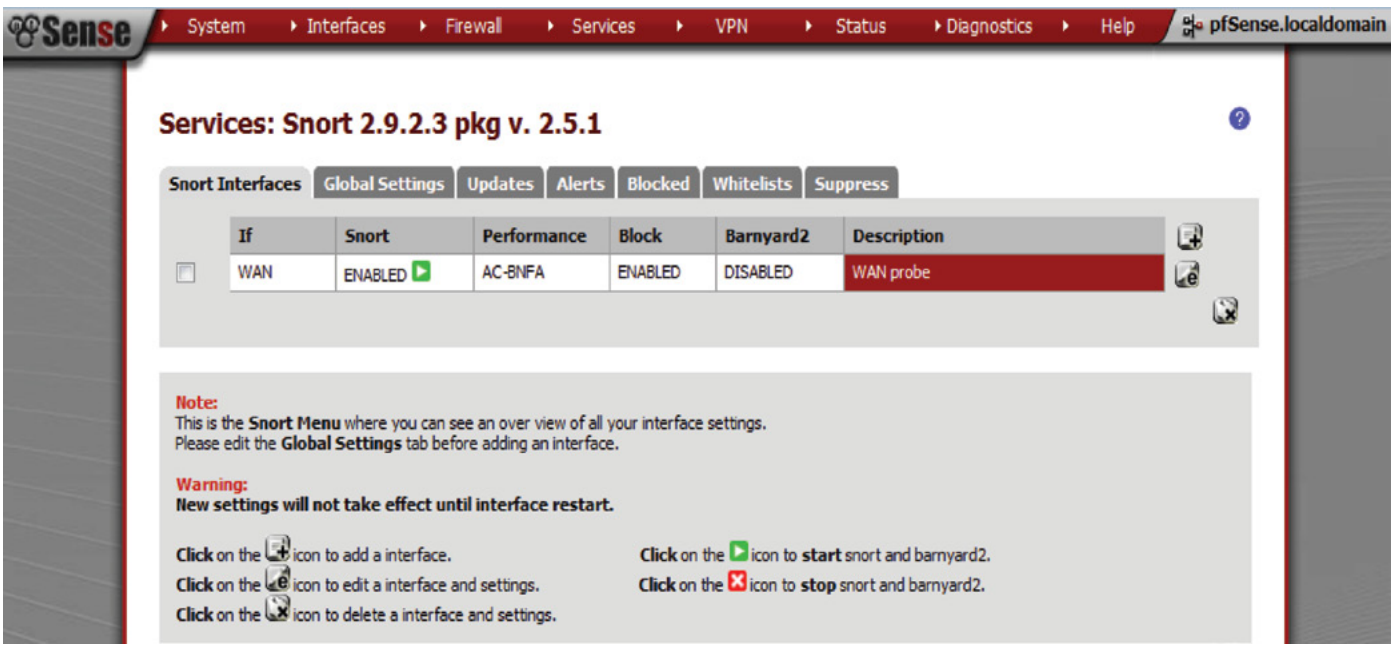


Figure 11. *Snort instance on WAN interface*

The last step is to select which kind of signatures we want to check on the interface. If you click on “edit icon” for instance, and go to “Categories”. A list of all signatures is shown. To start, check all of them. Later if one of them gives any problem, you can disable it by unchecking the box, or you can even enter the rules folder and disable only a signature rule of a category.

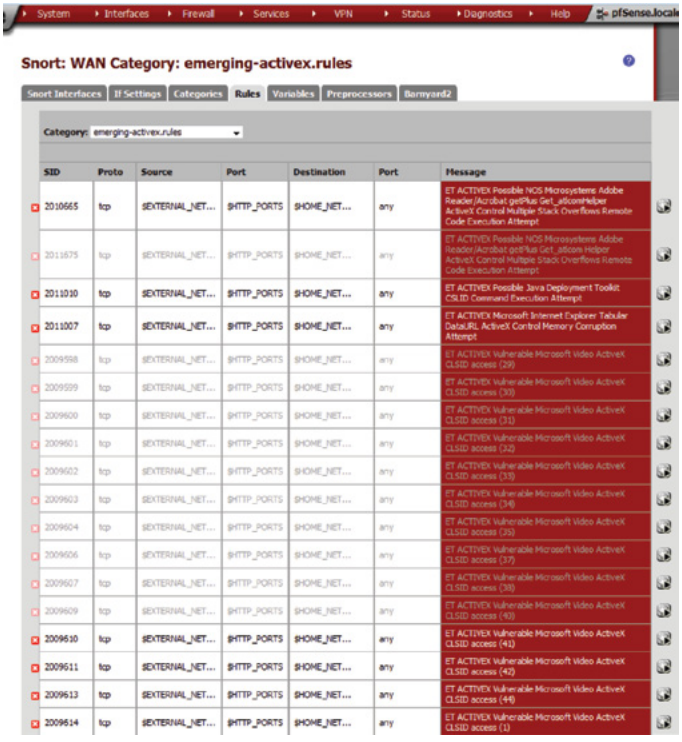


Figure 12. *Snort interface instance rules edit*

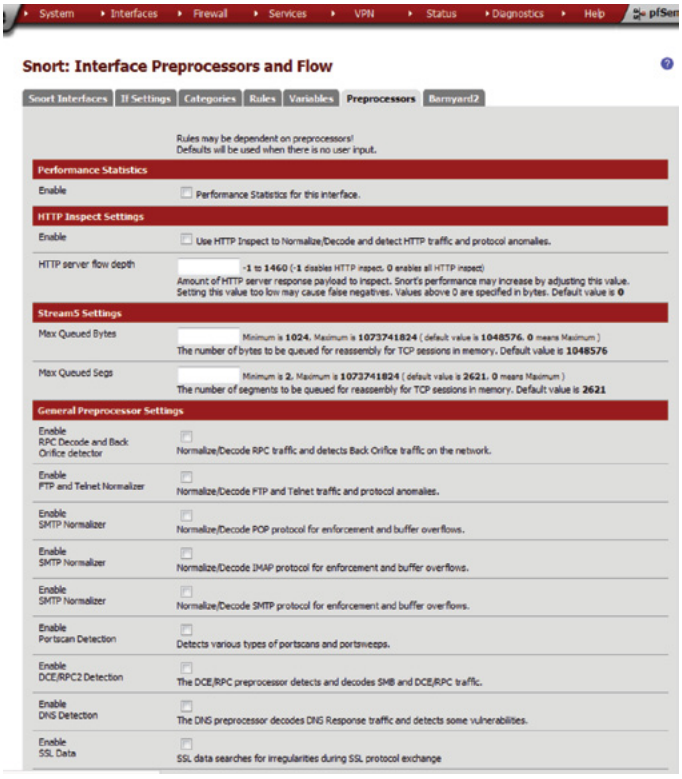


Figure 13. *Snort Interface Instance Preprocesors settings*

.Sometimes it happens that a signature has some kind of problem with the snort plugin or gives a problem with a company application, this allows you to disable conflicting rule and allow everything to work at least while another solution is found. Or perhaps, you want to restrict Instant messaging except for XMPP because it is used by Cisco Unified communications systems at your company. This feature allows you to disable a specific rule only at that interface.

Another important feature is the snort preprocessor options to interpret different protocols to detect anomalies. It can decode and Normalize a lot of protocols: HTTP, RPC, FTP, SMTP, DNS, SSL, portscan detection...

Also Pfsense by default does something called packed scrubbing, which tries to protect some vulnerable systems from attacks with fragmented packets, like teardrop or others.

Check whatever you want to use and go to snort interfaces folder, press play and you’re all done.

The last step is to test all the environments. In the alerts folder, you can see alerts generated by Snort. You must correlate alerts with your environment and define what is a real alert and what is not, perhaps some of your apps have a bug and send some kind of request that raises a false positive from a trusted location, or you must ignore or suppress alerts from Internal instant messaging, you must test and follow alerts for some time to define a base line of normal working traffic. Once you have done this, your IDS can be activated in in-line mode (blocking IDS offenders), more securely for production continuity. But don’t think this is a person doing their job, this is a program and needs supervision to keep on track but helps a lot protecting holes while you find something better to cover them.

