From Paranoia to Compliance: The Bumpy Road of System Hardening Practices on Stack Exchange

Niklas Busch*, Philip Klostermeyer*, Jan H. Klemmer*, Yasemin Acar†, Sascha Fahl*
*CISPA Helmholtz Center for Information Security, Germany, {firstname.lastname}@cispa.de
†Paderborn University, Germany, yasemin.acar@uni-paderborn.de

***Abstract*—Hardening computer systems against cyberattacks is crucial for security. However, past incidents illustrated that many system operators struggle with effective system hardening. Hence, many computer systems and applications remain vulnerable to security threats. To date, the research community lacks a comprehensive understanding of system operators' motivations, practices, and challenges related to system hardening. With a focus on practices and challenges, we qualitatively analyzed 316 Stack Exchange (SE) posts related to system hardening. We find that access control and deployment-related issues are the most challenging, and system operators suffer from misconceptions and unrealistic expectations. Most frequently, posts focused on operating systems and server applications. System operators were driven by the fear of their systems getting attacked or by compliance reasons. Finally, we discuss our research questions, make recommendations for future system hardening, and illustrate the implications of our work.**

I. Introduction

System hardening is the process of deploying and configuring computer systems to prevent adversaries from gaining access by identifying and addressing vulnerabilities [22]. A study from Rose et al. illustrates that implementing system hardening benchmarks can effectively reduce the risk of security breaches [38]. Although a critical part of the job of system operators, system hardening is challenging and many struggle to deploy effective hardening measures [17, 29, 23]. Hence, companies are regularly affected by successful cyberattacks facilitated by ineffectively hardened computer systems [32, 15, 37, 56, 51].

In 2022, Microsoft experienced a data breach via unauthenticated access to a misconfigured Azure blob storage. As a result, sensitive company data from over 65,000 companies and 548,000 users was compromised [51]. Microsoft faced another severe data breach in 2023 when an attacker successfully stole a signing key that had been transferred from the isolated production environment to the company's Internet-facing network due to another misconfiguration [32]. Amazon Web Services (AWS) customers also face challenges regarding misconfigured services. In 2021, Cosmolog Kozmetik suffered a data breach that exposed the information of approximately 567,000 users due to the incorrect configuration of AWS S3 buckets [15]. Rapid7's *Cloud Misconfiguration Report 2022* revealed that attacks targeting services hosted on AWS are particularly frequent, furthermore highlighting the need for increased vigilance in securely configuring and updating cloud-based infrastructure to protect sensitive data from unauthorized access [37]. Moreover, in 2022, LastPass suffered a breach caused by missed software updates and inadequate hardening processes [56]. Incidents like these highlight the criticality and challenges associated with effective system hardening for enhanced computer security. Previous research explored system hardening from various perspectives, as we elaborate in Section II. However, research lacks an in-depth understanding of system hardening-related practices, challenges, and guidance, as our findings suggest especially the latter to be troublesome. To better understand system operators' practices and challenges and provide actionable recommendations, we performed an in-depth analysis of 316 SE posts while consulting their answers for context to answer the following RQs:

RQ1 *What are common system hardening areas?* System operators need to harden many different systems, services, and applications. We aim to identify the most common domains and security aspects in system hardening.

RQ2 *What are drivers for system hardening?* Motivations for system hardening are manifold. We aim to better understand what drives system operators to strengthen their system and application security.

RQ3 *What are challenges in system hardening?* System or application misconfigurations can have severe security consequences. We are interested in identifying the most pressing challenges for system hardening.

