# HS - February 2015

How to secure a web server
~~like a boss~~ “comme un patron”

par Martin Dubé

# Tonight’s plan

- Security 5 Ws
- Scenario
- Strategy
  - Approach
  - Choosing an Operating System
  - Choosing a web server
  - Choosing a network topology
  - Choosing an architecture
- Implementation
- Test security of the solution (penetration test)

# About me

- Daddy
- Hackfest admin since 2011
  - Worked hard on Hacking Games
  - All other kind of tasks
- Security Analyst at GoSecure
  - Penetration Testing
  - Systems Hardening
  - Firewall Management (Checkpoint & Fortinet)

# [Security] What?

- From OSSTMM
  - “a form of protection where a separation is created between the assets and the threat.”
- Assets
  - The web application
  - The infrastructure around
- Threat
  - Outsiders (Those who access the web site)
  - Insiders (Those who manage the web site)

# [Security] Who?

- Everyone.

- Example:
  - Should the developer put time on security?
  - Should the sysadmin put time on security?

# [Security] Where?

- On all available layers

- Example:
  - Operating System (OS)?
  - Network?
  - Application vs Infrastructure
  - Client side?

# [Security] When?

- Until it is considered enough
  - Depend on risk acceptance
  - Depend on budgets

- Depends on the criticality of the assets and interest for the threat
  - An extranet
  - A public web site
  - An internal web site
  - A VPN web portal

# [Security] Why?

- Protect assets against Threats

# [Security] How?

- The following shall give some ideas.

# Scenario

- We are a system admin
- We need to integrate a web app in the infrastructure
    - It is extremely insecure (let’s say we know it)
    - It is extremely critical for the company
        - Do a shit load of things
        - Must work flawless

# Scenario

- The need is clear. We know that:
  - The app is coded in PHP
  - The app must run shell commands (dafuq?)
  - The app must access several servers outside the network

# Scenario

We won’t waste time on:

- Performance
  - Hardcore security sometimes means performance cost. Here we don’t care.

# Strategy

Securing by component

- OS
- Web server
- PHP
- ~~Database~~
- ~~Application arch.~~
- ~~Application source~~

# Strategy

For each component

- Ask about their communications (in / out)
- Ask about DIC
  - Authenticity
  - Integrity
  - Disponibility
- Implement relevant security mechanism
  - Fuck Security by Obscurity.
- Understand what the hell you do.

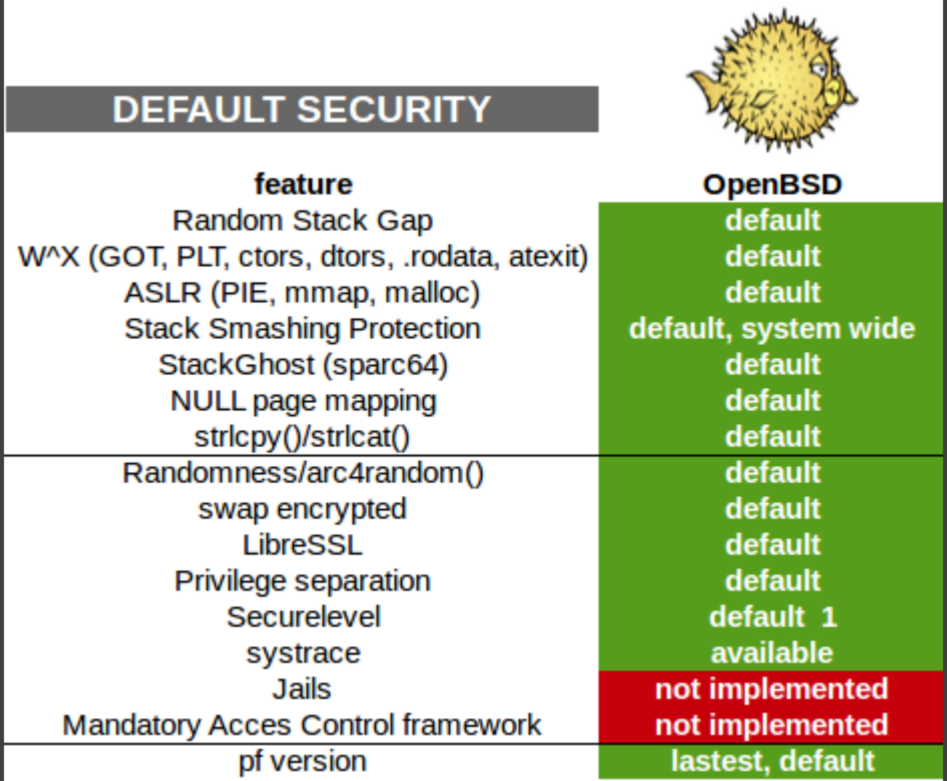
# [Strategy] Choosing a secure OS

- We want an OS that
  - Offer multiple relevant security mechanism
  - Is supported by an active community
    - Quick security fix
  - Is well documented

# [Strategy] Choice: OpenBSD

- Why?
  - Is secure by default
    - “Four years without a remote hole in the default install!”
  - Full of security mechanisms
    - Memory Protection (W^X, ProPolice, strlcpy/strlcat)
    - chroots (by default on some packages)
  - Privilege separation, by default (51 user, 39 for low priv.)
  - Powerful randomness (arc4random, libressl)
  - Regular source code audits (6 to 12 members security team)
  - Quick security updates
  - No damn /proc :)

# [Strategy] Choice: OpenBSD



DEFAULT SECURITY

| feature | OpenBSD |
|---|---|
| Random Stack Gap | default |
| W^X (GOT, PLT, ctors, dtors, .rodata, atexit) | default |
| ASLR (PIE, mmap, malloc) | default |
| Stack Smashing Protection | default, system wide |
| StackGhost (sparc64) | default |
| NULL page mapping | default |
| strlcpy()/strlcat() | default |
| Randomness/arc4random() | default |
| swap encrypted | default |
| LibreSSL | default |
| Privilege separation | default |
| Securelevel | default 1 |
| systrace | available |
| Jails | not implemented |
| Mandatory Acces Control framework | not implemented |
| pf version | lastest, default |

Thanks to: http://networkfilter.blogspot.ca/2014/12/security-openbsd-vs-freebsd.html

# [Strategy] Choice: OpenBSD

- Securelevel: 1 (default)
  - - /dev/mem and /dev/kmem may not be written to raw disk devices of mounted file systems are read-only
  - - system immutable and append-only file flags may not be removed
  - - kernel modules may not be loaded or unloaded
  - - a panic or trap cannot be forced

# [Strategy] Choice: OpenBSD

- Securelevel 2: "Highly secure mode"
  - all effects of securelevel 1
  - raw disk devices are always read-only whether mounted or not
  - settimeofday(2) and clock_settime(2) may not set the time backwards or close to overflow
  - firewall and NAT rules may not be altered

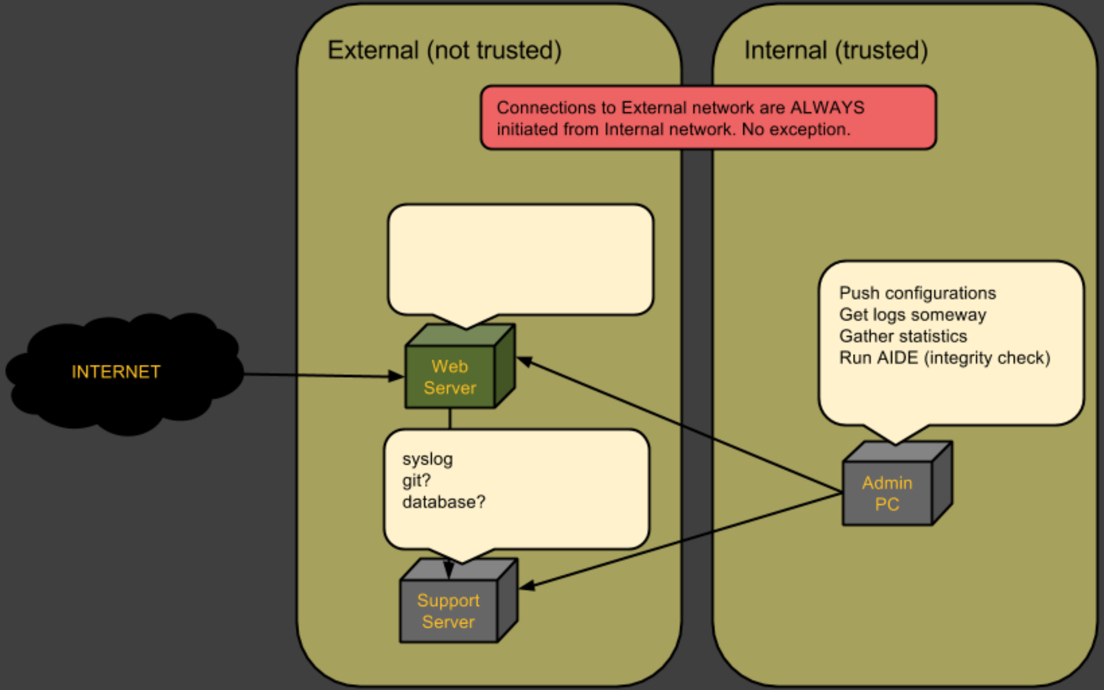
# [Strategy] Choosing a web server

- Nginx vs Apache
- Both are mature and well documented
  - Nginx pros
    - Chrooted by default
    - Less vulnerabilities documented (8 on nginx vs 284 on apache @ cvedetails)
  - Apache pros
    - mod_security
    - .htaccess
- Understanding how to use it is what matter!

# [Strategy] Choice: nginx

- Why?
  - For previously enumerated reasons.

# [Strategy] Choosing a topology



External (not trusted)

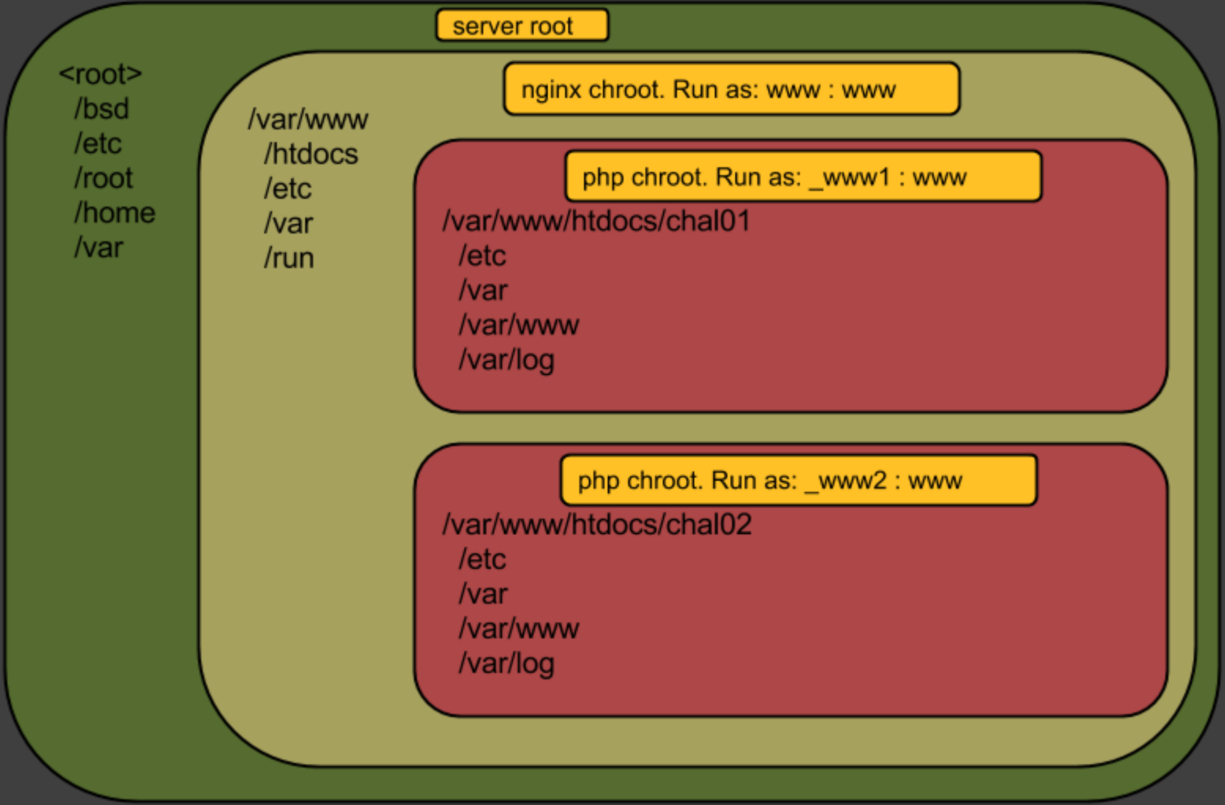
Internal (trusted)

Connections to External network are ALWAYS initiated from Internal network. No exception.

syslog
git?
database?