I encourage all of you to test this marvelous software and experiment with packets and plugins. There is an incredible amount of features to use at the cheapest price. You even have paid support if you need it.

I have no relation with the authors of the software and they are the artists of this opera. The applause is for them, long live pfsense.

SALIH KHAN

Great Specials

On FreeBSD® & PC-BSD® Merchandise

Give us a call & ask about our
SOFTWARE BUNDLES
1.925.240.6652

$39.95
FreeBSD 9.1 Jewel Case CD Set
or FreeBSD 9.1 DVD

$29.95
PC-BSD 9.1 DVD

$49.95 Save a BUNDLE!
The PC-BSD 9.0 Users Handbook
PC-BSD 9.1 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 9.1 CD or DVD set
FreeBSD Toolkit DVD

Stylish Dress Attire
Look Your Professional Best

Comfy Apparel
Stay Warm in Zip Ups & Pullovers

T-Shirts
Lots of Styles to Choose From

FreeBSD 9.1 Jewel Case CD/DVD $39.95

CD Set Contains:

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

FreeBSD 9.0 CD $39.95
FreeBSD 9.0 DVD $39.95

FreeBSD Subscriptions

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

FreeBSD Subscription, start with CD 9.1 ..... $29.95
FreeBSD Subscription, start with DVD 9.1 ..... $29.95
FreeBSD Subscription, start with CD 9.0 ..... $29.95
FreeBSD Subscription, start with DVD 9.0 ..... $29.95

PC-BSD 9.1 DVD (Isotope Edition)

PC-BSD 9.1 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 9.1 ..... $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

How Secure can Secure Shell (SSH) Be?

(BASIC CONFIGURATION of OpenSSH)

Secure Shell is one of the protocols that IT specialists use to ensure a secure and reliable connection between remote computer systems. This short guide explains a few things to make your SSH connection more secure.

What you will learn…

- How to configure OpenSSH.
- A few configuration options that may make your remote connections more secure, based on the OpenSSH.
- Good base to make up something new and secure on your own.

What you should know…

- Unix/Linux commands and SHELL environments.
- The basics of TCP/IP.
- Understanding the need for security.

For FreeBSD and OpenBSD, SSH starts automatically if you did not remove the check from the sshd box during installation.

OpenBSD (immediately):

```
# /etc/rc.d/sshd start
```

OpenBSD (permanently):

```
# vi /etc/rc.conf.local
```

References (order of relevance)

- man sshd_config
- man sshd
- *www.openssh.org*
- *www.openbsd.org*; *www.freebsd.org*
- www.rfc-editor.org and search for "SSH" phrase (for advanced users – programmers)

Note (only in code explanations):

Italics *word* means option.
Code *word/numbers* means value of that option.

Create the line used when starting sshd_flags and leave it empty like this `sshd _ flags=""` or you can specify your own path to the configuration file, so the line should look like `sshd _ flags="-f /MY _ PATH/my _ config _ filename"`.

Changes will take effect after restarting the system, or you can use the command line to do it immediately.

```
# /etc/rc.d/sshd start
```

or restart when sshd is running.

```
# /etc/rc.d/sshd restart
```

FreeBSD (immediately):

```
# /etc/rc.d/sshd start
```

or

```
# service sshd start
```

FreeBSD (permanently):

```
# vi /etc/rc.conf
```

Create the line started at sshd_enable="YES".

You can specify your own path to the configuration file. Edit the following file.

```
# vi /etc/rc.d/sshd
```

Find the line started at `command="/usr/sbin/${name}"` and change it as following.

```
command="/usr/sbin/${name} -f /MY_PATH/my_config_filename "
```

Changes will take effect after restarting the system, or you can use the command line to do it immediately.

```
# vi /etc/rc.d/sshd restart
```

Note
Do not specify your own configuration file when you run for the first time.

sshd configuration file explanation

The configuration file in this example was tested and operated correctly for FreeBSD 9.1 and OpenBSD 5.3.

Warning
AuthenticationMethods `publickey, password publickey, keyboard-interactive` does not work at FreeBSD (maybe in the next releases) and the value of the option `UsePrivilegeSeparation` sandbox works in OpenBSD. As for the rest `UsePrivilegeSeparation`, yes, it works.

Every time you change your configuration, you can check its validity by using the test mode, or extended test mode, as shown below respectively:

```
# sshd -t
# sshd -T
```

If you are not familiar with sshd, please use the following command to restart the process, or read the text from beginning. After installation of your *BSD system, sshd should be installed and run at start up by default.

```
# /etc/rc.d/sshd restart
```

You can experiment with the SSH options and values (in `sshd_config`) and you will not lose your current SSH connection(s) because the new sshd process serves only new connections. But if you close the current connection(s) and then you try to reach the server the new sshd process with the new configuration will be used.

Let's look at the sshd (ssh daemon) configuration file, bit by bit, from beginning to end. The file is always located at `/etc/ssh/sshd_config`, but it can be changed (moved or a new one created) if you wish (see PART 1).

Note
The commented lines are not listed if they are not necessary, so not all of your options have been displayed which you will see in your file.

sshd_config 1st part listing:

```
Port 4444
AddressFamily inet
ListenAddress 192.168.0.1
#ListenAddress ::
```

Port and then decimal number of the port, where sshd listens and waits for new connections (IP address + port number = socket). Standard port is `22`, so It is is recommended to change it for your own security, and select a range `<1024-65535>`. It reduces up to 90% of the sniffers that are searching for port `22` and then trying to log in using a dictionary attack, or brute force. Thus, it decreases the load on your server and reduces probability of access to your system.

AddressFamily and then `inet` for IPv4, `inet6` for IPv6 and `any` for both. If you do not use IP version 6, It is good practice to disable it, so only use `inet`. Please remember: if you do not need something, just do not use it and disable it.

ListenAddress and then an IPv4 decimal four octets IP address, such as `192.168.0.1`, where sshd listens and waits for new connections. Please do not use a locahost IP address (`127.0.0.1`), because you will never connect to your server from the outside (better security, but weaker functionality). Try to not use external (Internet) IP addresses due to attacks from the Internet network. Nevertheless, if you have to use an Internet IP address, later you will learn how to increase the security of your SSH.

#ListenAddress indicates that we do not want to use IPv6 (line commented) and do not want to assign IPv6 to `::`.

sshd_config 2nd part listing:

```
Protocol 2
# Lifetime and size of ephemeral version 1 server key
#KeyRegenerationInterval 1h
#ServerKeyBits 1024
SyslogFacility AUTH
LogLevel DEBUG3
```

Protocol and then the number `1` or `2`. The second one is newer, so use it and comment all lines indicated for 1 (look at the three commented rows above for example). This option selects the protocol version of SSH, and version `2` is more secure and has more features.

SyslogFacility and then `AUTH`, `DAEMON`, `USER` or others (see man page). It is for logging details. `AUTH` is sufficient for the security reason. The log file is `/var/log/auth.log` (absolute path). After logging in to your SSH, please list that file. You will learn more about the SSH connection and the security. It is is recommended to copy your log file to analyze it later in case of break in or other factors associated with sshd.

LogLevel and then depth of debugging level indicated by `INFO`, `ERROR`, `DEBUG3` (more at man page). `DEBUG3` is the most detailed logging level, so it is recommended for use for security analyzing or to make sure the process is working properly.

sshd_config 3rd part listing:

```
AllowUsers xyz007 backup John
AllowGroups wheel
LoginGraceTime 15
PermitRootLogin no
StrictModes yes
MaxAuthTries 3
MaxSessions 3
```

AllowUsers and then list of users separated with a space (almost always local users) `xyz007`, `backup`, `John`, or `root` (not recommended). The option gives the possibility to restrict connections to particular users. Try to use a non-standard user, different than admin, administrator, sql, etc., to hinder attackers. Note: `xyz007`, `backup` and `John` are not default users after an installation, so you have to create the new ones and add them to the group wheel.