To the best of our knowledge, our study is the first to investigate the challenges of hardening systems based on insights from the SE platforms. Using qualitative analysis, we examine 316 SE posts and identify six main domains and seven security aspects discussed in the field of system hardening, as well as six drivers that led to the creation of each post. We identify seven significant challenges that arise during system hardening and correlate them with relevant domains,

Symposium on Usable Security and Privacy (USEC) 2026
27 February 2026, San Diego, CA, USA
ISBN 978-1-970672-07-7
https://dx.doi.org/10.14722/usec.2026.23055
www.ndss-symposium.org, https://www.usablesecurity.net/USEC/

security aspects, and drivers. Based on our findings, we make suggestions to address the identified challenges. Further, we provide the dataset, including the entire qualitative coding, in our replication package[1].

II. Related Work

We discuss related work in three key areas: (i) security research using SE data, (ii) system hardening with a technical focus, and (iii) human centered research in system hardening. We also contextualize our contributions and highlight the novelty of our work.

System Operator Research with SE Data. SE data, such as Stack Overflow (SO) posts, have been widely used in research on security, privacy, and associated challenges, with most studies concentrating on developers and programming languages. As our focus lies on system operators, we draw on work that explicitly aligns with this perspective. One line of research examines access control issues. For instance, Xu et al. investigated system administrators' difficulties in resolving access-denial issues through a large-scale empirical study of 486 real-world cases, including cases found in posts from SE sites [62]. Their findings highlight recurring security misconfigurations, particularly excessive access privileges granted during troubleshooting, which can lead to substantial security risks [62]. Another relevant direction centers on penetration testing as a means of verifying hardening measures. Rahman et al. analyzed 548 questions from *Information Security SE* to identify practitioners' knowledge needs, revealing concerns about starting points, best practices, and legal aspects [36]. Overall, existing work covers only specific segments and small parts of system hardening, underscoring the need for a broader overview of the field to better understand its core challenges.

Technical System Hardening. Research on system hardening has primarily focused on technical solutions. Operating systems are a central focus, and numerous studies propose hardening techniques to improve OS-level security. These approaches include kernel hardening [1, 27], the use of custom kernel modules [57, 61], and enhancements leveraging features such as Intel SGX [11, 59, 34]. Alongside OS-centric research, practical strategies for hardening modern deployment environments have gained attention, particularly with the rise of container ecosystems. This includes suggestions for strengthening Docker deployments using virtualization, automated testing, deployment tools, and configuration management [4], as well as lightweight container-based approaches for securing cross-domain applications [16]. Additional studies investigate how existing hardening measures from other areas may be adapted to container environments [4, 52, 38].

Hardening guidelines and benchmarks also play a pivotal role, especially in contexts where compliance is legally mandated. Work in this area includes case studies examining how specific benchmarks could have mitigated large-scale security breaches [38] and analyses of Windows hardening efforts based on Center for Internet Security (CIS) controls and security tools [39]. Tools designed to support auditing against extensive benchmark catalogs, such as the 232 CIS benchmarks for Ubuntu [41], reflect the operational relevance of these standards. More recent research identifies practical challenges organizations face when implementing CIS benchmarks, highlighting their complexity and the burden of manual configuration, and proposes automation to improve scalability and reduce errors [20]. Taken together, these studies demonstrate the intricacy of hardening benchmarks but do not dissect which aspects of hardening pose the most significant difficulties or how guidelines relate to the broader landscape of challenges. Moreover, the technical literature does not capture the full spectrum of obstacles encountered when hardening a system.

Human Centered Research in System Hardening. Compared to developers, system operators have been less frequently examined in human centered security research. Existing work shows that misconfigurations represent a recurring problem: a mixed-methods study by Dietrich et al. found that system operators often encounter such issues, with one-third experiencing related security incidents [17]. Research on update management practices identifies further challenges and limitations in administrators' workflows, along with directions for future investigation [25]. Experimental work on HTTPS deployment reveals significant usability barriers for system operators when handling certificate-related tasks [23]. While these studies share our interest in the human factor, they rely primarily on interviews and surveys. Our work instead examines the challenges reported by system operators themselves, thereby expanding the existing body of knowledge. In particular, we show that operators frequently struggle to understand hardening measures and often implement them primarily to meet regulatory requirements. Because guidelines and benchmarks can be difficult to interpret, measures are sometimes applied without a clear understanding of their impact, which in turn fosters errors and misconceptions.

