Server Hardening For Web Applications

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, probably one of the most important tasks to be handled on your servers, becomes more understandable when you realize all the risks involved. The default config of most operating systems are not designed with security as the primary focus. Instead, default setups focus more on usability, communications and functionality. To protect your servers you must establish solid and sophisticated server hardening policies for all servers in your organization. Developing a server hardening checklist would likely be a great first step in increasing your server and network security. Make sure that your checklist includes minimum security practices that you expect of your staff. If you go with a consultant you can provide them with your server hardening checklist to use as a baseline.

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 Sever Version and Apache version
- Hardening sysctl.conf
- Server hardenining 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

BIOS (basic input/output system)

BIOS (basic input/output system) is the program a personal computer's microprocessor uses to get the computer system started after you turn it on. It also manages data flow between the computer's operating system and attached devices such as the hard disk, video adapter, keyboard, mouse and printer.

BIOS is an integral part of your computer and comes with it when you bring it home. (In contrast, the operating system can either be pre-installed by the manufacturer or vendor or installed by the user.) BIOS is a program that is made accessible to the microprocessor on an erasable programmable read-only memory (EPROM) chip. When you turn on your computer, the microprocessor passes control to the BIOS program, which is always located at the same place on EPROM.

What Is the Purpose of BIOS for a Computer?

BIOS enables computers to perform certain operations as soon as they are turned on. The principal job of a computer's BIOS is to govern the early stages of the startup process, ensuring that the operating system is correctly loaded into memory. BIOS is vital to the operation of most modern computers, and knowing some facts about it could help you troubleshoot issues with your machine.

Booting Process

Booting a computer refers to the **process** of powering on the computer and starting the operating system. The **boot process** loads the operating system into main memory or the random access memory (RAM) installed on your computer.

Steps for Booting Process:

- Power Up. The first **step** of any **boot process** is applying power to the machine.
- Power-On Self Test. The next **step** in the **boot process** is called the POST, or power on self test.
- Find a **Boot** Device.
- Load the Operating System.
- Transfer Control.

Unified Extensible Firmware Interface(UEFI)

The **Unified Extensible Firmware Interface** (**UEFI**) is a specification that defines a software interface between an operating system and platform firmware. UEFI replaces the Basic Input/Output System (BIOS) firmware interface originally present in all IBM PC-compatible personal computers, with most UEFI firmware implementations providing legacy support for BIOS services. UEFI can support remote diagnostics and repair of computers, even with no operating system installed.

Advantages:

- Ability to use large disks (over 2 TB) with a GUID Partition Table (GPT)
- CPU-independent architecture
- CPU-independent drivers

- Flexible pre-OS environment, including network capability
- Modular design
- Backward and forward compatibility

Difference Between LVM and RAID

S.No.	RAID	LVM
1.	RAID is used for redundancy.	LVM is a way in which you partition the hard disk logically and it contains its own advantages.
2.	A RAID device is a physical grouping of disk devices in order to create a logical presentation of one device to an Operating System for redundancy or performance or a combination of the two.	LVM is a logical layer that that can be anipulated in order to create and, or expand a logical presentation of a disk device to an Operating System.
3.	RAID is a way to create a redundant or striped block device with redundancy using other physical block devices.	LVM usually sits on top of RAID blocks or even standard block devices to accomplish the same result as a partitioning, however it is much more flexible than partitions. You can create multiple volumes crossing multiple physical devices, remove physical devices without loosing data, resize the volumes, create snapshots, etc
4.	RAID is either a software or a hardware technique to create data storage redundancy across multiple block devices based on required RAID levels.	LVM is a software tool to manage large pool of storage devices making them appear as a single manageable pool of storage resource. LVM can be used to manage a large pool of what we call Just-a-bunch-of-Disk (JBOD) presenting them as a single logical volume and thereby create various partitions for

		software RAID.
5.	RAID is NOT any kind of Data backup solution. Its a solution to prevent one of the SPOFs (Single Point of Failure) i.e. DISK failure. By configuring RAID you are just providing an emergency substitute for the Primary disk. It NEVER means that you have configured DATA backup.	LVM is a disk management approach that allows us to create, extend, reduce, delete or resize the volume groups or logical volumes.