AllowGroups and then list of groups to have an access into the SSH connection. Listing is the same as for users. Note: if the user is not listed in the *AllowUsers* option, but their group is listed in *AllowGroups* the user will not have the appropriate privileges to gain access into the SSH connection. Group `wheel` is the standard group for non-standard users, so if you even comment the *AllowGroups* user must be in the `wheel` group (just edit the file `/etc/group` and add the user to `wheel` group).

LoginGraceTime and then the number of seconds (i.e., `15`) to successful log in. If you exceed the limit of seconds, the server drops your connection to inactive and you have to start to log in again. Note: `60` is equivalent to `60s`, as well as `1m`.

PermitRootLogin and then one of the values `no`, `yes`, or *others* (see man page). The name of *PermitRootLogin* explains everything. Note: If the line is commented, that means that root log in is not allowed (recommended) to connect to SSH.

StrictModes and then one of the values `no` or `yes`. It is is recommended to set the value to `yes`, because sshd process checks the home directory and other user files permissions.

MaxAuthTries and then the number of tries to connect into the SSH connection. It is is recommended to set the value to less than `10`, (i.e., `3`). If an attacker tries to guess the password, or passphrase, then after 3 times of unsuccessful login attempts their connection is dropped.

MaxSessions and then the number (i.e., `3`) of simultaneous connections independent of users. So `xyz007`, `backup` and `John` all logged in at the same time would prevent others from logging in. For strict security reasons you can set the value to 1. Log in and keep the SSH connection as long as your network stability allows you to stay logged in (desktop locking, terminal locking and prevent others attack on your client).

sshd_config 4th part listing (keys generation and usage is explained at PART 2):

```
AuthenticationMethods publickey, password
   publickey,keyboard-interactive
RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile .ssh/authorized_keys
```

AuthenticationMethods and then list of authentication methods (i.e., `publickey`, `password publickey`, `keyboard-interactive`). It means that authentication will start step-by-step via `publickey`, after successful authentication via `publickey` it goes to `password publickey` and then at the end is `keyboard-interactive (user's password)`. You can use one of them, two or three in correct order (`publickey` before `password publickey` and `keyboard-interactive`), but `publickey` and `password publickey` are concatenated, so you have to use them together in that order.

RSAAuthentication and then one of the values `no` or `yes`. It is required to set `yes` when you set one of the authentication methods on public key and two below options as well.

PubkeyAuthentication and then one of the values `no` or `yes`. It is required to set `yes` if you want to use public/private key authentication.

AuthorizedKeysFile and then path to public key (i.e., .ssh/authorized_keys). Be informed that the path is not absolute but is relative and the root is user's home directory. Note: set the file permissions on 400 or u=r,g=-rwx,a=-rwx using command *chmod*, regardless public keys can be known for everyone.

Note: these four above options are the main pack of the security planks to keep safe your SSH connections safe.

sshd_config 5th part listing:

```
PasswordAuthentication yes
PermitEmptyPasswords no
```

PasswordAuthentication and then one of the values `no` or `yes`. It is required to set the value to `yes` due to using a user's password authentication method. Without `yes`, the SSH connection will be unsuccessful and then dropped.

PermitEmptyPasswords and then `no` or `yes`. For security reasons it is recommended to set the value to no. Note: no user from wheel group and other standard groups (your groups) should not have account without password. There are many non-standard (application) users without password, but shell/terminal logging in is denied or should be denied to them.

sshd_config 5th part listing:

```
#X11Forwarding no
#X11DisplayOffset 10
#X11UseLocalhost yes
```

In short: X11 should never be used in Unix, Unix-like, Linux systems until system is destined as a server. It decreases the system load and improves security. Professional and experienced OS terminal user is faster than OS window user.

sshd_config 6th part listing:

```
PrintMotd yes
PrintLastLog yes
```

PrintMotd and then one of the values `no` or `yes`. Option outputs the text from the file `/etc/motd` (absolute path). It is useful to inform unprivileged user about restrictions and other information before trying to log in. Try to devise your own text info.

PrintLastLog and then one of the values `no` or `yes`. It is recommended to set the value to `yes` to see your own log-in information and client IP address then to verify it was what was expected.

sshd_config 7th part listing:

```
UsePrivilegeSeparation sandbox
PermitUserEnvironment no
```

UsePrivilegeSeparation and then three values to set `no`, `yes` or `sandbox`. It is recommended to use `sandbox` or `yes`. Both values separate process and the sandbox does a jail environment. It is more secure to prevent escalation privilege due to code corruption, attack other host or kernel attack surface etc. Example process listing below (Listing 1) with comments started at `//`.

PermitUserEnvironment and then one of the values `no` or `yes`. It is recommended to set the value to `no` to prevent bypass access restrictions (see man page).

sshd_config 8th part listing:

```
ClientAliveInterval 60
ClientAliveCountMax 10
```

ClientAliveInterval and then value of seconds (i.e., 60) in conjunction with *ClientAliveCountMax* use encrypted channel to drop the SSH connection due to an inactive user through 60*10=600 seconds. It is good to use it, but *MaxSession* and keeping a session by one user will not work. *ClientAliveCountMax* and then value of multiple *ClientAliveInterval* (i.e., 10).

sshd_config 9th part listing:

```
MaxStartups 5:15:30
Banner /etc/ssh/motd
```

MaxStartups and then numbers X:Y:Z (i.e., 5:15:30) which mean randomly dropping the unauthenticated concurrent connections: 5 – the maximum of concurrent unauthenticated connections made example by attacker (alone value or with other as begin, see man page); 15 – base for ratio of probability (15/100)*100%=15%; 30 – the maximum of concurrent unauthenticated connections. Note: It is not the same as *MaxSessions*. *MaxSession* works after authentication and *MaxStartups* works before authentication. It is good to set it for more than *MaxSessions* due to zombie processes that could take the socket for new connections.

Listing 1. *ps aux command section*

```
USER       PID %CPU %MEM   VSZ   RSS TT  STAT  STARTED       TIME COMMAND
root     22784  0.0  0.3  1064  3108 ??  Ss    Thu05PM    0:52.46 sshd: John@ttyp0 (sshd) //UsePrivilegeSeparation=no
John      5754  0.0  0.3  3444  2724 ??   S      6:41PM    0:00.43 sshd: John@ttyp1 (sshd) //Parent process for sandbox
root     20080  0.7  0.3  3456  2820 ??  Ss     6:41PM    0:00.24 sshd: John [priv] (sshd) //UsePrivilegeSeparation=sandbox
```