Push configurations
Get logs someway
Gather statistics
Run AIDE (integrity check)

# [Strategy] Choosing an architecture



server root

<root>
/bsd
/etc
/root
/home
/var

nginx chroot. Run as: www : www

/var/www
/htdocs
/etc
/var
/run

php chroot. Run as: _www1 : www

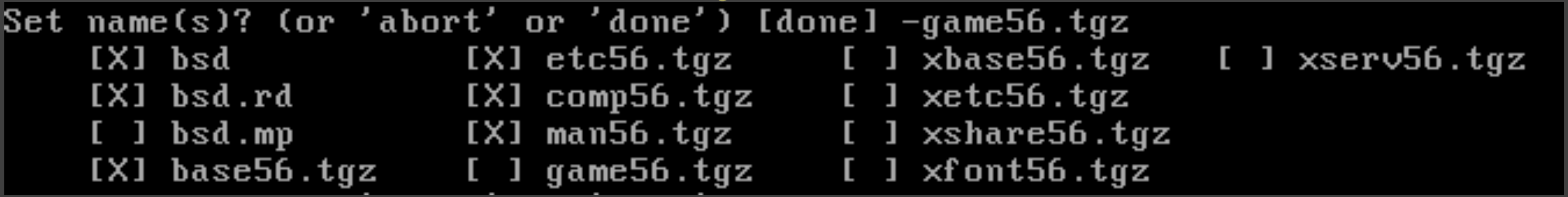
/var/www/htdocs/chal01
/etc
/var
/var/www
/var/log

php chroot. Run as: _www2 : www

/var/www/htdocs/chal02
/etc
/var
/var/www
/var/log

# [Impl.] OS Installation

- Minimal Install. Enable only what’s needed.



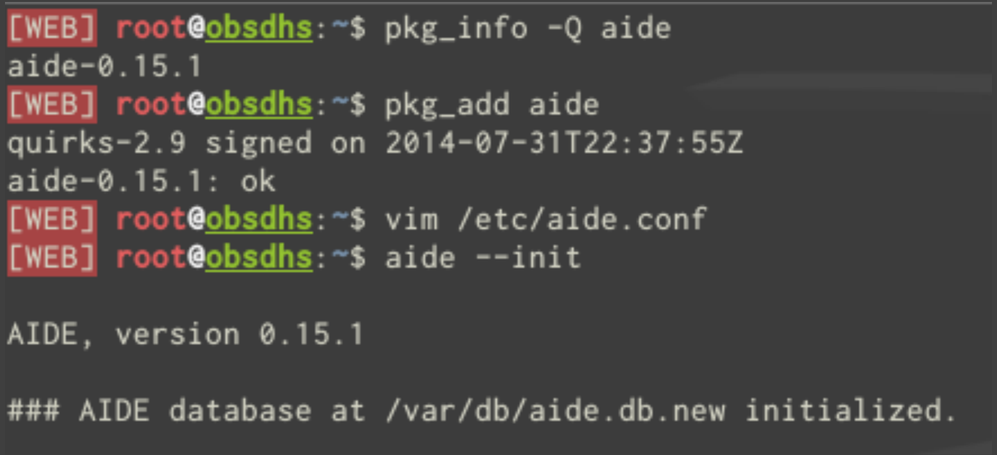
```
Set name(s)? (or 'abort' or 'done') [done] -game56.tgz
    [X] bsd               [X] etc56.tgz      [ ] xbase56.tgz    [ ] xserv56.tgz
    [X] bsd.rd            [X] comp56.tgz     [ ] xetc56.tgz
    [ ] bsd.mp            [X] man56.tgz      [ ] xshare56.tgz
    [X] base56.tgz        [ ] game56.tgz     [ ] xfont56.tgz
```

- During install
  - An admin user **can** be created
  - OpenSSH server **can** be installed
  - root login **can** be restricted on OpenSSH
  - A ntp server **can** be configured
- After install
  - Install your favourite tools

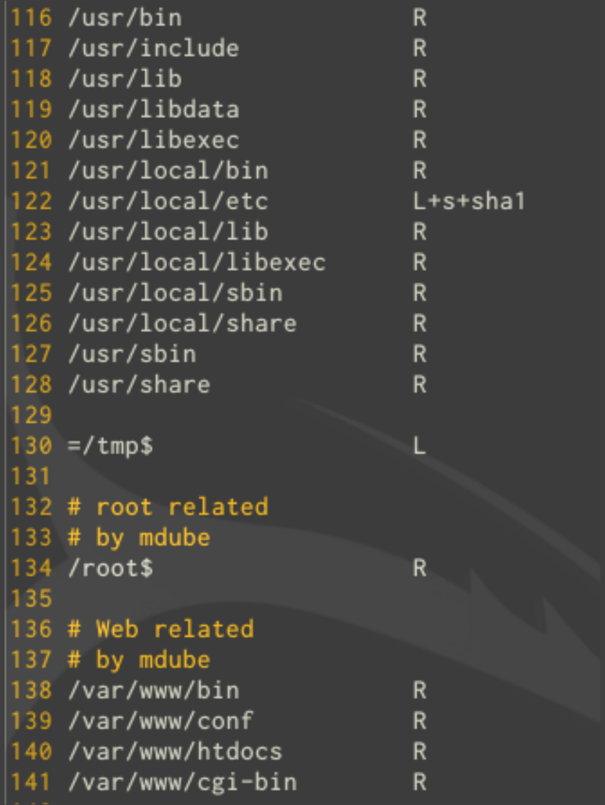
# [Impl.] OS Hardening

- Filesystem integrity
  - Install, configure and run AIDE

    (Advanced Intrusion Detection Environment)



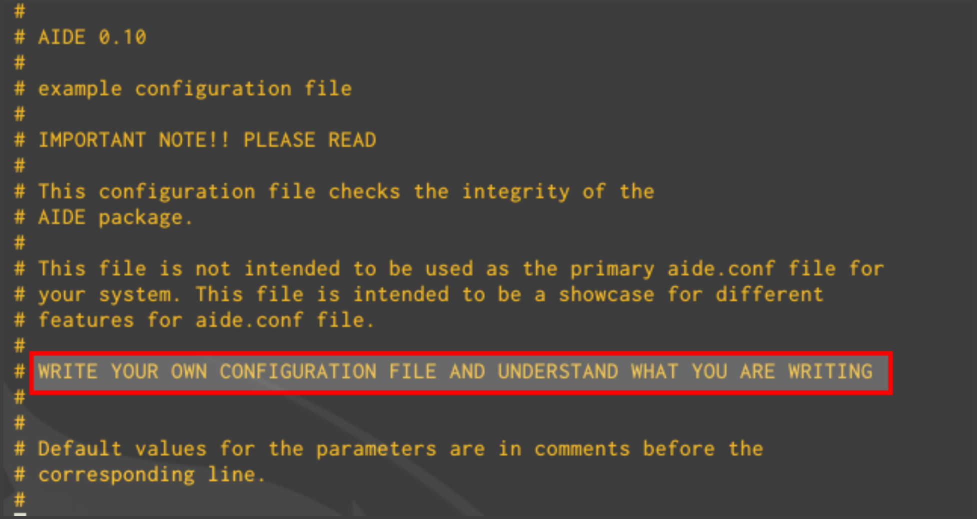
```
[WEB] root@obsdhs:~$ pkg_info -Q aide
aide-0.15.1
[WEB] root@obsdhs:~$ pkg_add aide
quirks-2.9 signed on 2014-07-31T22:37:55Z
aide-0.15.1: ok
[WEB] root@obsdhs:~$ vim /etc/aide.conf
[WEB] root@obsdhs:~$ aide --init

AIDE, version 0.15.1

### AIDE database at /var/db/aide.db.new initialized.
```



```
/usr/bin                R
/usr/include            R
/usr/lib                R
/usr/libdata            R
/usr/libexec            R
/usr/local/bin          R
/usr/local/etc          L+s+sha1
/usr/local/lib          R
/usr/local/libexec      R
/usr/local/sbin         R
/usr/local/share        R
/usr/sbin               R
/usr/share              R

=/tmp$                  L

# root related
# by mdube
/root$                  R

# Web related
# by mdube
/var/www/bin            R
/var/www/conf           R
/var/www/htdocs         R
/var/www/cgi-bin        R
```

# [Impl.] OS Hardening

- Filesystem integrity
  - Don’t forget to download the file at a safe place!
  - Keep understanding what you are doing :)



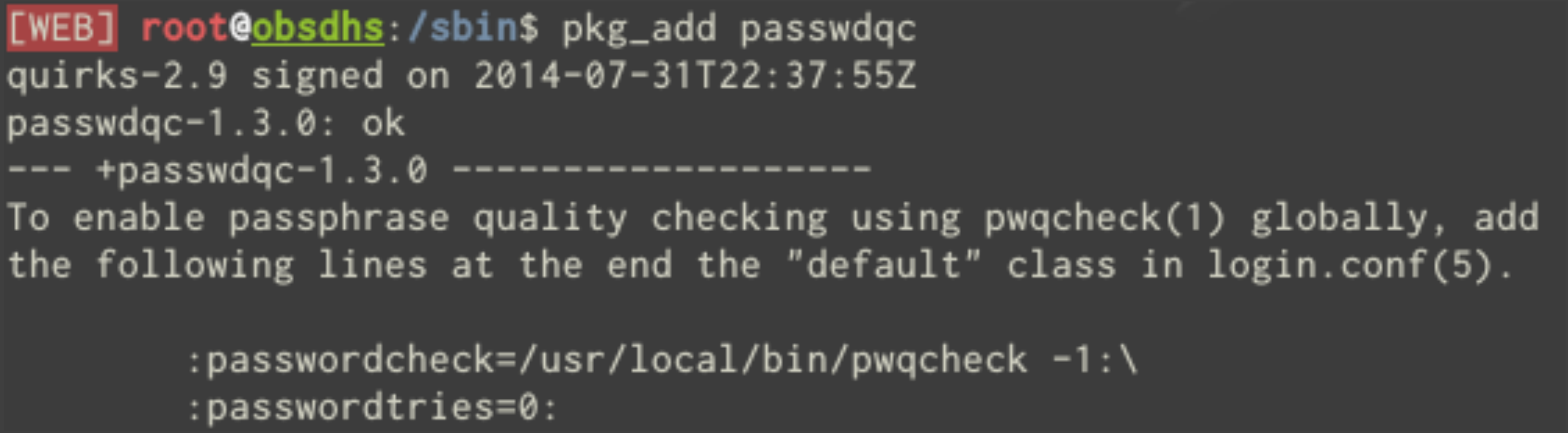
```
#
# AIDE 0.10
#
# example configuration file
#
# IMPORTANT NOTE!! PLEASE READ
#
# This configuration file checks the integrity of the
# AIDE package.
#
# This file is not intended to be used as the primary aide.conf file for
# your system. This file is intended to be a showcase for different
# features for aide.conf file.
#
# WRITE YOUR OWN CONFIGURATION FILE AND UNDERSTAND WHAT YOU ARE WRITING
#
#
# Default values for the parameters are in comments before the
# corresponding line.
#
```

# [Impl.] OS Hardening

- Filesystem integrity
  - To check for changes on the file system:
    - Upload latest baseline at /var/db/aide.db
    - aide --check
  - To update database with new files
    - aide --update

