April 17, 2023

Automatic System Hardening: Checklist + Examples to Ensure System Security

SECURITY & COMPLIANCE

By Robin Tatam

The broad umbrella of IT security includes standards, tools, technologies, and human practices that reduce risk and protect your systems. System hardening is one conceptual catch-all for those components of IT security – but what does system hardening mean in relation to your actual day-to-day operations? And how do you achieve system hardening without burdening your whole team?

In this blog, we'll cover common questions about system hardening, explain the types of system hardening often used in IT security, share system hardening examples, define the relationship between compliance and hardening, and highlight how infrastructure as code can automate system hardening across your infrastructure.

Table of Contents

Table of Contents

- What Does System Hardening Mean?
- System Hardening Examples
- A System Hardening Checklist: System Hardening Tips You Can Use
- Who is Responsible for System Hardening?
- How Do You Measure System Hardening?
- How Automation Makes System Hardening Easier (Even at Scale)

Back to top

What Does System Hardening Mean?

In cybersecurity, system hardening means using tools to secure technologies in an IT system. System hardening includes securing servers, networks, apps, databases, and more against vulnerabilities and attack.

System hardening is an important process because it minimizes the attack surfaces of a system, which are often exploited by cyber criminals. The purpose of system hardening is to remediate vulnerabilities before they can be exploited by threat actors, or before they cause business interruptions like an unplanned outage.

Back to top

System Hardening Examples

System hardening isn't just one thing; it's a comprehensive measurement of your system's security. Different technologies require different tactics for hardening.

As such, the system hardening examples shown below have their own nuances and checklists. They're all enacted and measured differently, so they should be evaluated against applicable best practices and mitigated as part of an overall risk management program.

Some system hardening examples include:

Server Hardening	Securing components and permissions of hardware, software, and firmware layers of a system. Can include patching, updating, multi-factor authentication (MFA), and strong password use.
Operating System Hardening (OS Hardening)	Securing the software that grants server permissions to application software. Usually handled with automatic updates and patches, but can also include tasks like unnecessary driver removal, limiting user creation, HDD/SSD encryption, and more.
Network Hardening	Configuring network firewalls, disabling services, auditing access privileges, encrypting traffic, and more. Usually achieved by using intrusion prevention and detection software to monitor suspicious activity and prevent unauthorized network access.
Application Hardening	Patching application code, using antivirus software, encrypting and managing passwords, and using firewalls to secure a server's applications.
Database Hardening	Controlling database privileges, disabling database functions, and encrypting database information. Includes patching the database management system (DBMS), using role-based access control (RBAC), restricting administrative privileges, and more.
Physical Hardening	Preventing access to technology resources in a physical space. Includes intrusion sensors, personnel barriers, and other solutions to harden physical perimeters around system technologies.

The Differences Between System Hardening, Server Hardening, App Hardening + More

All kinds of system hardening are important to maintaining a strong security posture across your infrastructure. A modern network comprises many heterogenous devices and disparate technologies. On top of that, those devices and technologies are managed across a variety of on-premises, private cloud, and public cloud environments. It is reasonable to expect good security hygiene regardless of the type and physical location of the software, firmware, or hardware.

But system hardening activities, while conceptually similar, will differ based on the type, role, or location of the asset. The requirements for an end user's workstation in an office center are going to be quite different than those of a web server operating in the DMZ, for example. Likewise, firewalls and routers will each have their own requirements

example. Likewise, firewalls and routers will each have their own requirements.

Multiple hardening layers may exist within even a single asset. Rules will need to be established and managed for each layer of a system. For example, once a server has been hardened at the OS and firmware level, its applications may then need to be hardened by installing a specific version from only trusted application repositories.

Prioritizing and then systematically hardening each of these elements makes systems more secure. In turn, that reduces the risk of a hacker accessing restricted resources, or the accidental or malicious disruption of critical business services.

