Linux Checklist: First Line of Defense

Set up initial firewall [iptables, ipfilter, ipfw, PF] allowing traffic on specific ports

|  |  |  |
| --- | --- | --- |
| Phase 2: Initial Defense | | |
| Purpose | Command | Checked? |
| Make firewalls persistent  Edit /etc/rc.local z  Make sure rc.local has execute permissions | iptables-save > iptables.rules  iptables-restore < /path/to/iptables.rules |  |
| Verify firewall and service accessibility from another machine | Use appropriate service check from [page number] |  |
| Backup important directories and database | sudo mkdir /data  tar -czf /data/back.tar.gz \ {/home,/etc,/var/www/html/}  mysqldump -u root -p --all-databases > dump.sql |  |
| Hash backups | find /data/back.tar.gz -type f -exec md5sum {} >> backup.tar.gz.md5 \; |  |
| Create password protected zip file for backups and sensitive files (userpass.csv) | zip --encrypt file.zip files |  |
| Install tools - tools in **bold** hold more importance | apt-get install **fail2ban nmap iptstate nikto** tshark clamav rkhunter sophos  pip install **pyinotify** |  |
| Alias commands and log on usage in .bashrc  To apply to all users, create a shell-script in /etc/profile.d/  Log user commands to bash\_history | vi 00-userprof.sh  alias gcc=’echo `date`" gcc" >> /var/log/alias.log && exit’  shopt -s histappend  export PROMPT\_COMMAND=”history -a; history -c; history -r; $PROMPT\_COMMAND” |  |
| Check for backdoors in /etc/rc.local  /etc/init.d/  /etc/rc3.d/  and dot files | cat /etc/rc.local  ls /etc/init.d/  ls /etc/rc3.d/  cat /home/{.bashrc, .login, .profile} |  |
| Check /etc/sudoers  and  /etc/ssh/sshd\_config for vulnerable options | NOPASSWD  PermitRootLogin no  AuthorizedKeysFile /strange\_location |  |
| If hosting SSH, limit remote login only to non-sudo users  Edit /etc/ssh/sshd\_config, then restart the service | MaxAuthTries 3  AllowUsers user1 user2  service ssh restart |  |
| Get base list of SUID files | find / -user root -perm 4000 2>/dev/null  find / -perm /+s |  |

|  |  |  |
| --- | --- | --- |
| Service Initial Lockdown | | |
| Purpose | Command | Checked? |
| Set mySQL bind to localhost | mysqld --help --verbose  //This will get the configuration file location ie:[/etc/mysql/mysql.conf.d/mysqld.cnf]  //Make sure the following option is 127.0.0.1 not 0.0.0.0  [mysqld] bind-address = 127.0.0.1 |  |
| Find web application configuration file  List of common web application’s default configuration file locations | grep -r -l -F “password” /var/www/html/  https://my.bluehost.com/hosting/help/2360 |  |
| Change database user password in both the web application’s configuration file and in mysql | // Change wordpress database user password in both config and mysql:  sed -i \ ‘s/password\_here/P@ssw0rd/’ \ /var/www/html/wp-config.php  mysql -u root -pP@ssw0rd -e “USE mysql;UPDATE user set password=PASSWORD(‘newPassword’) where user=’wordpress’;flush privileges;” |  |
| Change web admin passwords in mySQL | // These values and the password hashing algorithm will be different depending on the web application:  mysql> update wp\_users set user\_pass=‘$1$KjnibPoV$CkJ7A9Q0BTjw2/O7qPqXC.’ where user\_login=‘admin’; |  |
| Set admin web page to only be accessible via localhost  In wordpress this can be performed by editing /var/www/html/.htaccess and /var/www/html/wp-admin/.htaccess | <Files wp-login.php>  Order allow,deny  Deny from all  Allow from 172.16.49.16  <Files>  <LIMIT GET>  order deny,allow  deny from all  allow from 172.16.49.16  </LIMIT> |  |
| Reduce accessibility of web configuration file | // For a wordpress site, add the following to /var/www/html/.htaccess  # Protect wp-config.php  <Files wp-config.php>  order allow,deny  deny from all  </Files> |  |
| Troubleshooting: Enable .htaccess in /etc/apache2/sites-available/000-default.conf by adding the following after </VirtualHost>  Then restart apache | ....  </VirtualHost>  <Directory /var/www/html/>  Options Indexes FollowSymLinks  AllowOverride All  Require all granted  </Directory> |  |