III. Methodology

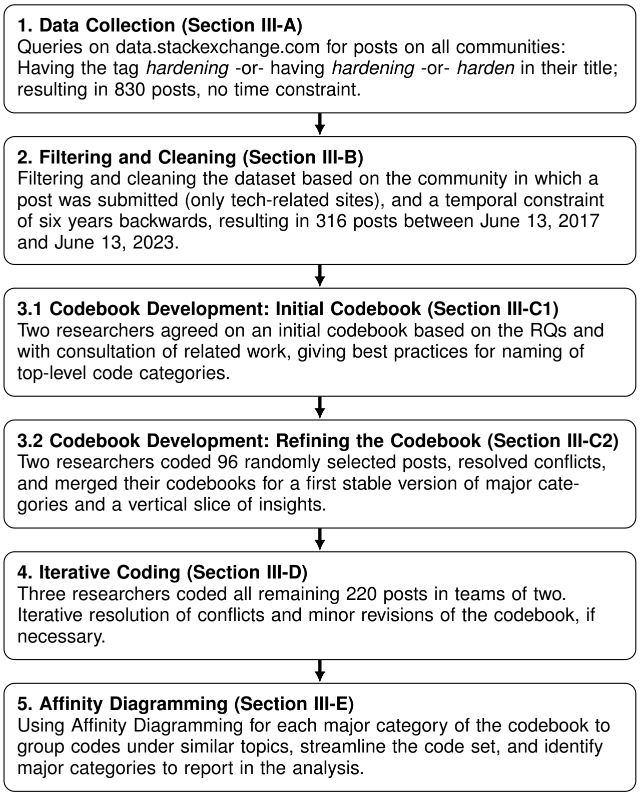
This section describes the data collection and qualitative analysis of 316 system hardening SE posts. We describe the development of our codebook, the coding process, and the affinity diagramming procedure used to identify groups and clusters based on relationships or similarities, which supports our analysis and the reporting of results. Figure 1 summarizes our procedure.

A. Data Collection

We used SE's API *Data Explorer* [42] to collect SE system hardening-related posts.

(1) We collected posts with the tag *hardening*. Tags [47] are used on SE to assign topics to a post and can be specified by the post creator. Alternatively, other users can suggest tags if they have the required reputation level. As tags are optional, not every post necessarily has a tag. On the *Information Security* SE [44] site, the *hardening* tag is described as *"the process of tightening security on a system*" [40], which is in

[1] https://doi.org/10.17605/OSF.IO/5DXAB


1. Data Collection (Section III-A)
Queries on data.stackexchange.com for posts on all communities: Having the tag *hardening* -or- having *hardening* -or- *harden* in their title; resulting in 830 posts, no time constraint.

2. Filtering and Cleaning (Section III-B)
Filtering and cleaning the dataset based on the community in which a post was submitted (only tech-related sites), and a temporal constraint of six years backwards, resulting in 316 posts between June 13, 2017 and June 13, 2023.

3.1 Codebook Development: Initial Codebook (Section III-C1)
Two researchers agreed on an initial codebook based on the RQs and with consultation of related work, giving best practices for naming of top-level code categories.

3.2 Codebook Development: Refining the Codebook (Section III-C2)
Two researchers coded 96 randomly selected posts, resolved conflicts, and merged their codebooks for a first stable version of major categories and a vertical slice of insights.

4. Iterative Coding (Section III-D)
Three researchers coded all remaining 220 posts in teams of two. Iterative resolution of conflicts and minor revisions of the codebook, if necessary.

5. Affinity Diagramming (Section III-E)
Using Affinity Diagramming for each major category of the codebook to group codes under similar topics, streamline the code set, and identify major categories to report in the analysis.