Back to top

A System Hardening Checklist: System Hardening Tips You Can Use

A system hardening checklist refers to a list of practices and action steps you can take to ensure system hardening. Here are a few common items you'll find on a system hardening checklist:

- **Require strong passwords.** This can be done using multi-factor authentication (MFA), SIEM, and other security automation tools.
- **Disable unnecessary services.** Tools like Windows Services Manager and SCCM for Linux can help.
- **Whitelist critical applications.** Try Windows Defender, macOS Gatekeeper, Carbon Black, SELinux – or, for scale, use a configuration management tool.
- **Remove or disable unnecessary system applications.** Package managers and configuration management tools make it a lot easier than manually disabling each app.
- **Remove or disable superfluous drivers.** A configuration management system can do this for large IT, but for smaller setups, you can also try custom automation scripts or even use the built-in tools that come with your chosen OS (like Windows Device Manager or Linux command line).
- **Maintain supported OS versions.** A patch management tool or a configuration management system is practically a necessity for environments running multiple OSes.
- **Apply patches for known vulnerabilities.** Do it in a timely manner – as long as it's been tested, don't wait to patch software!
- **Alter unsecure default settings** with configuration management or even policy as code (PaC).
- **Establish appropriate firewall rules** with built-in tools like Windows Firewall, SIEM tools, or configuration management tools like Puppet, Ansible, and Chef.

In reality, system hardening is bigger than a single checklist. But these common items can give you a point from which to work toward hardening your systems. Expect to implement security-centric technologies, such as multi-factor authentication (MFA), privileged user vaulting, and encryption. System hardening requirements often includes establishing a cadence of security activities, including penetration testing and audits.

Back to top

Who is Responsible for System Hardening?

Typically, security teams handle many aspects of system hardening. But IT operations can be instrumental in a more effective, less invasive system hardening approach.

IT operations departments take the lead when it comes to the initial configuration and rollout of technology. If security standards have already been established, then these kinds of system hardening requirements can be incorporated into the build and deployment processes. But what happens when the security configuration is not part of the initial process, or when a change is subsequently made to either the asset or to the baseline?

Because your attack surfaces aren't getting smaller – and neither is the cost of a failed audit.

DISCOVER CONTINUOUS COMPLIANCE

Security teams define and then uphold security standards. However, the frequency with which the security team performs scans may be inconsistent with ops's desire for continuous compliance. Giving ops the ability to swiftly make minor course adjustments with automation – without giving them more work to do or a whole bunch of new tools to learn – can help you achieve continuous compliance and system hardening rather than waiting for the next audit.

Combining the talents of operations and security teams provides an optimal balance. Security staff define and oversee compliance, while operations evaluates current state and realigns configurations along the way.

Back to top

How Do You Measure System Hardening?

Compliance regulations often demand a recurring assessment of how well your systems adhere to a desired state. They also require that you show proof of that adherence as part of self-certification or to satisfy a formal audit. Organizations lacking any formal directives should establish a comprehensive internal security policy to guide their activities.

Hardening is accomplished by first establishing the baseline configuration, which is a declaration of the hardened state of the system. Experts encourage the use of security standards as they provide prescriptive guidance and expert input from cybersecurity specialists. Systems should be initially aligned with the pertinent baseline, and then monitored frequently to expose deviations from a hardened state. Deviations must be addressed quickly to minimize the risk of a vulnerability being exploited.

IT operations teams may experience compliance scans as part of a formal (and often infrequent) audit activity by a different team or department – often relying on expensive external resources. This approach is prone to fire drills and unexpected spikes in remediation activity any time non-compliance is discovered, potentially long after the drift initially occurred.

While formal audits and security scans are beneficial for validating the effectiveness of hardening policies, compliance verification should happen with far greater frequency to reduce the impact on staff while minimizing the risk profile.

How Does Compliance Help Harden Your System?