Listing 2. *SSH successful connection screen shot*

```
Using username "John".


                      Access Restricted Equipment
                  All Activities are Monitored and Logged
                      Unauthorized Use Prohibited

    By Accessing, You Are Agree Your Activities to be Monitored and Logged

Authenticating with public key "imported-openssh-key"
Passphrase for key "imported-openssh-key":
Further authentication required
John@192.168.0.1's password:
Last login: Mon Oct 21 18:41:48 2013 from 192.168.0.18
OpenBSD 5.3 (GENERIC) #50: Tue Mar 12 18:35:23 MDT 2013

           #     #
           #     #  #    #     #    ######     #    ######  #####
           #     #  ##   #     #    #          #    #       #    #
           #     #  # #  #     #    #####      #    #####   #    #
           #     #  #  # #     #    #          #    #       #    #
           #     #  #   ##     #    #          #    #       #    #
            #####   #    #     #    #          #    ######  #####

           #######
              #     #    #  #####   ######    ##     #####   ####
              #     #    #  #    #  #        #  #      #    #
              #     ######  #    #  #####   #    #     #     ####
              #     #    #  #####   #       ######     #         #
              #     #    #  #   #   #       #    #     #    #    #
              #     #    #  #    #  ######  #    #     #     ####


#     #
##   ##    ##    #    #    ##     ####   ######  #    #  ######  #    #   #####
# # # #   #  #   ##   #   #  #   #    #  #       ##  ##  #       ##   #     #
#  #  #  #    #  # #  #  #    #  #       #####   # ## #  #####   # #  #     #
#     #  ######  #  # #  ######  #  ###  #       #    #  #       #  # #     #
#     #  #    #  #   ##  #    #  #    #  #       #    #  #       #   ##     #
#     #  #    #  #    #  #    #   ####   ######  #    #  ######  #    #     #

                        ###            ###    #####
                       #   #          #   #  #     #
                      #     #        #     #       #
                      #     #        #     #  #####
                 #    # #     #        #     #       #
                  #  #   #   #    ###    #   #  #     #
                    ##     ###     ###     ###    #####
```

```
                              MATRIX
--------------------------------------------------------------------------------
| 962451 | 498960 | 161272 | 198028 | 892602 | 862209 | 602676 | 188704 | 465660 |
| 589275 | 745029 | 843831 | 260712 | 757845 | 787293 | 731988 | 839178 | 885996 |
| 404109 | 932625 | 100494 | 892413 | 321237 | 840015 | 221832 | 861696 | 128539 |
| 119556 | 255357 | 637839 | 138754 | 298872 | 322758 | 665937 | 269883 | 104410 |
| 100629 | 747423 | 772029 | 494253 | 467937 | 743454 | 668178 | 543690 | 346527 |
| 294183 | 249246 | 587565 | 365796 | 469818 | 115669 | 125460 | 601956 | 776394 |
| 240741 | 709317 | 919638 | 549441 | 126622 | 192384 | 157798 | 597762 | 544014 |
| 352449 | 329364 | 754560 | 234324 | 531387 | 601101 | 142417 | 248202 | 780705 |
| 681732 | 975816 | 134973 | 788337 | 852849 | 783306 | 616518 | 431829 | 622215 |
--------------------------------------------------------------------------------

Unlock key:
```

Banner and then the absolute path to the text file (i.e., `/etc/ssh/motd`). Functionality the same that *PrintMod* has.

Please look at the screen shot (Listing 2) of the successful SSH connection. First the banner is *PrintMotd* and the second one, big is *Banner* (generated by banner application). There is an *AuthenticationMethods* order starting from "Authentication with public key with passphrase for that key" and then "user's password authentication". You can find logging in details at the `/var/log/auth.log` or at another file you set. You can see the last login details and system details as well. If you do not want to see/show the system info, just delete the line from `/etc/motd` file or change it to something else, (i.e., fake OS). You see MATRIX and Unlock key text also. This application is my own and you can try it by downloading from *www.iptrace.pl* (go to Download and click on Locker). Application is free of charge and based on the BSD License. Any suggestions and errors about the Locker please send via e-mail *locker@iptrace.pl*.

Keys generation and usage

We have to generate both private and public key pairs. Next, we have to set up our OpenSSH server and then client to use it.

The server side stores the public key because from the server side it is a "public" machine. Public means, other administrators (especially roots) have access to your home directory and is not advisable to keep the private key there. You are the owner of this key, so you have to keep it safe like the key to your house.

Let's start to generate the key pair with 4096 bits of RSA. It is better to generate the key pair logged in as the owner of the keys, because the default keys location will be proper and the owner of the files will appropriate.

```
# ssh-keygen -b 4096
```

The output should look like Listing 3. Note: You can define the passphrase to improve the security. It is referred to password publickey in the AuthenticationMethods option. Setting the passphrase is not strictly required, but it

Listing 3. *ssh-keygen command output*

```
Generating public/private rsa key pair.
Enter file in which to save the key (/home/John/.ssh/
   id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/John/.ssh/
   id_rsa.
Your public key has been saved in /home/John/.ssh/
   id_rsa.pub.
The key fingerprint is:
97:15:3f:a7:57:a5:c4:20:5f:69:12:f4:5a:aa:7c:16 John@
   server.iptrace.pl
The key's randomart image is:
+--[ RSA 4096]----+
|          ..=+o..|
|           o.*+..|
|            oo* o|
|           o + +.|
|        S o E . .|
|         o . . . |
|          o o    |
|           o     |
|                 |
+-----------------+
```

is a good security approach. Sometimes the passphrase is omitted when users login via other SSH clients, especially Unix/Linux systems, just for one command to speed up these procedures.

So we have two keys in the .ssh directory placed in user's home directory. The public key is named id_rsa.pub and the private key is named id_rsa. Rename the public key to *authorized_keys* (referred to AuthorizedKeysFile option) and change the permissions to 400. Changes on the server side have been completed.

Let's look closer at the client side configuration. At first, we have to copy data from the private key file to the standard, text file. The file data is shown in Listing 4, you will have more data for the same 4096 bits, but I have truncated it to look better.

The next step is to select the data and copy to the file placed in the client side, example named ssh_key.txt. Note: the client side in this instance is based on MS Windows, WinSCP and Putty.

Now we have to regenerate our private key for Putty recognition. For this case we use one of the WinSCP ap-