Fig. 1: Methodology of the study.

line with both the definition of system hardening (cf. Section I) and our research questions.

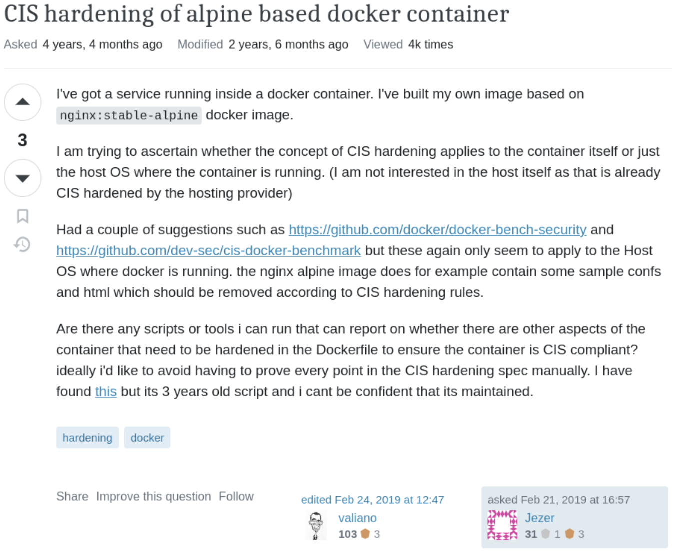
(2) To compensate for untagged posts, we also included all posts containing the word *hardening* or *harden* in their title. We searched for both, as the search terms *harden* and *hardening* resulted in different posts since the SE API searches only for complete words, not substrings.

We carefully chose the present filtering criteria. The chosen filtering has proven to be a good compromise between not missing many posts and maintaining a low false positive rate. First, we tried to use all posts containing the keyword *hardening* or *harden* anywhere in the post, not just the title and tags. This approach resulted in $\approx 2,200$ posts. A random analysis of the posts showed that many posts were off-topic. We also evaluated the tags used beside the *hardening* tag in the posts in the dataset. We analyzed them by frequency and created bigrams and trigrams of tags to identify relevant posts that were not tagged with 'hardening'. However, we skipped this approach since the bigrams and trigrams contained overly generic tags such as *apache*, *operating system*, or *security*.

Our approach is common practice in related research [55, 21, 53, 54, 28]. Overall, we collected a total of 830 posts.

B. Filtering and Cleaning

Two researchers analyzed all posts by using their titles. All posts that seemed off-topic by their title were inspected in detail. This revealed the need for additional filtering. Therefore, we applied the following exclusion criteria.


CIS hardening of alpine based docker container

Asked 4 years, 4 months ago Modified 2 years, 6 months ago Viewed 4k times

I've got a service running inside a docker container. I've built my own image based on `nginx:stable-alpine` docker image.

I am trying to ascertain whether the concept of CIS hardening applies to the container itself or just the host OS where the container is running. (I am not interested in the host itself as that is already CIS hardened by the hosting provider)

Had a couple of suggestions such as https://github.com/docker/docker-bench-security and https://github.com/dev-sec/cis-docker-benchmark but these again only seem to apply to the Host OS where docker is running. the nginx alpine image does for example contain some sample confs and html which should be removed according to CIS hardening rules.

Are there any scripts or tools i can run that can report on whether there are other aspects of the container that need to be hardened in the Dockerfile to ensure the container is CIS compliant? ideally i'd like to avoid having to prove every point in the CIS hardening spec manually. I have found this but its 3 years old script and i cant be confident that its maintained.

hardening
docker
Share Improve this question Follow
edited Feb 24, 2019 at 12:47
valiano
103 3
asked Feb 21, 2019 at 16:57
Jezer
31 1 3

Fig. 2: Example post *"CIS hardening of alpine based docker container"* (204026) from Information Security SE.