# [Impl.] OS Hardening

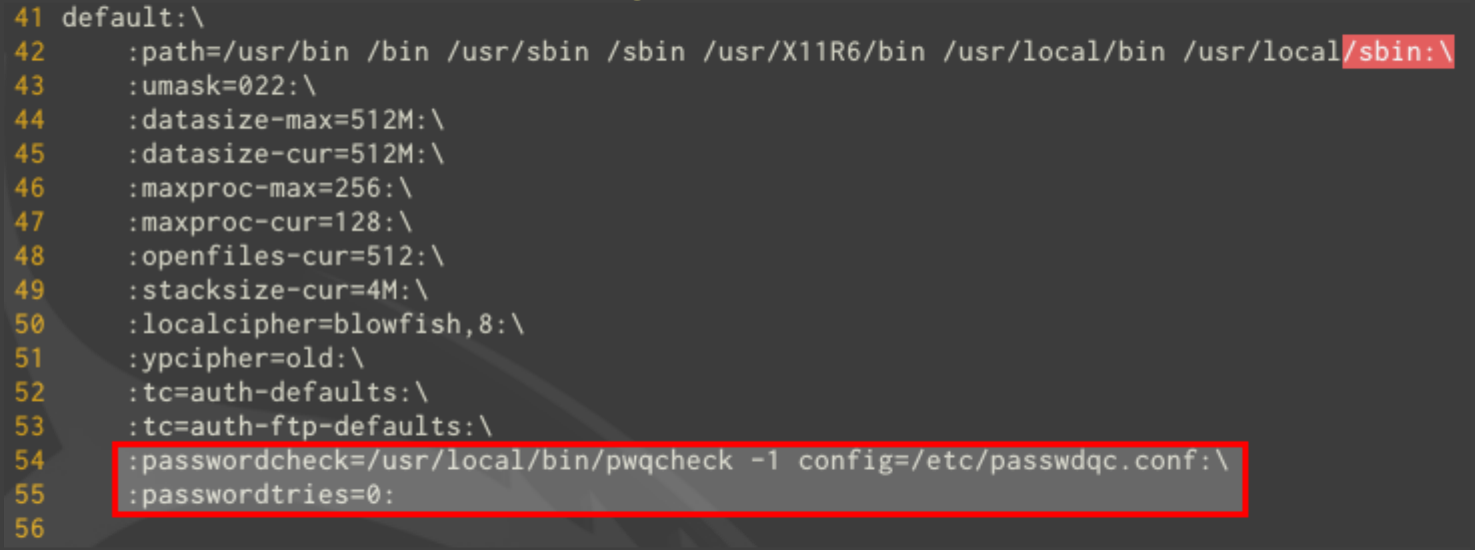
- Password Policy



```
[WEB] root@obsdhs:/sbin$ pkg_add passwdqc
quirks-2.9 signed on 2014-07-31T22:37:55Z
passwdqc-1.3.0: ok
--- +passwdqc-1.3.0 -------------------
To enable passphrase quality checking using pwqcheck(1) globally, add
the following lines at the end the "default" class in login.conf(5).

        :passwordcheck=/usr/local/bin/pwqcheck -1:\
        :passwordtries=0:
```

# [Impl.] OS Hardening

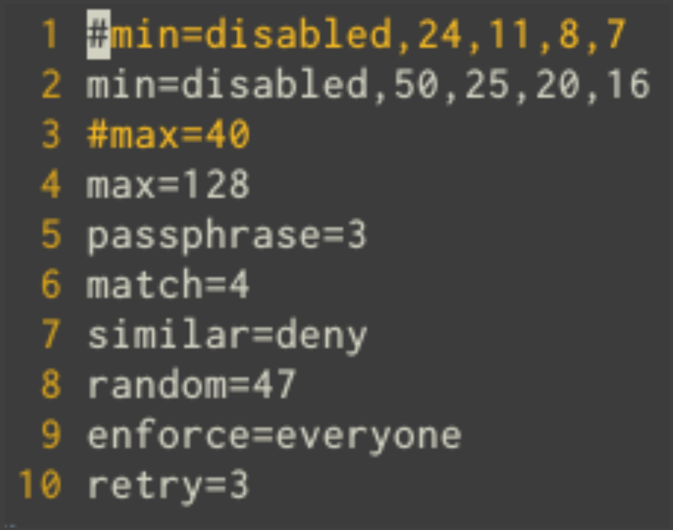
- Password Policy
  - Add these lines in /etc/login.conf



```
default:\
        :path=/usr/bin /bin /usr/sbin /sbin /usr/X11R6/bin /usr/local/bin /usr/local/sbin:\
        :umask=022:\
        :datasize-max=512M:\
        :datasize-cur=512M:\
        :maxproc-max=256:\
        :maxproc-cur=128:\
        :openfiles-cur=512:\
        :stacksize-cur=4M:\
        :localcipher=blowfish,8:\
        :ypcipher=old:\
        :tc=auth-defaults:\
        :tc=auth-ftp-defaults:\
        :passwordcheck=/usr/local/bin/pwqcheck -1 config=/etc/passwdqc.conf:\
        :passwordtries=0:
```

# [Impl.] OS Hardening

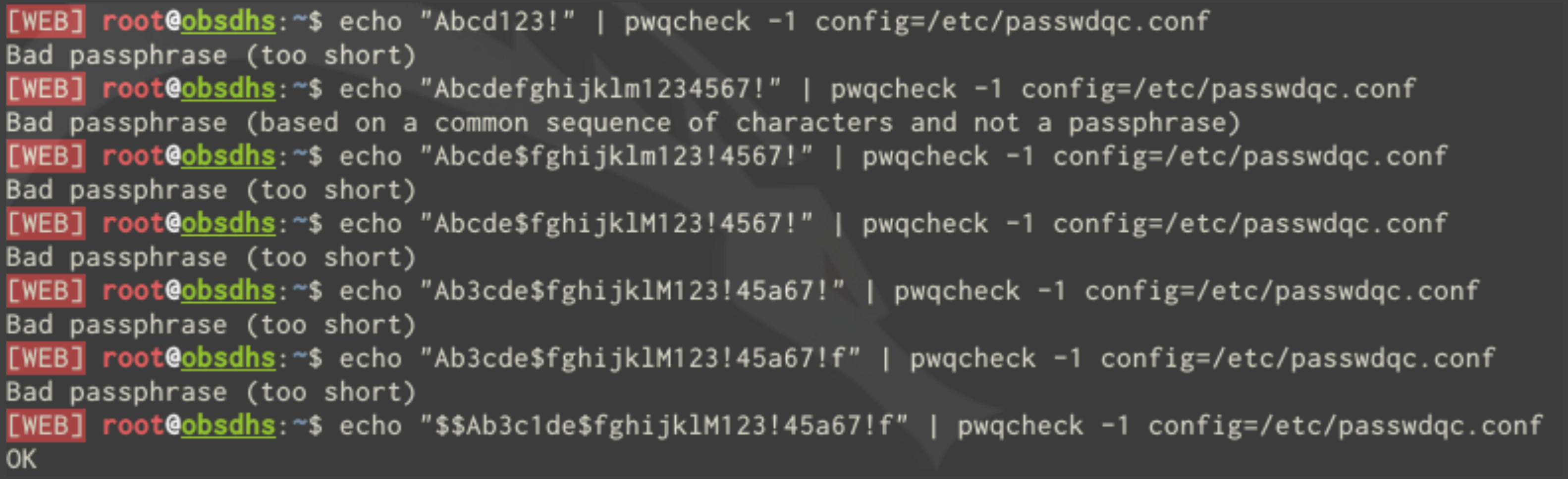
- Password Policy
  - Configure policy in /etc/passwdqc.conf



```
#min=disabled,24,11,8,7
min=disabled,50,25,20,16
#max=40
max=128
passphrase=3
match=4
similar=deny
random=47
enforce=everyone
retry=3
```

# [Impl.] OS Hardening

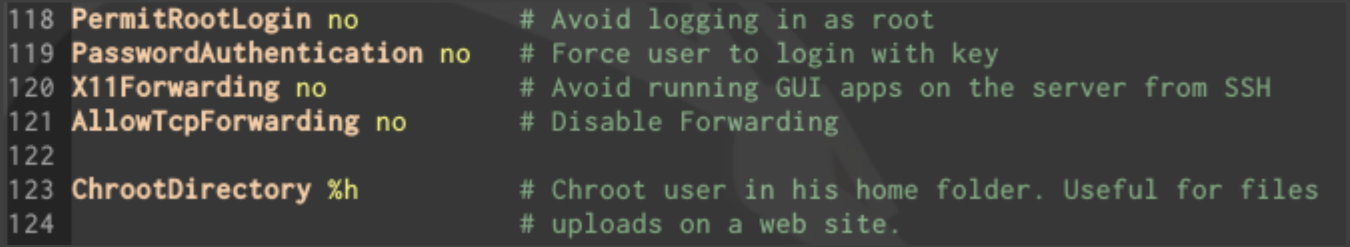
- Password Policy
  - Test the policy



```
[WEB] root@obsdhs:~$ echo "Abcd123!" | pwqcheck -1 config=/etc/passwdqc.conf
Bad passphrase (too short)
[WEB] root@obsdhs:~$ echo "Abcdefghijklm1234567!" | pwqcheck -1 config=/etc/passwdqc.conf
Bad passphrase (based on a common sequence of characters and not a passphrase)
[WEB] root@obsdhs:~$ echo "Abcde$fghijklm123!4567!" | pwqcheck -1 config=/etc/passwdqc.conf
Bad passphrase (too short)
[WEB] root@obsdhs:~$ echo "Abcde$fghijklM123!4567!" | pwqcheck -1 config=/etc/passwdqc.conf
Bad passphrase (too short)
[WEB] root@obsdhs:~$ echo "Ab3cde$fghijklM123!45a67!" | pwqcheck -1 config=/etc/passwdqc.conf
Bad passphrase (too short)
[WEB] root@obsdhs:~$ echo "Ab3cde$fghijklM123!45a67!f" | pwqcheck -1 config=/etc/passwdqc.conf
Bad passphrase (too short)
[WEB] root@obsdhs:~$ echo "$$Ab3c1de$fghijklM123!45a67!f" | pwqcheck -1 config=/etc/passwdqc.conf
OK
```

# [Impl.] OS Hardening

- Server Management: OpenSSH
  - Server side consideration

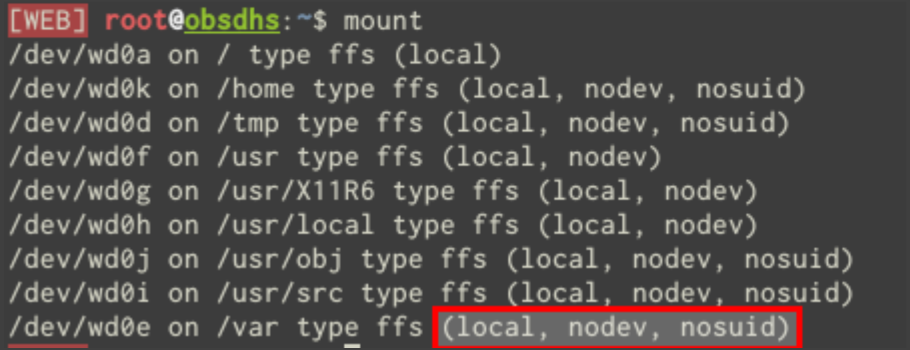


```
PermitRootLogin no          # Avoid logging in as root
PasswordAuthentication no   # Force user to login with key
X11Forwarding no            # Avoid running GUI apps on the server from SSH
AllowTcpForwarding no       # Disable Forwarding

ChrootDirectory %h          # Chroot user in his home folder. Useful for files
                            # uploads on a web site.
```

  - Client side consideration
    - Create a strong key (-t rsa -b 4096)
    - Encrypt it with PKCS#8

# [Impl.] OS Hardening

- Partitions security
  - Default partitions scheme



```
[WEB] root@obsdhs:~$ mount
/dev/wd0a on / type ffs (local)
/dev/wd0k on /home type ffs (local, nodev, nosuid)
/dev/wd0d on /tmp type ffs (local, nodev, nosuid)
/dev/wd0f on /usr type ffs (local, nodev)
/dev/wd0g on /usr/X11R6 type ffs (local, nodev)
/dev/wd0h on /usr/local type ffs (local, nodev)
/dev/wd0j on /usr/obj type ffs (local, nodev, nosuid)
/dev/wd0i on /usr/src type ffs (local, nodev, nosuid)
/dev/wd0e on /var type ffs (local, nodev, nosuid)
```

