

SERVER HARDENING:

Server Hardening is the process of enhancing server security through a variety of means which results in a much more secure server operating environment. This is due to the advanced security measures that are put in place during the server hardening process.

Server Hardening Tips & Tricks:

Every server security conscious organization will have their own methods for maintaining adequate system and network security. Often you will find that server hardening consultants can bring your security efforts up a notch with their specialized expertise.

Some common server hardening tips & tricks include:

- Use Data Encryption for your Communications
- Avoid using insecure protocols that send your information or passwords in plain text.
- Minimize unnecessary software on your servers.
- Disable Unwanted SUID and SGID Binaries
- Keep your operating system up to date, especially security patches.
- Using security extensions is a plus.
- When using Linux, SELinux should be considered. Linux server hardening is a primary focus for the web hosting industry, however in web hosting SELinux is probably not a good option as it often causes issues when the server is used for web hosting purposes.
- User Accounts should have very strong passwords
- Change passwords on a regular basis and do not reuse them
- Lock accounts after too many login failures. Often these login failures are illegitimate attempts to gain access to your system.
- Do not permit empty passwords.
- SSH Hardening
 - Change the port from default to a non standard one
 - Disable direct root logins. Switch to root from a lower level account only when necessary.
- Unnecessary services should be disabled. Disable all instances of IRC - BitchX, bnc, eggdrop, generic-sniffers, guardservices, ircd, psyBNC, ptlink.
- Securing /tmp /var/tmp /dev/shm
- Hide BIND DNS Server Version and Apache version
- Hardening sysctl.conf
- Server hardening by installing Root Kit Hunter and ChrootKit hunter.
- Minimize open network ports to be only what is needed for your specific circumstances.
- Configure the system firewall (Iptables) or get a software installed like CSF or APF. Proper setup of a firewall itself can prevent many attacks.
- Consider also using a hardware firewall
- Separate partitions in ways that make your system more secure.
- Disable unwanted binaries
- Maintain server logs; mirror logs to a separate log server
- Install Logwatch and review logwatch emails daily. Investigate any suspicious activity on your server.

- Use brute force and intrusion detection systems
- Install Linux Socket Monitor - Detects/alerts when new sockets are created on your system, often revealing hacker activity
- Install Mod_security as Webserver Hardening
- Hardening the Php installation
- Limit user accounts to accessing only what they need. Increased access should only be on an as-needed basis.
- Maintain proper backups
- Don't forget about physical server security