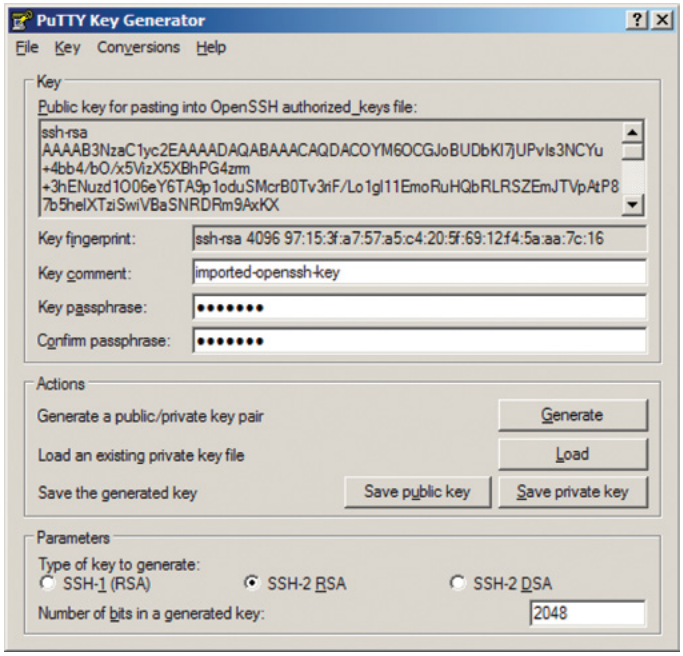


Figure 1. *Generating private key for Putty recognition*

Listing 4. *Listing private key data*

```
# cat $HOME/.ssh/id_rsa
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,FECD4B5E7A255C6C271B4E3935D71E73

/xjlpVJrsEiqOeZ/D1iGOKy+S4Oai8xs4aYSIpjO9Y8yRgGmNIG9spylRNAYAVG+
+qJm/Fme/x5brVCINl4yqDe9nBei4UQrWTCPm9fZYErOERBW7WQJxgFLa2eWLl2H
iFaEXnkUViyAVNFtQvdC29njKkaIBt/SAL8NxWjU8ECyzDPtc6e+dOgku7/blmM4
0KikRA40JkkDQbibyDbq1TeYmePyQ0VoPs41YhOHx6fEfGaJ2jn9o3Uc8KtQdzmI
IwMuZ2josH6/rM2uDrowQJ47nZ6epgASKfAbrE8xhZfczrgd8EWHWZZ7JrwDYYmu
B8MP/E75ScwC6O8VHwtSLh0K33DEWc3pJvP9KGqlp84rG1r9UP4TqkGL88f1ChY2
Y3kEqOQm6C5h3bFVPd8ByObebS0yQoEk8AqfT9gRS5S2bbG5K5rQstVSCTC87AZP
H7l2ZYvmdeWzqpdSePjEv/RgJHP74lT5dCvUDjQk2JmhJW8m/zp9VEBCskFB/aMy
b6Axk+ZoXsZHJ5pq7AAmBvQBlvXDNx0SyQyPmIZpFL8KlEUPTsBc5PPIghQK4HAf
V5ZnL8BarLsQ3GRmtgMwjeadwty2RRu7XiMGApsBgQleC4rBd2KyDSDvvB49vKQ9
6cNpEJHBC9MKjl/5txZ9KiBshz9kYttHvGq1a93DzZjIEyp26smV4TrFlTSUvfDS
Uq91D22so631+Uc+P+wA8h7oR5+1qctfrxH3uhwf111xlDwhayPwSp2WgLT/d55n
xL1j0rs4XsIYX0Zer/RXgJmvcYcpKtl+MPBdHJcb7tHqFHJYn0klwzdX2QI2ymAD
lJ4kMC13J2HikFn5JoeGNuw0NyfVburqtE5SYdAsLSCyDkq9iQZDgrA23Uilb/x+
cb7UbqtsoJvlThOemTByBOgGGwt6n0lP3xFl0v+e0OHCC7/Kj04q6m/lUeH4ILHV
tUaQk41de6bdKsYzomsh50Ko6KX07teOQVHCZVOiri4ao4UEWo1ud+5KzOcpvVCC
Fof/N2JcVZL9sAgAcpYoEbZVV7hR8KWrzQuxSY32Rrwb1qac2nC+1ThdXTS+LJe6
ekPpqUz96F5DF7vDqLI3mEuLKDZ9ie+MilmgDTbyNWMaRvK3ZRbVf1DZuahLfCIu
NlLIqNtkenzxa0/FMOx89LtE9NIpnJqPD8KRr+B6x/NaYmagUCHamAcvF0KqSwOg
rBf7piUpxQTJvCoYHppAxGQB5j8UhIzLSe3nxaHq0rxGzTyoAy6FRpB1BnlLi21e
-----END RSA PRIVATE KEY-----
```

ps. It names PuTTYgen. So, run the application and click on the Load, to load your private key. During loading the file, the application asks for a passphrase. When complete (Figure 1), you will see the following window. Click *Save* for the private key using a ppk extension.

At the end of our tour, we have to configure Putty to use our new private key. Run Putty and then go to category Connection->SSH>Auth then you will see the field Private key file authentication, click Browse and get your new private key file. Insert your server IP address and other features and save your session for future use.

Conclusions

Look at man pages of sshd and sshd_config to learn more about other interesting options for SSH connections.

SSH is a great and a rich protocol and can be used not only for SSH connections (terminal connections), but for files transfer, known as an SFTP, or for VPNs tunneling. The OpenSSH configuration works great for SFTP connections using mentioned WinSCP application. WinSCP is easy and similar to Putty configuration.

You shall find or devise a lot of authentication methods, but one that is interesting, is known as one time password OTP. You can find more information about it by searching the Internet. Try to use it in conjunction with e-mail, SMS or token. Good Luck!

In the next series you will find out about:

OTP – one time password to beef SSH connections up.
VPN tunnelling – creating Virtual Private Networks using OpenSSH
SFTP – known as SSH File Transfer Protocol to opposite of a standard FTP

ARKADIUSZ MAJEWSKI, BENG

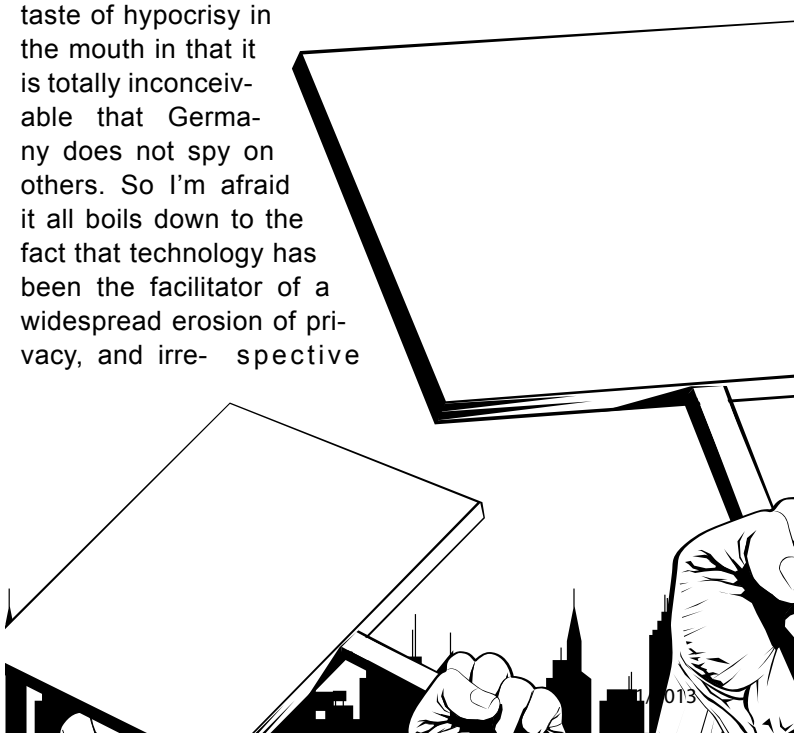
Arkadiusz Majewski comes from Poland. He has 15 years experience with ICT technologies, including 15 years of IT networks, 10 years of BSD systems and MS Windows Server solutions. He also has 5 years experience with programming languages and Telco solutions. He is interested in security on all business and ICT levels. In his free time he reads ICT books, deepens his knowledge about science (math, physics, chemistry). His hobby is cycling and motorization. He is a graduate of Warsaw Information Technology, under the auspices of the Polish Academy of Sciences.

Feel free to contact the author via e-mail: bsd.magazine@iptrace.pl.

With the Recent Revelation That the United States Spied on Angela Merkel and the Subsequent Outrage From Politicians – is this a Case of the "Lady Doth Protest Too Much"?

Here in the United Kingdom, we are well ahead of the game that the USA, the German leader, Angela Merkel, and the media is currently playing the game of faux outrage and hurt feelings over the surveillance of telephone conversations by the USA of 33 world leaders. After all, the scandal which has been simmering away for the past few years concerning the hacking of phone messages by News International culminating with the Leveson enquiry (and the subsequent recommendations that press freedom be legally curtailed) is not latest news. While quite rightly there was outrage that the media used such underhand tactics against the general public, politicians, celebrities and even the Royal Family, the other side of the argument has had scant coverage – that we live in a very different society from Victorian times.

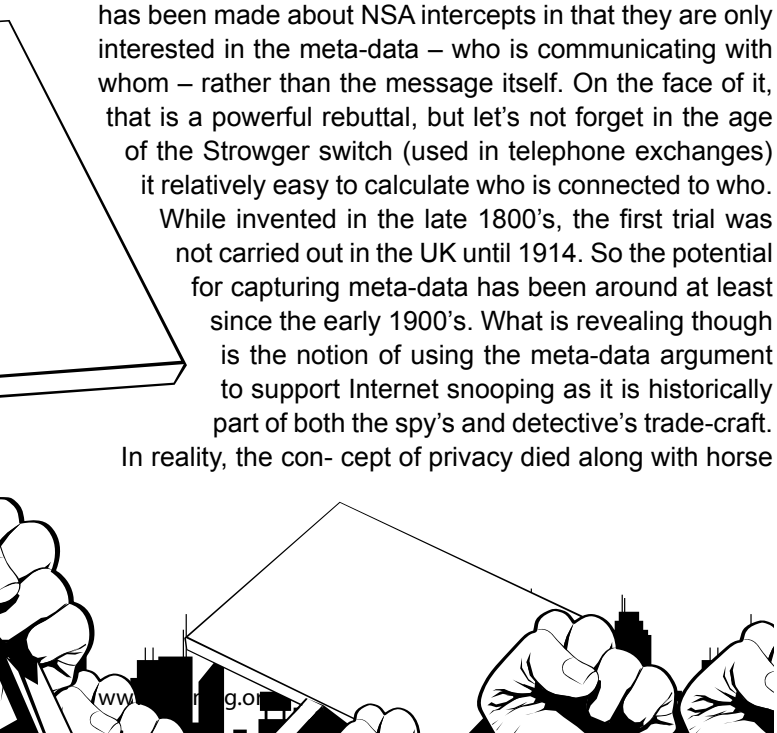
You might be shocked at my blatant anchoring of loss of privacy values prior to the publication of George Orwell's masterpiece 1984 in the post war years, but please bear with me. Casting aside the fact that Government, Kings and the Church have used spies throughout history, the Victorian age was the last epoch where the average man, woman or child could be guaranteed a relatively strong sense of privacy. That is why I must take Angela Merkel's "outrage" with a pinch of salt the size of a Siberian salt mine. Any politician who is unaware of the historical precedent of dirty dealings between warring states (or indeed allies) is either naive, uneducated or deceived, and even more so in the case of Angela Merkel who must be aware of the worst abuses of government power that took place over the border in East Germany by the Stasi during the Cold War. If I was to be generous, I'd say the outrage is driven by that historical fact but that leaves the unpleasant taste of hypocrisy in the mouth in that it is totally inconceivable that Germany does not spy on others. So I'm afraid it all boils down to the fact that technology has been the facilitator of a widespread erosion of privacy, and irre- spective