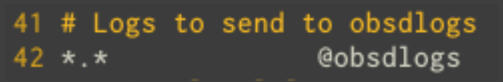
  - Other interesting options: noexec, rdonly

# [Impl.] OS Hardening

- Logs export
  - Distributed setup
    - Authenticate and Encrypt communication with log server
    - http://cromwell-intl.com/cybersecurity/syslog-tls-cloud.html
  - Local setup
    - Just send logs on a remote server
    - Use a dedicated network

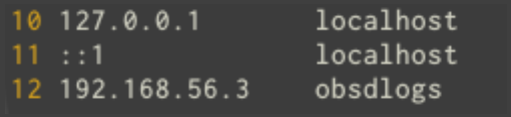
# [Impl.] OS Hardening

- Logs export
  - Config on client
    - /etc/syslog.conf



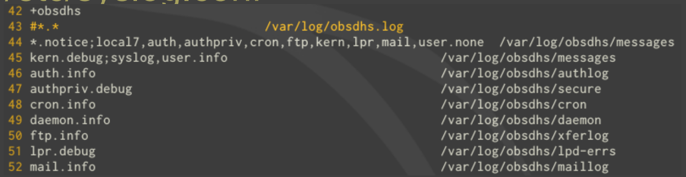
```
# Logs to send to obsdlogs
*.*                 @obsdlogs
```

    - /etc/hosts



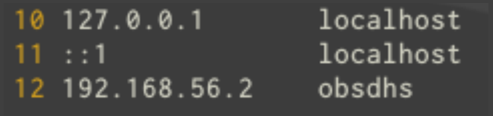
```
127.0.0.1         localhost
::1               localhost
192.168.56.3      obsdlogs
```

  - Config on server
    - /etc/syslog.conf



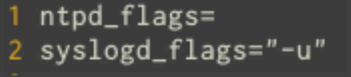
```
+obsdhs
#*.*                                        /var/log/obsdhs.log
*.notice;local7,auth,authpriv,cron,ftp,kern,lpr,mail,user.none  /var/log/obsdhs/messages
kern.debug;syslog,user.info                                 /var/log/obsdhs/messages
auth.info                                                   /var/log/obsdhs/authlog
authpriv.debug                                              /var/log/obsdhs/secure
cron.info                                                   /var/log/obsdhs/cron
daemon.info                                                 /var/log/obsdhs/daemon
ftp.info                                                    /var/log/obsdhs/xferlog
lpr.debug                                                   /var/log/obsdhs/lpd-errs
mail.info                                                   /var/log/obsdhs/maillog
```

    - /etc/hosts



```
127.0.0.1         localhost
::1               localhost
192.168.56.2      obsdhs
```

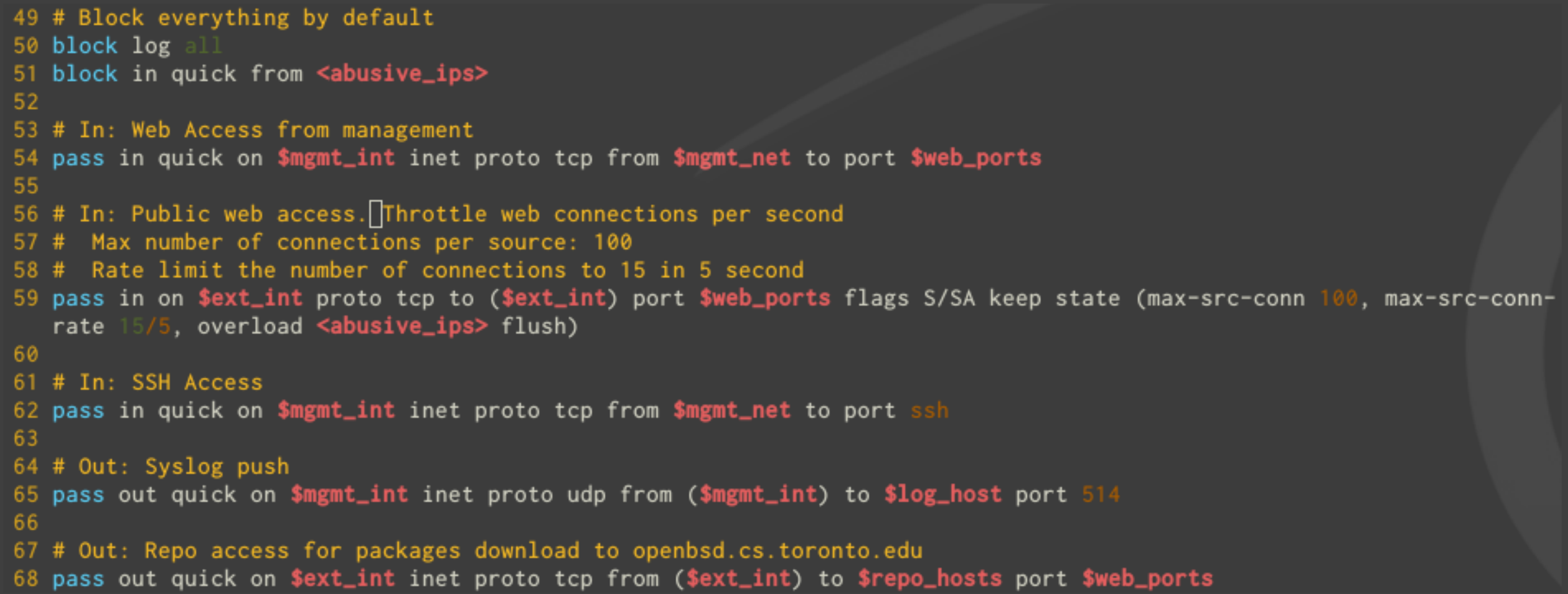
    - /etc/rc.conf.local (*Note: -u flag is considered insecure* )



```
ntpd_flags=
syslogd_flags="-u"
```

# [Impl.] OS Hardening

➢ Firewall rules



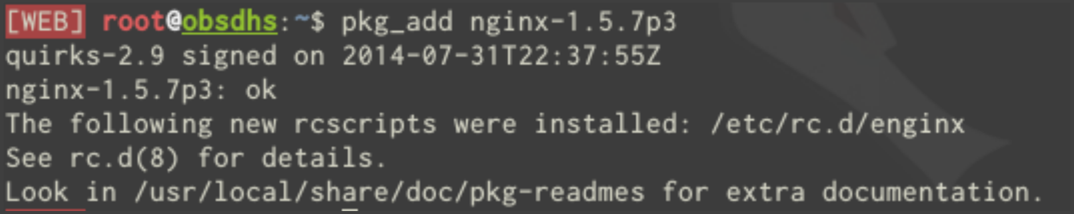
```
# Block everything by default
block log all
block in quick from <abusive_ips>

# In: Web Access from management
pass in quick on $mgmt_int inet proto tcp from $mgmt_net to port $web_ports

# In: Public web access.□Throttle web connections per second
#  Max number of connections per source: 100
#  Rate limit the number of connections to 15 in 5 second
pass in on $ext_int proto tcp to ($ext_int) port $web_ports flags S/SA keep state (max-src-conn 100, max-src-conn-rate 15/5, overload <abusive_ips> flush)

# In: SSH Access
pass in quick on $mgmt_int inet proto tcp from $mgmt_net to port ssh

# Out: Syslog push
pass out quick on $mgmt_int inet proto udp from ($mgmt_int) to $log_host port 514

# Out: Repo access for packages download to openbsd.cs.toronto.edu
pass out quick on $ext_int inet proto tcp from ($ext_int) to $repo_hosts port $web_ports
```

# [Impl.] OS Hardening

- If SSH must be publicly accessible
  - Implement fail2ban with PF
  - http://www.bsdguides.org/2012/fail2ban-with-pf-on-openbsd-5-2/

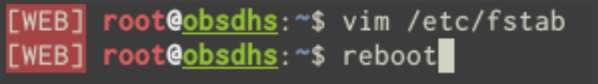
# [Impl.] Web server Installation

- Installation

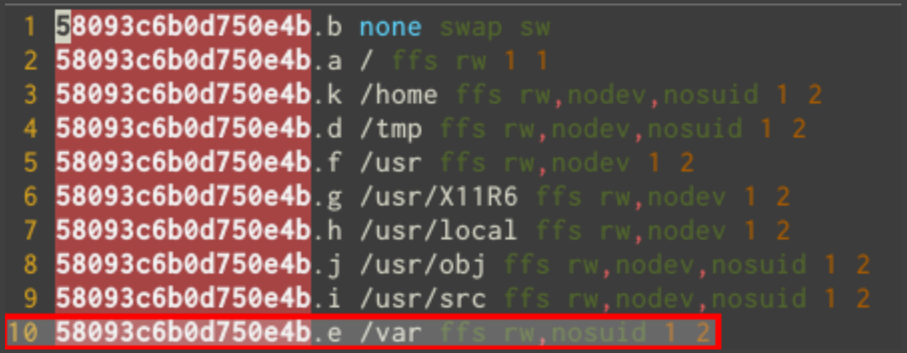


```
[WEB] root@obsdhs:~$ pkg_add nginx-1.5.7p3
quirks-2.9 signed on 2014-07-31T22:37:55Z
nginx-1.5.7p3: ok
The following new rcscripts were installed: /etc/rc.d/enginx
See rc.d(8) for details.
Look in /usr/local/share/doc/pkg-readmes for extra documentation.
```

- Remove “nodev” flag from /var



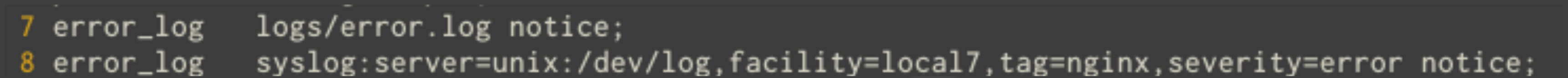
```
[WEB] root@obsdhs:~$ vim /etc/fstab
[WEB] root@obsdhs:~$ reboot
```



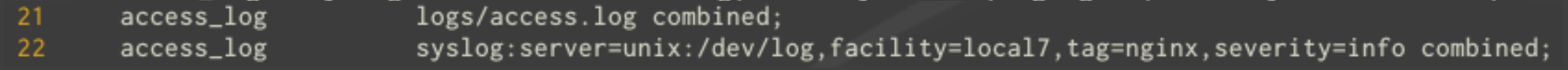
```
58093c6b0d750e4b.b none swap sw
58093c6b0d750e4b.a / ffs rw 1 1
58093c6b0d750e4b.k /home ffs rw,nodev,nosuid 1 2
58093c6b0d750e4b.d /tmp ffs rw,nodev,nosuid 1 2
58093c6b0d750e4b.f /usr ffs rw,nodev 1 2
58093c6b0d750e4b.g /usr/X11R6 ffs rw,nodev 1 2
58093c6b0d750e4b.h /usr/local ffs rw,nodev 1 2
58093c6b0d750e4b.j /usr/obj ffs rw,nodev,nosuid 1 2
58093c6b0d750e4b.i /usr/src ffs rw,nodev,nosuid 1 2
58093c6b0d750e4b.e /var ffs rw,nosuid 1 2
```

# [Impl.] Web server Hardening

- Remove default web site
  - /etc/nginx/nginx.conf
    - Delete: section http -> server
    - Add: include conf.d/*.conf;
- Configure logging

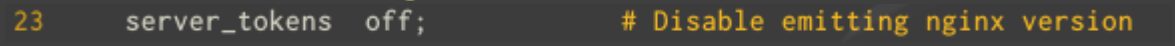


```
error_log   logs/error.log notice;
error_log   syslog:server=unix:/dev/log,facility=local7,tag=nginx,severity=error notice;
```



```
    access_log          logs/access.log combined;
    access_log          syslog:server=unix:/dev/log,facility=local7,tag=nginx,severity=info combined;
```

