235 lines (222 sloc) 5.95 KB

Secure Solaris

Firewall/IPv6

1. Disable IPv6

```
ifconfig -a(6) (to get address)
ipadm delete-ip net0
ipadm create-ip net0
ipadm create-addr -T static -a <IP_ADDRESS> net0/v4
```

2. Enable firewall if possible

```
sudo svcadm enable firewall
svcs -x firewall:default
svcprop firewall:default | grep ^firewall
cd /etc/firewall/pf.conf
sudo rm -f pf.conf
sudo touch pf.conf (firewall won't work with the default configuration, must be changed)
```

3. Add rules for http and ssh and block the rest

```
# sudo vim pf.conf
ext_if = "net0"

set reassemble yes no-df
set skip on lo0

block log all
antispoof for $ext_if
table <bruteforce> persist
block quick from <bruteforce>

pass proto icmp all
pass in proto tcp from any to $ext_if port 22 keep state (max-src-conn 2, max-src-conn-rate 5/3, \
  overload <bruteforce> flush global)
pass in proto tcp from any to $ext_if port 80 keep state (max-src-conn 10, max-src-conn-rate 5/3, \
  overload <bruteforce> flush global)
pass out on $ext_if proto tcp from $ext_if to any port 22
```

```
pass out on $ext_if proto tcp from $ext_if to any port 80
```

Services

1. Check crontab and remove any suspicious cron jobs

```
crontab -l(or e)
cat /etc/crontabs
svcadm disable cron
```

2. Disable sendmail

```
svcadm disable sendmail(-client)
cd /etc/init.d/
./sendmail stop
cd /etc/default
# Edit sendmail or create it
MODE=""
```

3. List processes

```
# View
ps -ef | less
svcs -a
svcs enable/disable
fuser
# Kill
kill <http>
kill <process_id>
```

4. List open ports `netstat -an | less`
    - Make sure only 22 and 80 are open
5. Use SSH public key based login

```
sudo svcadm disable ssh
ssh-keygen -t rsa -b 4096 -f ~/.ssh/id_space_force
cd .ssh/
cat id_space_force.pub > file
sudo mv file authorized_keys
```

6. Disble root login and other security measures

```
# /etc/ssh/sshd_config
PermitRootLogin no
ChallengeResponseAuthentication no
PasswordAuthentication no
UsePAM no
AuthenticationMethods publickey
PubkeyAuthentication yes
AllowUsers <allowed users>
PermitEmptyPasswords no
```

```
IgnoreRhosts yes
```

7. Enable ssh

```
sudo svcadm enable ssh
```

Users on system

1. Remove any uneeded users or suspicious accounts

```
who -ua
w
kill -9 -u <user>
kill -9 <uid>
```

2. If an account is needed, add or update the password (frequently)
3. Look for accounts without passwords

```
cat /etc/shadow | awk -F: '($2 == "") {print $1}'
cat /etc/shadow | awk -F: '($2 == "") {print $1}' > ~/no_password_users.txt
```

AV

1. Install ClamAV and definitions and scan machine

```
wget https://www.clamav.net/downloads/production/clamav-0.101.1.tar.gz
tar -xf clamav
cd clam/
./configure && make && make install
```

2. Scan machine

Secure Wordpress/MySQL

MySQL

1. Run mysql_secure_installation

```
locate mysql | grep secure
/path/to/secure_installation and configure securly
```

2. Remove any uneeded users and databases

```
SELECT User FROM mysql.user; or SELECT * FROM mysql.user;
DROP USER user;
SHOW DATABASES;
DROP DATABASE table;
```

3. Change all passwords of accounts left
    - Change the password in the wp-config.php file as well

```
UPDATE mysql.user SET PASSWORD=PASSWORD('password') WHERE user="username" AND Host="hostname";
```

4. Restart MySQL

```
sudo svcadm restart mysql
```

Wordpress

1. Login as admin, add new user with admin privieleges, and delete admin account
2. Remove any users that are not needed
3. Change passwords of users that are needed
4. Remove any uneeded plugins (mostly all of them)
    - Recent Backups
    - WP Symposium Pro
    - WPTF Image Gallery
    - Google MP3 Audio Player
    - More here
5. (Maybe) Log out and move wp-admin/ folder to another location, or rename it so it is not on the server and rename wp-login.php

```
sudo mv wp-admin/ ~
sudo mv wp-login.php sfxli.php
```

6. Create a file called .htaccess in /etc/apache2/2.x/htdocs (or wherever wordpress is located)

```
# Add this
AuthType Basic
AuthName "Login"
AuthUserFile /export/home/<user>/.htpasswd
require valid-user
```

7. Edit /etc/apache2/2.x/httpd.conf and delete any uneeded modules, remove page indexes, and hide Apache and PHP versions

```
ServerSignature Off
ServerTokens Prod

<Directory />
    AllowOverride None
    Require all denied
</Directory>
DocumentRoot "/var/apache2/2.x/htdocs"
<Directory /var/apache2/2.x/htdocs>
    Options -Indexes
    AllowOverride AuthConfig
```

```
    Require all granted
</Directory>

# If we know IP address of score bot add this in .htaccess file

Order Deny,Allow
Deny from all
Allow from <scorebot_ip>
Satisfy Any
```

8. Create a password using the `htpasswd` command in the home directory

```
htpasswd -c .htpasswd <user>
```

9. Restart http

```
sudo svcadm restart http
```

10. Install ModSecurity and add these rules

```
SecAction phase:1,nolog,pass,initcol:ip=%{REMOTE_ADDR},initcol:user=%{REMOTE_ADDR},id:5000134
<Locationmatch "/wp-login.php">
# Setup brute force detection.
# React if block flag has been set.
SecRule user:bf_block "@gt 0" "deny,status:401,log,id:5000135,msg:'ip address blocked for 5 minutes,
more than 10 login attempts in 3 minutes.'"
# Setup Tracking. On a successful login, a 302 redirect is performed, a 200 indicates login failed.
SecRule RESPONSE_STATUS "^302" "phase:5,t:none,nolog,pass,setvar:ip.bf_counter=0,id:5000136"
SecRule RESPONSE_STATUS "^200"
"phase:5,chain,t:none,nolog,pass,setvar:ip.bf_counter=+1,deprecatevar:ip.bf_counter=1/180,id:5000137"
SecRule ip:bf_counter "@gt 10"
"t:none,setvar:user.bf_block=1,expirevar:user.bf_block=300,setvar:ip.bf_counter=0"
</locationmatch>
```

11. If all else fails move wp-login to a new file, copy all the content from wp-login into it, and replace all instances of wp-login within the file with the new file name

```
mv wp-login.php wp-login.php.old
cp wp-login.php.old new.php
vim new.php > :%s/wp-login.php/new.php/g
```

Log management

Wordpress page

```
tail -f /var/apache2/2.x/logs/access_log
```

System

```
tail -f /var/log/syslog
tail -f /var/cron/log
```

Network

```
ipstat
tcpdump -n -e -ttt -r /var/log/pflog
tcpdump -n -e -ttt -r /var/log/pflog port 80
```