of political hue or ideal, any power will use that lever to their advantage where at all possible.

Let's get back to the Victorians. To send a letter (or to communicate) the letter was sealed, delivered and the recipient would break the seal and read the communication. Like the well proven process used by the Roman General, the unbroken seal on the scroll was the guarantee of authenticity, unless of course the senders credentials had been compromised. Forgery will always be the weak link in the chain, proving the identity of an individual with 100% accuracy always elusive, as the immoral can always get round this provided they have enough resource. The question "Who am I" extends past philosophical debate into real life wherever the identity of a person requires confirmation. So we can assume that privacy was reasonably well guaranteed unless there was sufficient reason to commit the offence of intercepting the Queens Mail, an offence that carried the death penalty.

From the 1900's up until the Second World War, the widespread adoption of the telegraph, telephone and radio communications muddied the field and this is where the root of the issue lies. The more people that are involved with the transmission of your message, the more open the transport medium used, the greater the coverage and penetration, the greater the chance your message is no longer private. Then came the Second World War. The necessity for documenting citizens became paramount in the interests of national security, and in the UK the adoption of the National Health Service and welfare state allowed a huge paper bank to develop of the characteristics of the general population. While crude, this database is the basis of the current conundrum – who can we trust to be the guardians of confidential data? Much has been made about NSA intercepts in that they are only interested in the meta-data – who is communicating with whom – rather than the message itself. On the face of it, that is a powerful rebuttal, but let's not forget in the age of the Strowger switch (used in telephone exchanges) it relatively easy to calculate who is connected to who. While invented in the late 1800's, the first trial was not carried out in the UK until 1914. So the potential for capturing meta-data has been around at least since the early 1900's. What is revealing though is the notion of using the meta-data argument to support Internet snooping as it is historically part of both the spy's and detective's trade-craft. In reality, the con- cept of privacy died along with horse and cart. Whether it be multi-nationals gathering marketing information, software manufacturers knowing your physical location or government reading your email, there are too many areas to cover. Sadly, by the time Orwell's book was published, the cracks had already started in the edifice of personal privacy, and while the Zeitgeist has been looking at the physical manifestations society wide, those that have the means, ability and reason to monitor have been increasing their power base under the radar (and I am being generous here) since the Second World War. What is truly telling though is the penalty for intercepting communications. In the Victorian age, the penalty was death. Today, a hefty fine and maybe a prison sentence will be your fate, despite the much greater opportunity for abuse. So I totally agree with the editor of the British satirical magazine Private Eye, Ian Hislop, that we have enough legislation as it is to combat any excesses of power, but the fact remains that there is one law for "them" and one law for "us". Angela Merkel has managed to capture the ear of the US President, and I suspect while an apology will be made and steps taken by the German government to improve domestic security, this is no comfort for the man in the street. The only comfort I can see is that while the USA has probably the most advanced spying infrastructure in the world today, at least this is offset to a small degree by a Freedom of Information culture that has teeth, and a legal system that is not afraid to sue even government. Other countries are not so fortunate. Angela Merkel as a world leader has an intelligence service and a legislature at her beck and call. No doubt there will be some major changes to European legislation on the back of this revelation. But this is all smoke and mirrors. Unless people truly appreciate we no longer live in the age of privacy, and the full weight of the law executed without bias to protect individuals, corporates and governments from abusive governments, corporates and individuals, we are all going to remain vulnerable, paranoid and an easy target. The concept of "who watches the watchers" has never been more apt.



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.

Maximising Website Runtime on Host Servers Running FreeBSD

We've all seen the damage caused by web traffic spikes. During this year's Super Bowl, the websites of 13 companies that advertised during the match went down within five minutes of the advert airing. With advertising slots being sold for up to $5,840,000 (£3.6m, €4.3), and run to drive traffic, that's one (well, 13) costly website failure.

Another very high profile crash came just last month, when the US's new healthcare insurance programme, Obamacare, launched. The exceptional traffic and various bottlenecks took the website out almost instantly.

Over 2 per cent of the world's websites run on BSD (roughly 14 million websites). And a quick Google and forum search highlights that OpenBSD users are not immune to this problem. Therefore traffic spikes need to be part of any business continuity plan when working with BSD or any other Unix variant. But it's not only website traffic spikes, as we move to cloud services we add extra dimensions to business continuity planning and we have to look at disaster striking when a major cable is cut – or a myriad of other major system failures.

Business continuity planning

There are two key metrics used by industry to evaluate available disaster recovery (DR) solutions. These are called recovery point objective (RPO) and recovery time

Figure 1. *Coca Cola's traffic spike following it's Superbowl Commercial – credit Yotta.net*

objective (RTO). A typical response to DR is to have a primary site and a DR site, where data is replicated from the primary to the DR site at a certain interval.

RPO is the amount of data lost in a disaster (such as the failure of a server or data center). This depends on the backup or replication frequency, since the worst-case is that the disaster occurs just before the next scheduled replication occurs.

RTO defines the amount of time it takes an organization to react to a disaster (whether automatically or manually; typically there will be at least some manual element such as changing IP addresses), performing the reconfiguration necessary to recreate the primary site at the DR site. For example if there is a fire at the primary site you would need to order new hardware and re-provision your servers from backups. For most web hosting companies retaining an exact replica of every server at the primary site at the DR site is not economically viable.

Scaling Websites When There Are Spikes In Traffic

The traditional model

In the common shared hosting model, a web hosting company will install a lot of websites on a single server without any high availability (HA) or redundancy, and set up a nightly backup via rsync.

In this model when a website gets very popular, the server which is hosting it is also busy serving requests for a lot of other websites and becomes over-loaded. Typical consequences of this are that the server will start to respond very slowly as the required number of I/O operations per second exceeds the capacity of the server. The server will soon run out of memory as the web requests stack up, start swapping to disk, and “thrash itself to death”. This results in everyone’s websites going offline.

The CloudLinux model

An alternative is the CloudLinux model, which contains the spike of traffic by imposing OS-level restrictions on the site experiencing heavy traffic. This is an improvement because the other sites on the server stay online.

However the disadvantage to this approach is that the website which is gaining the traffic is necessarily slowed down or stopped completely. If the server were to try to fully service all incoming requests for that site, it would crash, as above.

A better model

Rather than strangling the site experiencing the spike in traffic, it is possible to dynamically live-migrate the server’s other websites on that server to other servers in the

cluster. This eradicates downtime on any of the cluster's sites, and therefore allows a host to enable full and automatic scalability.