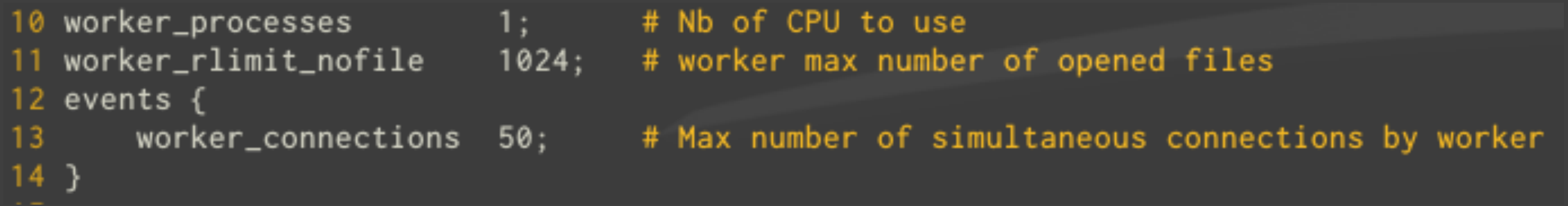
- Remove nginx version in errors



```
    server_tokens  off;           # Disable emitting nginx version
```

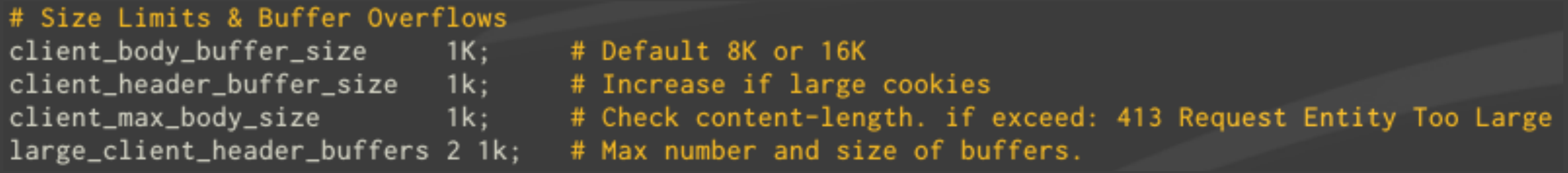
# [Impl.] Web server Hardening

- Set limits



```
worker_processes        1;      # Nb of CPU to use
worker_rlimit_nofile    1024;   # worker max number of opened files
events {
    worker_connections  50;     # Max number of simultaneous connections by worker
}
```

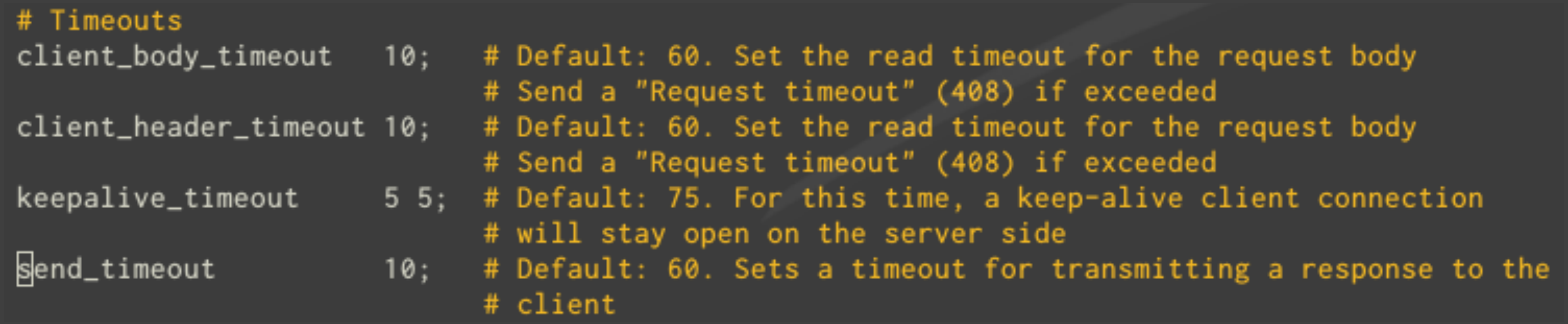
- Set more limits



```
# Size Limits & Buffer Overflows
client_body_buffer_size     1K;     # Default 8K or 16K
client_header_buffer_size   1k;     # Increase if large cookies
client_max_body_size        1k;     # Check content-length. if exceed: 413 Request Entity Too Large
large_client_header_buffers 2 1k;   # Max number and size of buffers.
```

# [Impl.] Web server Hardening

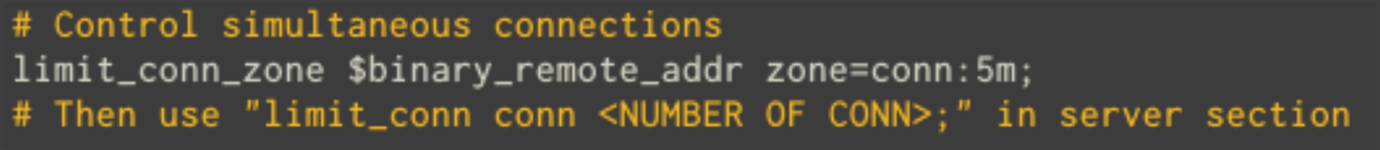
- Set timeouts



```
# Timeouts
client_body_timeout   10;    # Default: 60. Set the read timeout for the request body
                             # Send a "Request timeout" (408) if exceeded
client_header_timeout 10;    # Default: 60. Set the read timeout for the request body
                             # Send a "Request timeout" (408) if exceeded
keepalive_timeout     5 5;   # Default: 75. For this time, a keep-alive client connection
                             # will stay open on the server side
send_timeout          10;    # Default: 60. Sets a timeout for transmitting a response to the
                             # client
```

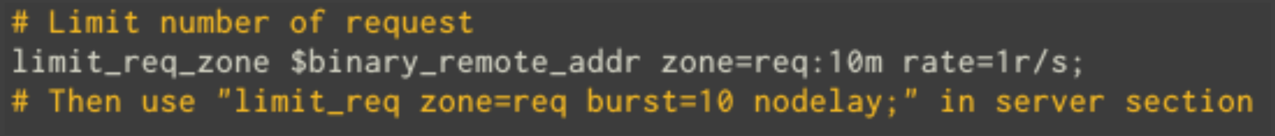
# [Impl.] Web server Hardening

- Set max simultaneous connections



```
# Control simultaneous connections
limit_conn_zone $binary_remote_addr zone=conn:5m;
# Then use "limit_conn conn <NUMBER OF CONN>;" in server section
```

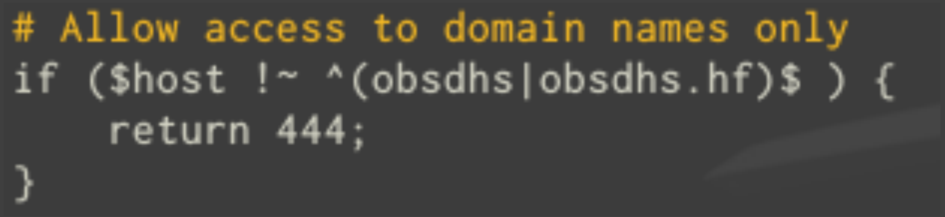
- Set max concurrent connections



```
# Limit number of request
limit_req_zone $binary_remote_addr zone=req:10m rate=1r/s;
# Then use "limit_req zone=req burst=10 nodelay;" in server section
```

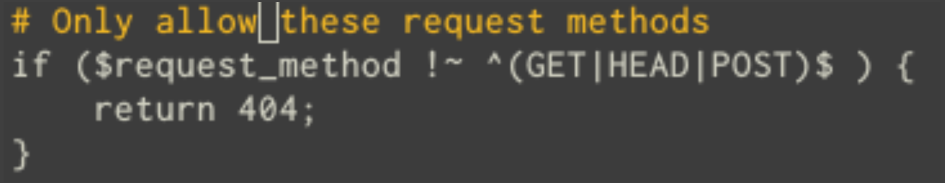
# [Impl.] Web server Hardening

- Limit access to our domain(s)



```
# Allow access to domain names only
if ($host !~ ^(obsdhs|obsdhs.hf)$ ) {
     return 444;
}
```

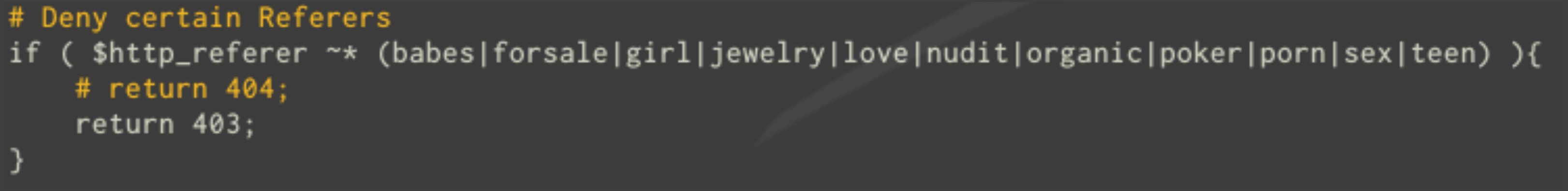
- Limit Request Methods



```
# Only allow these request methods
if ($request_method !~ ^(GET|HEAD|POST)$ ) {
     return 404;
}
```

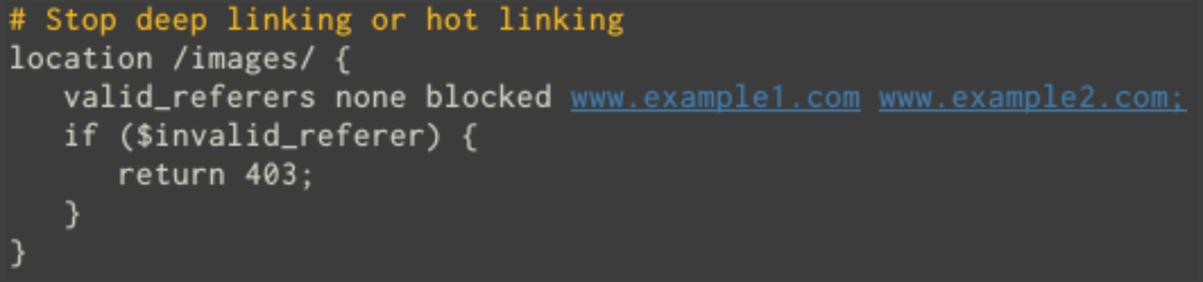
# [Impl.] Web server Hardening

- Block Referrer (Spam)



```
# Deny certain Referers
if ( $http_referer ~* (babes|forsale|girl|jewelry|love|nudit|organic|poker|porn|sex|teen) ){
    # return 404;
    return 403;
}
```

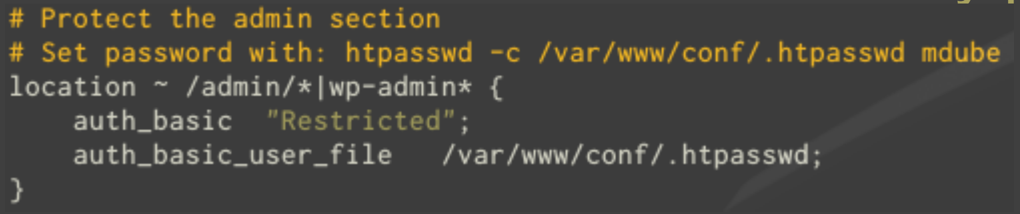
- Block Image Hotlinking



```
# Stop deep linking or hot linking
location /images/ {
   valid_referers none blocked www.example1.com www.example2.com;
   if ($invalid_referer) {
      return 403;
   }
}
```

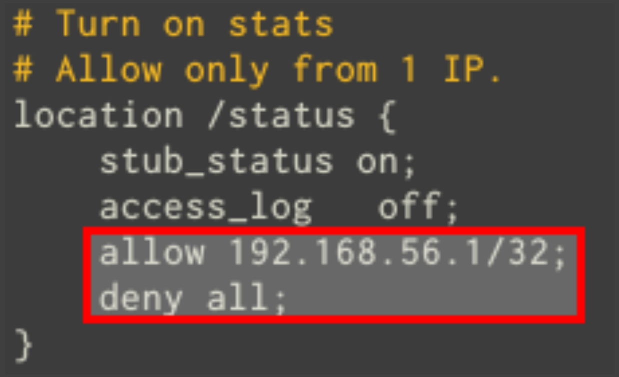
# [Impl.] Web server Hardening

- ➢ Limit access to some folder/files by password



```
# Protect the admin section
# Set password with: htpasswd -c /var/www/conf/.htpasswd mdube
location ~ /admin/*|wp-admin* {
    auth_basic  "Restricted";
    auth_basic_user_file   /var/www/conf/.htpasswd;
}
```