System hardening should be aligned to reputable frameworks and widely accepted standards that vary by industry. Those include security standards like CIS Benchmarks from the Center for Internet Security and DISA STIG (Security Technical Implementation Guidelines from the U.S. Defense Information Systems Agency).

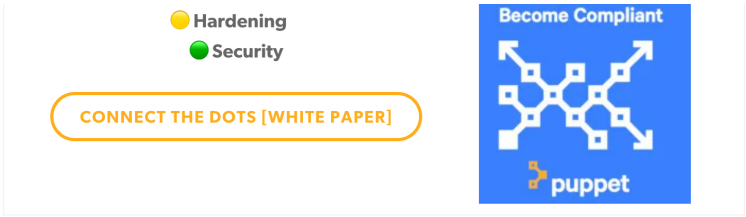
Compliance frameworks and security standards typically provide guidance on system hardening and standardizes the risk evaluation and mitigation process. It also reduces duplication of effort by letting you craft, adhere to, and continually refine a set of rules rather than reinventing the wheel for each new common vulnerability and exposure (CVE).

Back to top

How Automation Makes System Hardening Easier (Even at Scale)

System hardening is a complex and resource-intensive activity. It also requires specialized security skills. Multiply this effort by the large number of IT assets that most organizations are managing, along with the frequency with which they should be reassessed, and it doesn't take long to determine that automation is the only way to achieve success.

For example, think about managing inventory in a modern supermarket. Dozens of aisles, hundreds of shelves, and thousands of items must all be verified for adherence with an item's expected location, accurate pricing, and expiration date.



Supermarkets calculate stock on hand based on purchases that pass through the checkout, combined with an occasional overnight audit usually performed by temporary staff. But how often have you visited a store and seen eggs discarded and spoiling in the cereal aisle? Or found out that the inventory in-store or sale price didn't match what was in their computer?

Doing it all manually would take weeks, and the review would be long out of date before you're done. Now think about doing that messy inventory analysis regularly. It's the same with taking inventory of technology assets. It takes constant care to ensure that…

- Only authorized applications are installed
- Configuration settings are appropriate
- Operating systems are supported and patched
- Critical firewalls have the correct rules applied

As you can imagine, that's practically impossible to ensure without some degree of automation.

Automation eliminates the error-prone and time-consuming effort of performing even a cursory inventory. You may not be able fully to automate system hardening (again, system hardening is a qualitative measurement, not a product or service), but with deep visibility, you can perform in-depth analysis of every aspect of configuration across your system faster and with less human input.

Instead of discovering configuration drift once a year during an audit, you can detect that drift – which could've just popped up or it could've been sitting there for months – within minutes. Issues can even be corrected automatically. In that way, automation helps you achieve hardening of the various components of your system. Then, you can enforce continuous compliance with those vital benchmarks and frameworks – all with automation as the backbone.

Puppet Enterprise is the leading solution for deploying infrastructure, configuring a desired state, and managing infrastructure with confidence at scale. Puppet unearths drift and other vulnerabilities using an integrated edition of the official CIS Configuration Assessment Tool (CIS-CAT), and can enforce system-hardening compliance functions to keep your systems in alignment with CIS Benchmarks and DISA STIGs automatically.

Get a demo of Puppet automation for hardening or download our free ebook on how to achieve continuous compliance now.

DEMO PUPPET COMPLIANCE EBOOK

Back to top

Robin Tatam

Senior Technical Marketer and Evangelist, Puppet by Perforce

Robin Tatam (CISM CPFA CTSP CTMA PCI-P) is a Senior Technical Marketer and Evangelist at Puppet by Perforce, where he promotes the benefits of managing compliance using Puppet. Prior to his role with Puppet, Robin worked as a Security Evangelist, and was a globally recognized SME and five-time IBM Champion. Robin also loves travel and cultural exploration, is an accomplished photographer, and considers himself an amateur mixologist.

SEND FEEDBACK