Using this method it's possible to deliver three (or even four) orders of magnitude greater scalability than shared hosting solutions – assuming just 500 websites per server, you can burst to 2 dedicated servers or 1,000x scalability – allowing websites to scale by intelligently and transparently migrating them between hosts.

This is the method used for HybridCluster's Site Juggler Live Migration.

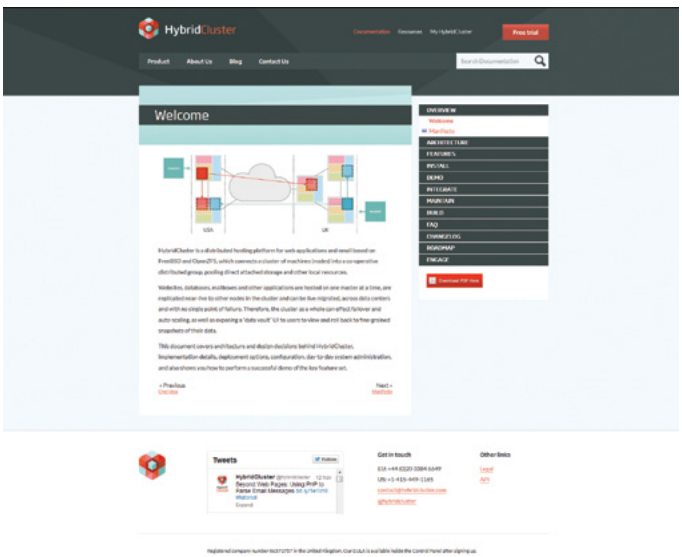
Keeping you online in a disaster

When building distributed (cloud) systems we're faced with three tradeoffs:

- Consistency – if a system is consistent, then queries to different nodes for the same data will always result in the same answer
- Availability – the system always responds to requests with a valid response
- Partition tolerance – if the parts of a distributed system become disconnected from each other they can continue to operate

In reality you can select two and these should be availability and partition tolerance over consistency. Doing this allows the website to stay online in a disaster scenario, for example if an under-sea cable gets cut, and the European and US components of a cluster can no longer communicate with each other.

If we look at a typical web request, for example a user uploading a photo to a WordPress blog, we can see content is backed up on multiple continents to prevent loss in a disaster (natural or man made).



But once disaster strikes it is essential to then elect new masters for all the sites on both sides of the partition in order to keep the websites online on both sides of the Atlantic; something we worked into the HybridCluster protocols.

Using this approach, when traffic is re-routed or the under-sea cable is repaired, the cluster can also rejoin and the masters negotiate which version of the website is more valuable based on how many changes have been made on both sides of the partition. This keeps your websites online all the time, everywhere in the world.

Summary

Regardless of whether you're running a huge multinational organization or an open source software site, downtime is costly.

New techniques need to be applied to both cope with increased demands and, as we shift to cloud computing, factor in disaster. Intelligent handling of data to move sites between clusters and through the automatic assigning of parent clusters after a partition is formed… and merge them again once it is fixed. By using an integrated suite of storage, replication and clustering technologies – such as HybridCluster – it's possible to shift to a true cloud computing model and enable intelligent auto-scaling, as well as integrated backup and recovery.

About HybridCluster

HybridCluster has triggered a rethink about cloud and hosting industry's dependency on high cost, legacy virtualisation and storage stacks that fail to fully protect both businesses and end users. Computer scientists and industry experts have combined at HybridCluster to deliver breakthrough storage and hosting platform technology that automatically detects and recovers data centre outages in less than one minute, delivers 4x better density of customers per server, and offers end user to self-recover lost files and data. (www.HybridCluster.com / @HybridCluster)

LUKE MARSDEN
CEO HybridCluster
www.hybridcluster.com

WHD.local 2013

THE MOST EXCLUSIVE EUROPEAN HOSTING AND CLOUD EVENT

Register now for free!

Register now for free at www.worldhostingdays.com/local
Promo code: LMUJR99

PGDay.IT 2013

Another year, another PGDay.IT! The seventh edition of the main Italian conference fully dedicated to PostgreSQL, the world's most advanced Open Source database, is over and ITPUG, the Italian PostgreSQL User's Group, did a really great job in organizing this event.

PGDay.IT is a well known event in both the Italian and international PostgreSQL communities. Thanks to the excellent work of ITPUG, the event has kept growing year after year and is today the main Italian conference fully dedicated to PostgreSQL.

But PGDay.IT is not *just a conference*, it is a party – the party of the PostgreSQL community and a party about Open Source. And as it was for the previous editions, the party started with a good dinner based on the famous *Fiorentina* steak, where attendees can meet and relax together.

The Conference

PGDay.IT took place on October the 25th in Prato, Tuscany, in the great Vaj Palace of the Monash University. The conference schedule was very rich, including 9 different speakers and 13 regular talks in two parallel sessions. Due to the high number of attendees and the need to register all of them, the conference started with a little delay that organizers were later able to make up, ensuring the conference schedule was not compromised.

The keynote talk was given by Mr. Bruce Momjian, a PostgreSQL Core Member and very active developer. The talk was very interesting and enlightening about the Open Source world and the PostgreSQL community. After a short coffee break, it was time for the technical part of the conference to begin. The Sala Veneziana was focused on the development aspects tied to PostgreSQL, for instance, the usage of the GIS extension, the JSON

formats, the unit testing of SQL pieces of code, and so on. The Salone Grollo was focused on the administration of a database cluster and on the new features of PostgreSQL, like for instance updatable views, foreign data wrappers, etc. There was also space for a few short talks about the widespread use of PostgreSQL in different projects, from education to health-care contexts.

Unlike previous editions of the conference, the whole day was not comprised of two parallel sessions full of regular talks. In the afternoon, a session continued with regular talks while the other was dedicated to the first ITPUG Lab, an interactive session discussed in more detail later.

At the end of the day, all attendees participated in the *Lightning Talks* plenary session. The rules for this session were quite simple:

- everyone can do a speech;
- a speech can be on whatever subject (but possibly related to PostgreSQL);
- each speech can be no more than 5 minutes.

The Lightning Talk session is a well established tradition in PostgreSQL related events, and PGDay.IT has had one at pretty much all editions of the conference. The aim of this session is to make attendees feel like part of the community, giving them the opportunity and, to some extent, forcing them to talk about their experiences, opinions, small or big projects and so on. And as in the previous editions, participants were happy to propose to the audience their own utilities, projects and use cases related to PostgreSQL.

A full day of PostgreSQL and technology surrounding it! You could have been walking around anywhere and found enthusiasts and professionals exchanging tips and tricks, experiences and opinions about the software they love. It did not matter if it was a coffee break, lunch, or a talk break: everyone was looking for hints and new things to learn. This is the real aim of ITPUG and PGDay.IT: allowing and easing the experience/knowledge sharing and community aggregation.

The day closed with a group picture and a lottery for ten one-year subscriptions to a developers' magazine, kindly offered by one of the event sponsors.

Organization

The organization of PGDay.IT took almost 6 months, in order to fully close the conference and 7 months work is a more appropriate evaluation. This year the ITPUG boards of directors changed, and therefore there were some initial difficulties to organize the work. Eventually, ITPUG got the right momentum and was able to deliver a very high quality event. It is worth noting that, for the first time, ITPUG took a clear approach at the organization tracking down each single activity (see Box 2), and this represents a very important value. Not only because there is a clear history about what was done, who did the work, which skills were required, and how much time and resources a single piece of the whole picture took, but also because it eases the migration of knowledge across the ITPUG members themselves.

A More Accessible Event

This year ITPUG made an event more accessible to everyone. In particular, two aspects were really important in the organization of the event: (I) define special fees and discounts depending on the attendee professional context and (ii) provide live streaming. The former allowed, for instance, university and/or high school students to approach the event and get in touch with the PostgreSQL community, while the latter delivered part of the conference contents to those who were unable to physically attend the event. In particular, PGDay.IT 2013 has been the first edition of the conference with live streaming.