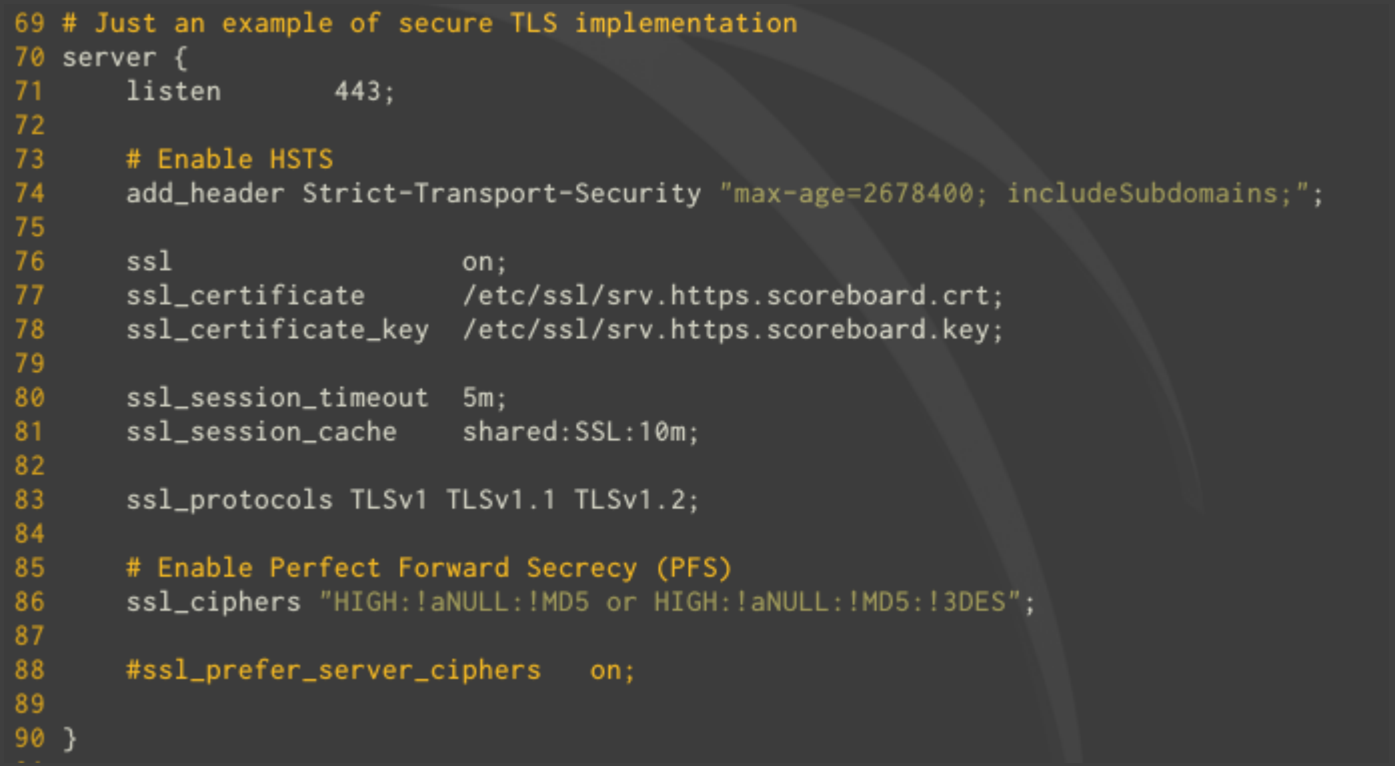
- ➢ Limit access to some folder/files by IP



```
# Turn on stats
# Allow only from 1 IP.
location /status {
    stub_status on;
    access_log   off;
    allow 192.168.56.1/32;
    deny all;
}
```

# [Impl.] Web server Hardening

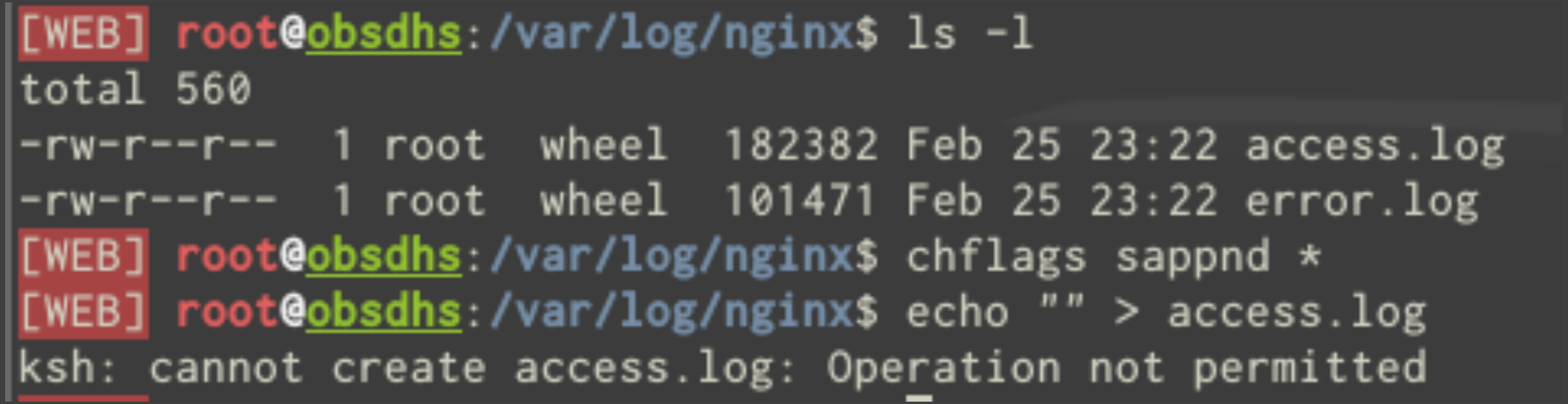
- TLS



```
# Just an example of secure TLS implementation
server {
    listen       443;

    # Enable HSTS
    add_header Strict-Transport-Security "max-age=2678400; includeSubdomains;";

    ssl                     on;
    ssl_certificate         /etc/ssl/srv.https.scoreboard.crt;
    ssl_certificate_key     /etc/ssl/srv.https.scoreboard.key;

    ssl_session_timeout     5m;
    ssl_session_cache       shared:SSL:10m;

    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;

    # Enable Perfect Forward Secrecy (PFS)
    ssl_ciphers "HIGH:!aNULL:!MD5 or HIGH:!aNULL:!MD5:!3DES";

    #ssl_prefer_server_ciphers   on;

}
```

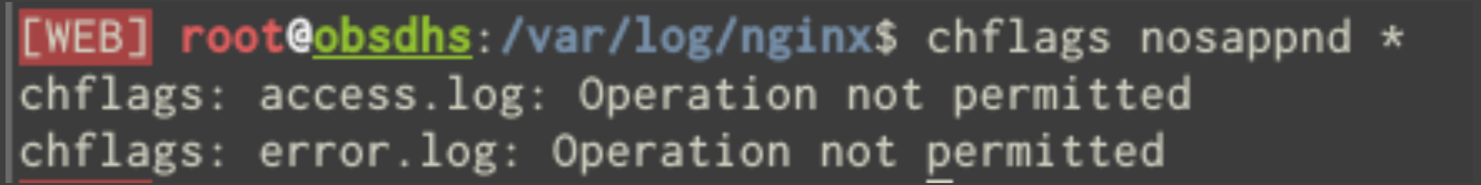
# [Impl.] Web server Hardening

- Secure logs



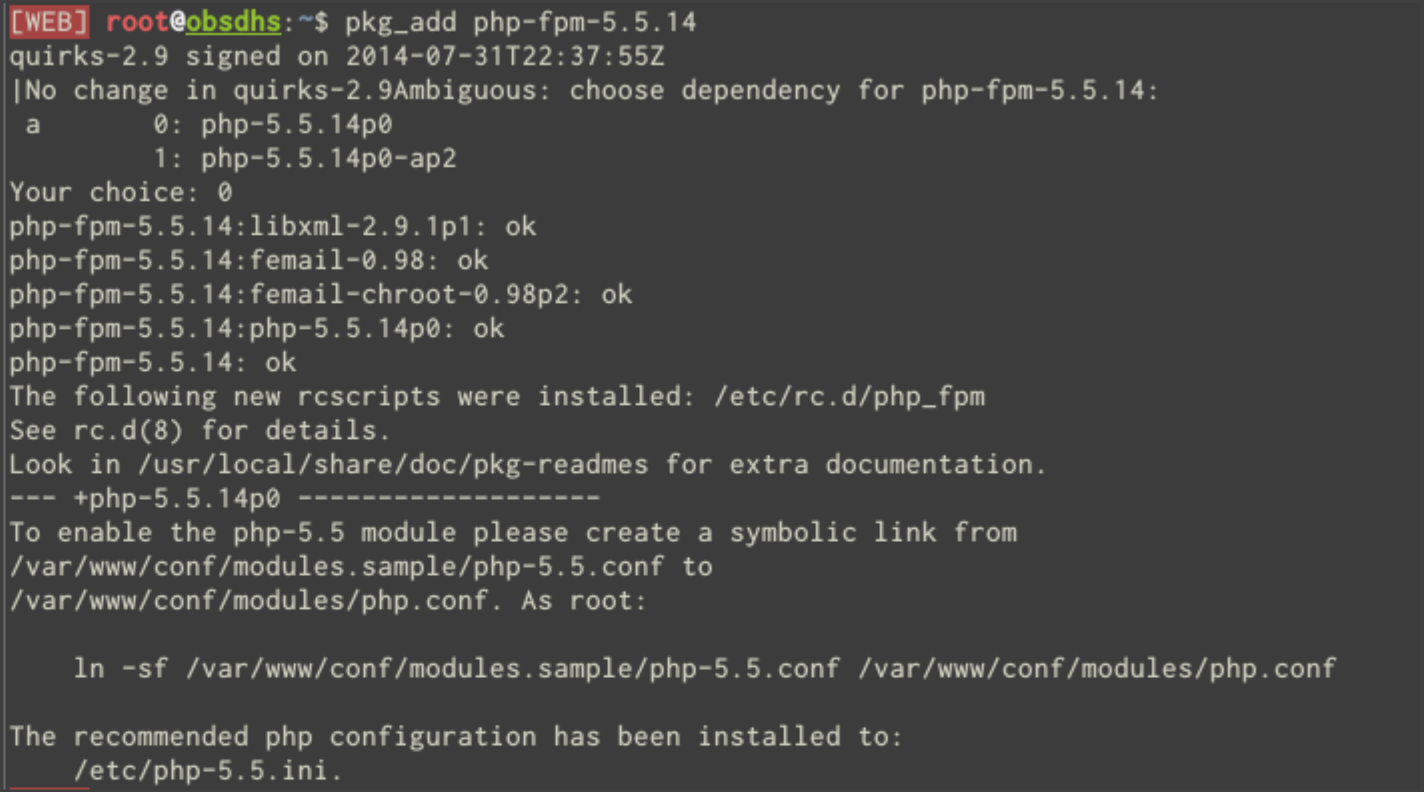
```
[WEB] root@obsdhs:/var/log/nginx$ ls -l
total 560
-rw-r--r--  1 root  wheel  182382 Feb 25 23:22 access.log
-rw-r--r--  1 root  wheel  101471 Feb 25 23:22 error.log
[WEB] root@obsdhs:/var/log/nginx$ chflags sappnd *
[WEB] root@obsdhs:/var/log/nginx$ echo "" > access.log
ksh: cannot create access.log: Operation not permitted
```

Remember what does Securelevel=1 ?



```
[WEB] root@obsdhs:/var/log/nginx$ chflags nosappnd *
chflags: access.log: Operation not permitted
chflags: error.log: Operation not permitted
```

# [Impl.] PHP Installation



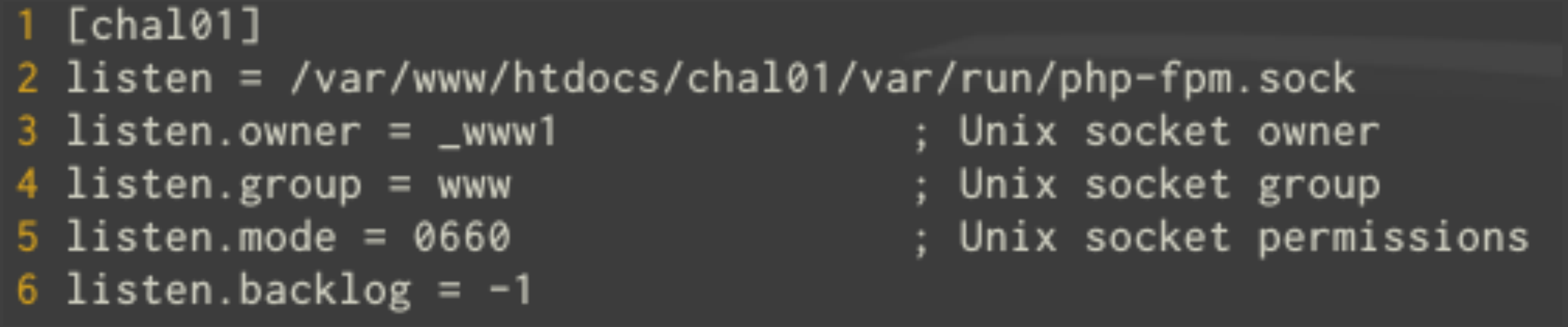
```
[WEB] root@obsdhs:~$ pkg_add php-fpm-5.5.14
quirks-2.9 signed on 2014-07-31T22:37:55Z
|No change in quirks-2.9Ambiguous: choose dependency for php-fpm-5.5.14:
 a        0: php-5.5.14p0
          1: php-5.5.14p0-ap2
Your choice: 0
php-fpm-5.5.14:libxml-2.9.1p1: ok
php-fpm-5.5.14:femail-0.98: ok
php-fpm-5.5.14:femail-chroot-0.98p2: ok
php-fpm-5.5.14:php-5.5.14p0: ok
php-fpm-5.5.14: ok
The following new rcscripts were installed: /etc/rc.d/php_fpm
See rc.d(8) for details.
Look in /usr/local/share/doc/pkg-readmes for extra documentation.
--- +php-5.5.14p0 -------------------
To enable the php-5.5 module please create a symbolic link from
/var/www/conf/modules.sample/php-5.5.conf to
/var/www/conf/modules/php.conf. As root:

    ln -sf /var/www/conf/modules.sample/php-5.5.conf /var/www/conf/modules/php.conf

The recommended php configuration has been installed to:
    /etc/php-5.5.ini.
```

# [Impl.] PHP Hardening

- Remove default pool
  - /etc/php-fpm.conf
    - Comment section [www]
    - Add: include=/etc/fpm.d/*.conf
- Create and harden one php file per apps
  - php_value vs php_admin_value
    - The application cannot change its php.ini parameters with php_admin_value with ini_set()

# [Impl.] PHP Hardening

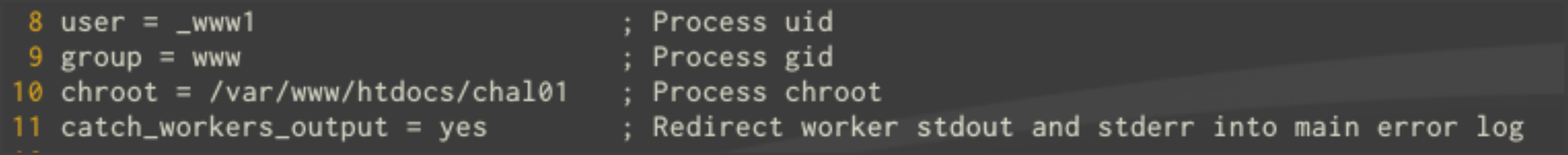
- Harden unix socket security



```
[chal01]
listen = /var/www/htdocs/chal01/var/run/php-fpm.sock
listen.owner = _www1              ; Unix socket owner
listen.group = www                ; Unix socket group
listen.mode = 0660                ; Unix socket permissions
listen.backlog = -1
```

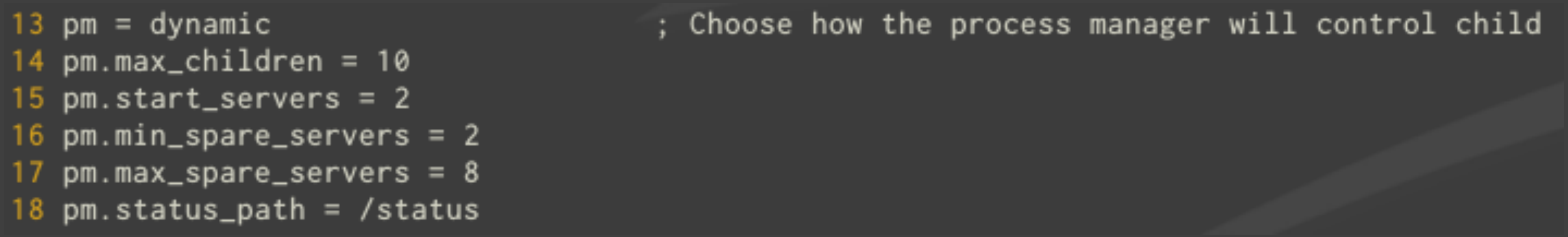
# [Impl.] PHP Hardening

- Harden process security



```
user = _www1                          ; Process uid
group = www                           ; Process gid
chroot = /var/www/htdocs/chal01       ; Process chroot
catch_workers_output = yes            ; Redirect worker stdout and stderr into main error log
```

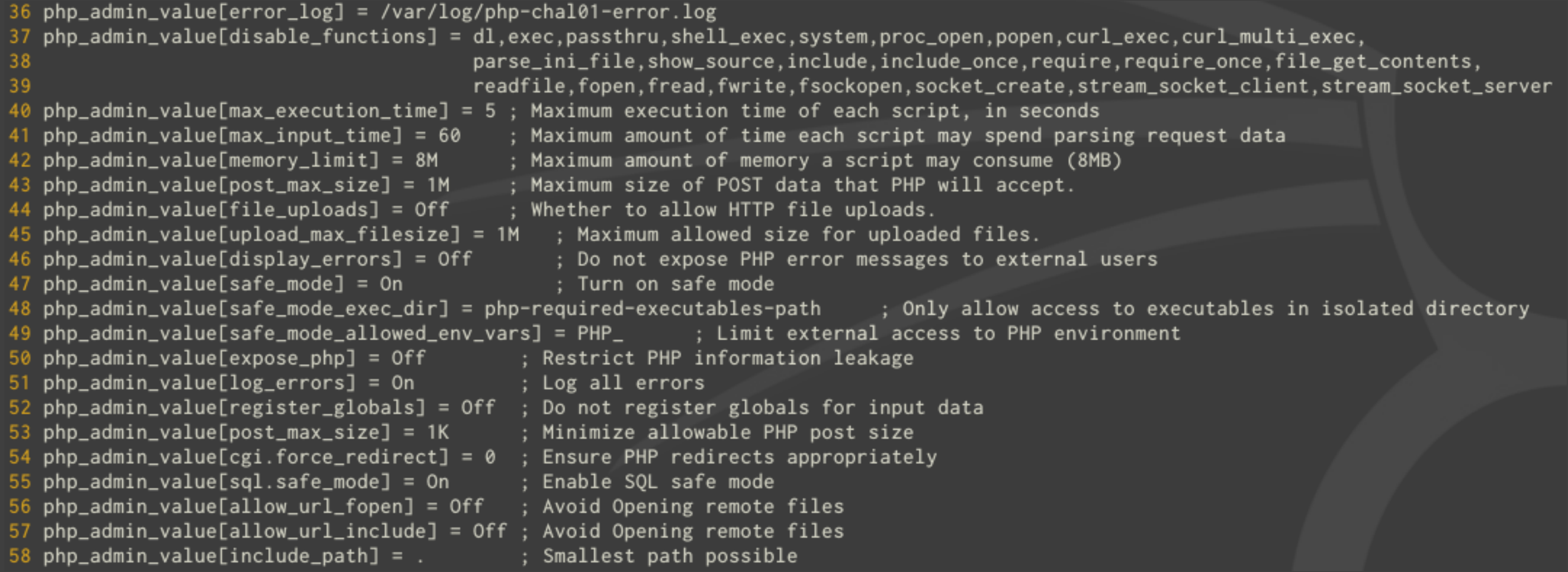
- Harden memory management



```
pm = dynamic                            ; Choose how the process manager will control child
pm.max_children = 10
pm.start_servers = 2
pm.min_spare_servers = 2
pm.max_spare_servers = 8
pm.status_path = /status
```

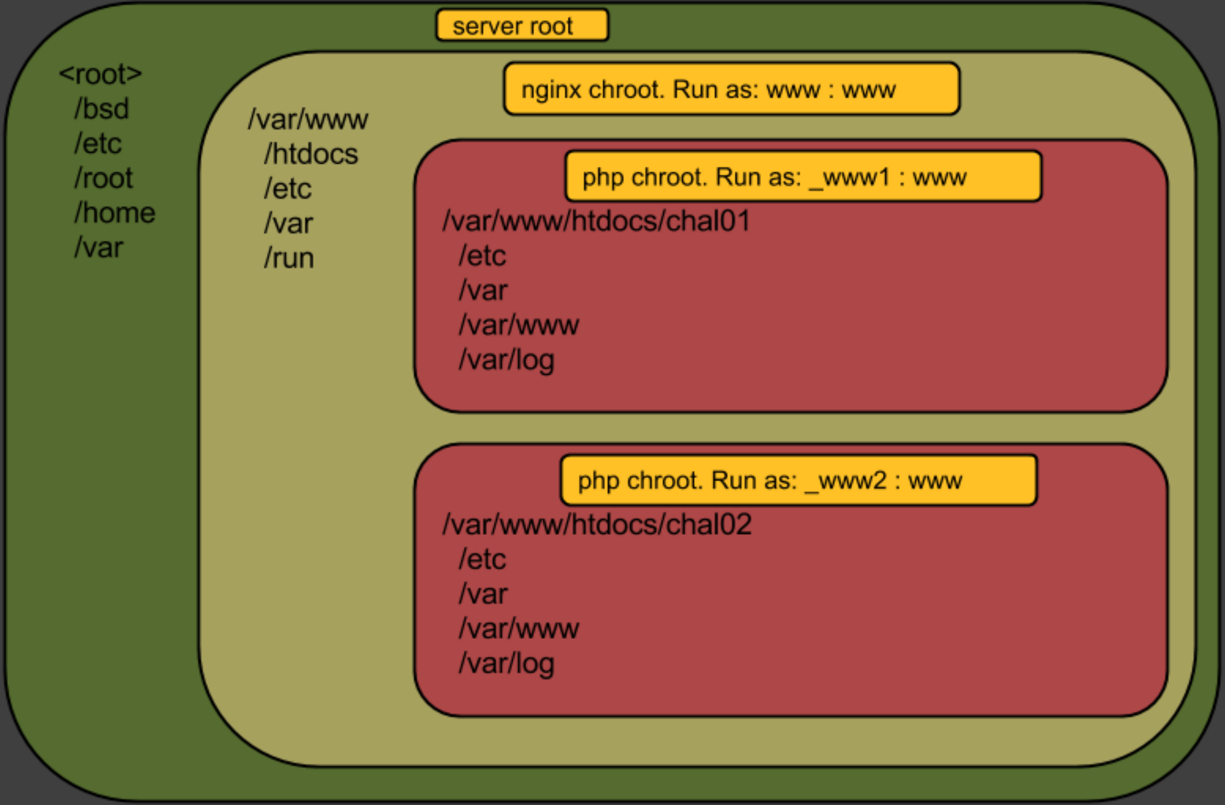
# [Impl.] PHP Hardening

- PHP ini file



```
php_admin_value[error_log] = /var/log/php-chal01-error.log
php_admin_value[disable_functions] = dl,exec,passthru,shell_exec,system,proc_open,popen,curl_exec,curl_multi_exec,
                                     parse_ini_file,show_source,include,include_once,require,require_once,file_get_contents,
                                     readfile,fopen,fread,fwrite,fsockopen,socket_create,stream_socket_client,stream_socket_server
php_admin_value[max_execution_time] = 5 ; Maximum execution time of each script, in seconds
php_admin_value[max_input_time] = 60    ; Maximum amount of time each script may spend parsing request data
php_admin_value[memory_limit] = 8M      ; Maximum amount of memory a script may consume (8MB)
php_admin_value[post_max_size] = 1M     ; Maximum size of POST data that PHP will accept.
php_admin_value[file_uploads] = Off     ; Whether to allow HTTP file uploads.
php_admin_value[upload_max_filesize] = 1M   ; Maximum allowed size for uploaded files.
php_admin_value[display_errors] = Off       ; Do not expose PHP error messages to external users
php_admin_value[safe_mode] = On             ; Turn on safe mode
php_admin_value[safe_mode_exec_dir] = php-required-executables-path     ; Only allow access to executables in isolated directory
php_admin_value[safe_mode_allowed_env_vars] = PHP_        ; Limit external access to PHP environment
php_admin_value[expose_php] = Off        ; Restrict PHP information leakage
php_admin_value[log_errors] = On         ; Log all errors
php_admin_value[register_globals] = Off  ; Do not register globals for input data
php_admin_value[post_max_size] = 1K      ; Minimize allowable PHP post size
php_admin_value[cgi.force_redirect] = 0  ; Ensure PHP redirects appropriately
php_admin_value[sql.safe_mode] = On      ; Enable SQL safe mode
php_admin_value[allow_url_fopen] = Off   ; Avoid Opening remote files
php_admin_value[allow_url_include] = Off ; Avoid Opening remote files
php_admin_value[include_path] = .        ; Smallest path possible
```