The streaming was provided by one of the event sponsors and covered half of the conference (all of the Salone Grollo speeches). ITPUG believes that live streaming was important not only to deliver conference contents, but also to make known the high quality of the event, and therefore the good work of the ITPUG community.

PGDay.IT and Other Communities

This edition was also the first that has included different communities. First of all, the day before PGDay.IT there was another Computer Science event: the first OpenERPDay, organized at the Monash too. While OpenERP and PostgreSQL are really different projects, everyone who deals with an ERP application has to deal with a data-

Box 1: PGDay.IT 2013 by numbers

Numbers do not express the quality of an event very well, but in order to give to the readers an estimate of the size of the PGDay.IT, the following are some interesting statistics:

- 91 registered attendees (including staff members)
- 13 regular talks
- 9 and one half hours of content
- 9 speakers
- 8 sponsors
- 8 staff members
- 6 free-of-charge patronages
- 3 rooms
- 2 parallel sessions
- 1 lab session

Box 2: History of the event: more than ever!

One important aspect of PGDay.IT is the way it has been organized. ITPUG committed to the usage of an issue tracking system for all the activities tied and related to the event itself. This not only provides a history of the event more reliable and classified than a *simple* mailbox or *plain* wiki, but also allows ITPUG to literally clone the event for future arrangement. Of course, the aim of ITPUG and the meaning of PGDay.IT is not to stay the same year after year, and therefore this is unlikely to happen, but the important aspect of all this tracking activity is the detailed *knowledge* about required skills, times, deliveries, cross-dependencies and so on that each single piece of the puzzle requires. ITPUG strongly believes this knowledge and methodology will help organization of future events and will ease the joining of new forces into the staff.

base, and this explains why some attendees of the OpenERPDay event participated also in the PGDay.IT one.

But at the PGDay.IT there was also the advertising of BSD related products, most notably PC-BSD and FreeNAS, and this has shown how related these communities are to the PostgreSQL one. In other words, there are a lot of professionals that are using BSD on their development or production machines and manage large amounts of data using PostgreSQL.

A Social Event

As a tradition, once the event was over attendees and organizers met at a local pub to drink some great beer and relax together. The beer was kindly offered by one of the event sponsors.

The day after the event, for those who were spending the whole week-end in Prato, ITPUG advertised a few cultural events.

ITPUG strongly believes in social events, because they are the easiest way to make your professional network grow. More importantly, they are fun!

Box 3: Who Comes to PGDay.IT?

Our average attendee is a computer science professional with a clear interest in the database area, that already uses some kind of enterprise level database (possibly PostgreSQL) and wants to know more about PostgreSQL. Of course, PGDay.IT is open to everyone without any regard to their knowledge or skills, and in fact ITPUG worked hard to ensure the conference schedule included talks of different levels.

This year there were a good number of professionals coming from Italian Public Administration (local governments, research institutes, universities, and so on), a good thing that means that PostgreSQL is not only used by private professionals, but is becoming more interesting also for governments.

On the Web

- Italian PostgreSQL Users' Group (ITPUG) official website: *http://www.itpug.org*
- PGDay.IT 2013 official website: *http://2013.pgday.it*
- PostgreSQL official website: *http://www.postgresql.org*

The ITPUG Lab

ITPUG believes that the *evolution* of the PGDay.IT has to include a laboratory session, but not an ordinary one: the ITPUG Lab was indeed derived from the Open Space Technology (OST). This is not a new approach at all, but it is new in the database scenario, and judging by the success of the session, ITPUG strongly believes it has to be improved and introduced into other PostgreSQL-related events.

The ITPUG Lab was organized in a separate room and lasted exactly two hours. Attendees were invited to bring their own computers, even if there were a couple of extra computers made available by the staff. In order to keep attendees together and promote the spontaneous generation of *working teams*, tables were appropriately arranged and network connections were provided only by wire.

The session started with a brief presentation of all attendees, and then everyone was invited to propose a specific subject (of course tied to the PostgreSQL world) and to write it on a shared whiteboard. Once subjects had been proposed, interested people started gathering in small teams (up to 5 members) to work on a specific subject. For instance, there was a team dedicated to installation, one to the local monitoring of the health of a database, one to remote monitoring and one to replication. Attendees were free to move from one team to another depending on their interest, capabilities, and team needs.

Almost half of the time there was a database fully up and running, with two teams working on it in order to respectively monitor it locally and remotely, and later on a third team was trying to replicate the database.

After two hours the session ended, and a quick summary of the experience was collected. All the participants reported full satisfaction with the laboratory, asking in particular to organize a longer one.

LUCA FERRARI

Luca Ferrari lives in Italy with his beautiful wife and son. He is an Adjunct Professor at Nipissing University, Canada, a co-founder and the 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.

NET OPEN SERVICES IS AN APPLICATION HOSTING COMPANY FOCUSED ON OPEN SOURCE APPLICATIONS MANAGEMENT IN HIGH AVAILABILITY ENVIRONMENT.

NET OPEN SERVICES IS PROUD TO PROVIDE A HIGH QUALITY SERVICE TO OUR CUSTOMERS SINCE 10 YEARS.

OUR EXPERTISE INCLUDES:

- **CLOUD COMPUTING, PUBLIC, PRIVATE AND HYBRID CLOUD MANAGEMENT (OPENSTACK, CLOUDSTACK, RED HAT ENTERPRISE VIRTUALIZATION)**
- **REMOTE MONITORING AND MANAGEMENT 24/7**
- **NETWORKING AND SECURITY (OPEN BSD, IP TABLE, CHECKPOINT, CISCO,...)**
- **OS AND APPLICATION MANAGEMENT (FREE BSD, OPEN BSD, SOLARIS, UNIX, LINUX, AIX, MS WINDOWS)**
- **DATABASE MANAGEMENT (ORACLE, MYSQL, CASSANDRA, NOSQL, MS SQL, SYBASE...)**
- **MANAGED HOSTING IN CARRIER CLASS DATA CENTERS**
- **DISASTER RECOVERY**

Net Open
SERVICES

WE PROVIDE SERVICES IN EVERY STEP OF THE PROJECT LIFE, DESIGN, DEPLOYMENT, MANAGEMENT AND EVOLUTIONS. **NETOPENSERVICES** TEAM INCLUDES EXPERIENCED LEADERS AND ENGINEERS IN THE INTERNET SERVER INDUSTRY.

OUR TEAM HAS **15 YEARS OF EXPERIENCE** IN DEVELOPING INTERNET INFRASTRUCTURE-GRADE SOLUTIONS AND PROVISIONING INTERNET DATACENTERS AND GLOBAL SERVICE NETWORKS TOGETHER.

WE OFFER EXCEPTIONAL HARDWARE SUPPORT AS SOFTWARE SUPPORT ON UNIX/LINUX AND OPEN SOURCE APPLICATION. **NETOPENSERVICES** DELIVERS THESE CUSTOM-BUILT LINUX AND UNIX SERVERS, AS WELL AS PRECONFIGURED SERVERS AND SCALABLE STORAGE SOLUTIONS, TO OUR CUSTOMERS. WE ALSO OFFER CUSTOM DEVELOPMENT AND ADVANCED-LEVEL UNIX/LINUX CONSULTING SOLUTIONS.

WWW.NETOPENSERVICES.COM • CONTACT@NETOPENSERVICES.COM

Headquarters:
San Jose, CA

855.GREP.4.IX | Contact Us

99% Compatibility

online now...

IXSYSTEMS AND YOU ARE THE PERFECT MATCH

SHARED INTERESTS

- [x] Enterprise Storage Solutions
- [x] Personalized Customer Service
- [x] Bold New Information Technology

I'm a: Storage Reseller

In: The EU

Looking for: Storage Solutions to Sell

- A Technology Partner
- More Technical Experience
- New Business Opportunities

Visit Today!

iXsystems

Technology Partner Seeking Resellers/Integrators for TrueNAS™ Storage Appliance

WWW.IXSYSTEMS.COM/PERFECTMATCH