# [Impl.] Application Integration



server root

<root>
  /bsd
  /etc
  /root
  /home
  /var

nginx chroot. Run as: www : www

/var/www
  /htdocs
  /etc
  /var
  /run

php chroot. Run as: _www1 : www

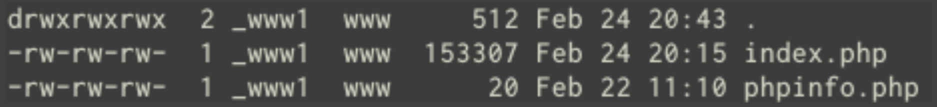
/var/www/htdocs/chal01
  /etc
  /var
  /var/www
  /var/log

php chroot. Run as: _www2 : www

/var/www/htdocs/chal02
  /etc
  /var
  /var/www
  /var/log

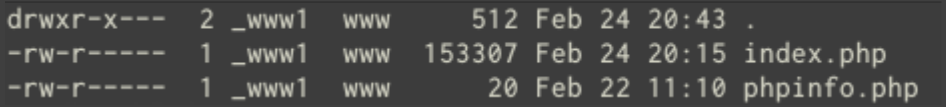
# [Impl.] File/Folder Security?

- Whereas php run as _www1:www and nginx run as www:www
  - Is this acceptable?
    - nop



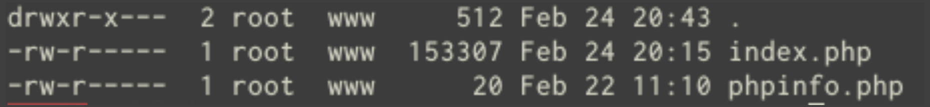
```
drwxrwxrwx  2 _www1  www     512 Feb 24 20:43 .
-rw-rw-rw-  1 _www1  www  153307 Feb 24 20:15 index.php
-rw-rw-rw-  1 _www1  www      20 Feb 22 11:10 phpinfo.php
```

  - Is this enough?
    - nop



```
drwxr-x---  2 _www1  www     512 Feb 24 20:43 .
-rw-r-----  1 _www1  www  153307 Feb 24 20:15 index.php
-rw-r-----  1 _www1  www      20 Feb 22 11:10 phpinfo.php
```

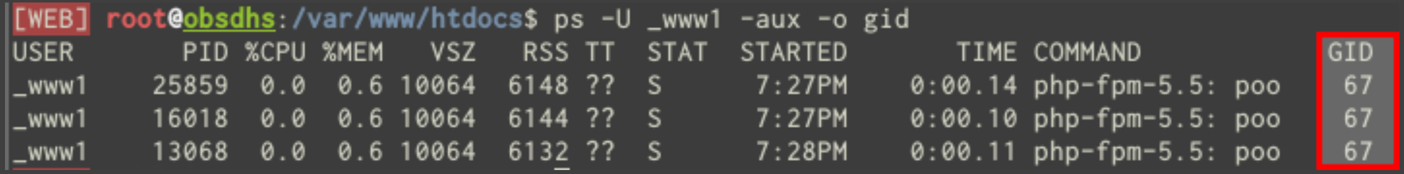
  - Is this secure?
    - yes



```
drwxr-x---  2 root  www     512 Feb 24 20:43 .
-rw-r-----  1 root  www  153307 Feb 24 20:15 index.php
-rw-r-----  1 root  www      20 Feb 22 11:10 phpinfo.php
```

# [Impl.] More Hardening

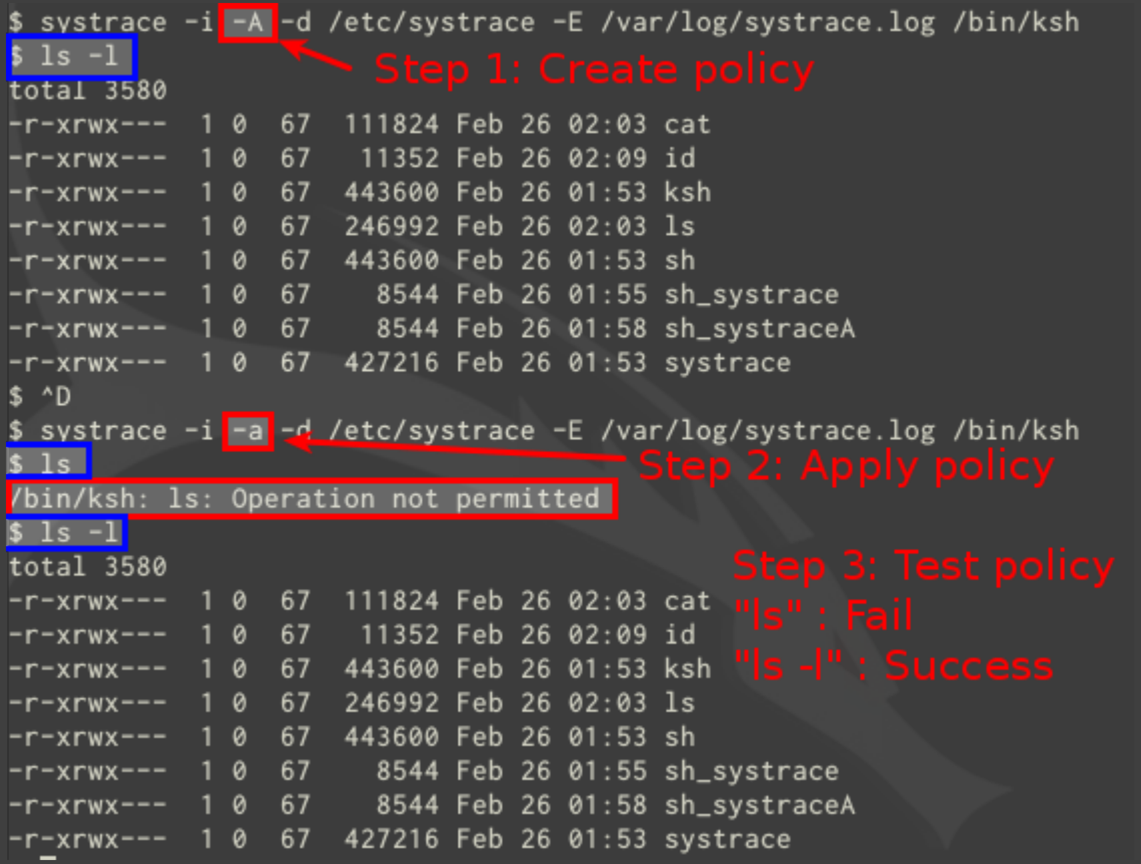
- User / Groups / Files security
  - Make /tmp be writable but not readable
    - chmod 730 tmp
  - Verify that php user (_www1) run as gid 67 (www)



```
[WEB] root@obsdhs:/var/www/htdocs$ ps -U _www1 -aux -o gid
USER        PID %CPU %MEM   VSZ   RSS TT  STAT  STARTED       TIME COMMAND            GID
_www1     25859  0.0  0.6 10064  6148 ??  S       7:27PM    0:00.14 php-fpm-5.5: poo    67
_www1     16018  0.0  0.6 10064  6144 ??  S       7:27PM    0:00.10 php-fpm-5.5: poo    67
_www1     13068  0.0  0.6 10064  6132 ??  S       7:28PM    0:00.11 php-fpm-5.5: poo    67
```

# [Impl.] Still need more?

- ➢ systrace demo



```
$ systrace -i -A -d /etc/systrace -E /var/log/systrace.log /bin/ksh
$ ls -l
total 3580
-r-xrwx---   1 0   67   111824 Feb 26 02:03 cat
-r-xrwx---   1 0   67    11352 Feb 26 02:09 id
-r-xrwx---   1 0   67   443600 Feb 26 01:53 ksh
-r-xrwx---   1 0   67   246992 Feb 26 02:03 ls
-r-xrwx---   1 0   67   443600 Feb 26 01:53 sh
-r-xrwx---   1 0   67     8544 Feb 26 01:55 sh_systrace
-r-xrwx---   1 0   67     8544 Feb 26 01:58 sh_systraceA
-r-xrwx---   1 0   67   427216 Feb 26 01:53 systrace
$ ^D
$ systrace -i -a -d /etc/systrace -E /var/log/systrace.log /bin/ksh
$ ls
/bin/ksh: ls: Operation not permitted
$ ls -l
total 3580
-r-xrwx---   1 0   67   111824 Feb 26 02:03 cat
-r-xrwx---   1 0   67    11352 Feb 26 02:09 id
-r-xrwx---   1 0   67   443600 Feb 26 01:53 ksh
-r-xrwx---   1 0   67   246992 Feb 26 02:03 ls
-r-xrwx---   1 0   67   443600 Feb 26 01:53 sh
-r-xrwx---   1 0   67     8544 Feb 26 01:55 sh_systrace
-r-xrwx---   1 0   67     8544 Feb 26 01:58 sh_systraceA
-r-xrwx---   1 0   67   427216 Feb 26 01:53 systrace
```

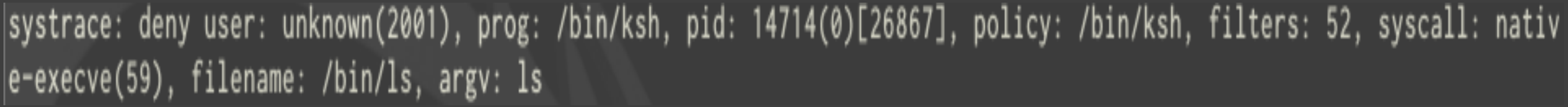
Step 1: Create policy

Step 2: Apply policy

Step 3: Test policy
"ls" : Fail
"ls -l" : Success

# [Impl.] Still need more?

- systrace logs!



```
systrace: deny user: unknown(2001), prog: /bin/ksh, pid: 14714(0)[26867], policy: /bin/ksh, filters: 52, syscall: nativ
e-execve(59), filename: /bin/ls, argv: ls
```

# Pentest Time!

- Challenge #1
  - Find the flag file and read its content
- Challenge #2
  - Exploit an eval()
- URL: http://192.168.1.x/

# Thanks!

# References

- http://www.tldp.org/HOWTO/Chroot-BIND-HOWTO-2.html
- http://www.cyberciti.biz/tips/linux-unix-bsd-nginx-webserver-security.html
- http://cromwell-intl.com/cybersecurity/syslog-tls-cloud.html
- http://networkfilter.blogspot.ca/2014/12/security-openbsd-vs-freebsd.html
- http://www.cyberciti.biz/tips/php-security-best-practices-tutorial.html
- http://martin.kleppmann.com/2013/05/24/improving-security-of-ssh-private-keys.html
- http://www.openbsd.org/faq/pf/config.html