(1) Some posts were from non-technical SE sites, like *The Great Outdoors SE* [46]. Upon scrutinizing these posts, it became evident that they did not align with the topic of our study. Consequently, we opted to narrow our search to only technical SE sites.

(2) The search yielded some very old posts, the oldest dating back to May 7, 2009. Upon reviewing older posts, we observed a recurring issue with outdated information or treatment of outdated topics, such as the hardening of particularly obsolete operating systems and versions. Therefore, and in line with similar research [54], we excluded posts older than six years to examine current challenges appropriately.

At the time of collection in June 2023 and after filtering and cleaning, the final dataset included $N = 316$ posts. Figure 2 shows an example post from our dataset.

Table I lists the number of posts and their respective SE site.

C. Codebook Development

In this section, we present the design of our initial codebook based on our research questions and its refinement by coding a random subset of posts from the dataset.

1) Initial Codebook: We followed a semi-open coding approach. Two researchers created an initial codebook based on the RQs and with lessons learned from related work, especially regarding top-level codebook categories [8, 55, 5]. Since our work aims to depict system hardening challenges that system operators discuss online, we included (1) the topic of the post, separated into general *Domains*, such as *Server Application* and security-related *Security Aspects*, such as *Access Control*, (2) the specific *Driver*, if any, in which a questioner expresses why they are asking their question, and (3) the actual *Challenge* a questioner faces. Furthermore, we included additional code categories, such as *Context of Usage*,

TABLE I: Number of posts per SE site.

Platform	Posts
security.stackexchange.com	105
stackoverflow.com	85
serverfault.com	41
unix.stackexchange.com	31
askubuntu.com	9
sitecore.stackexchange.com	6
superuser.com	4
apple.stackexchange.com	4
raspberrypi.stackexchange.com	3
crypto.stackexchange.com	3
networkengineering.stackexchange.com	2
dba.stackexchange.com	2
devops.stackexchange.com	2
drupal.stackexchange.com	2
joomla.stackexchange.com	2
wordpress.stackexchange.com	1
monero.stackexchange.com	1
tor.stackexchange.com	1
ethereum.stackexchange.com	1
webmasters.stackexchange.com	1
softwarerecs.stackexchange.com	1
codegolf.stackexchange.com	1
$\sum$ without Duplicates	308
Duplicates	8
Total	316

Technologies, and *Resources Provided*. These categories aimed to capture whether the post pertains to a professional or private environment, the specific technology involved, and any supplementary resources (e. g., links, CLI commands, or error messages provided by the questioner), respectively.

2) *Refining the Codebook:* With the initial codebook, two researchers coded 96 randomly selected posts in an iterative semi-open coding approach [14, 50, 13]. We coded our observations by mutual agreement, adding, removing, and discussing subcodes, periodically resolving any conflicts, and verifying that the codebook correctly reflected the targeted elucidation of the RQs.

Two of these posts were marked as a *Duplicate* of others on SE. In such cases, SE links to the similar but older original post [48]. Upon closer inspection, we decided to include both duplicate and associated older posts in the dataset. We reasoned that if a question was reposted during our analysis period, its content remained relevant. Furthermore, we marked 13 posts as *Unrelated* when they were not related to hardening in the sense of our work.

D. *Iterative Coding*

Subsequently, three researchers continued coding the remaining 220 of the 316 posts in alternating teams of two. They regularly merged their codebooks, discussed any changes, and resolved conflicts until reaching saturation [10, 58]. We refrain from reporting an intercoder agreement because each conflict was resolved when it occurred [30]. With eight additions from posts marked as duplicates, our final dataset encompasses 316 posts.

E. *Affinity Diagramming*

To finally sort our codebook and identify the most critical topics, we used *affinity diagramming* [7] after completing the coding process. Affinity diagramming is a visual technique that identifies related topics or words from a coding process and sorts them into common categories. Similar to our approach, Beyer et al. [9] used this method to find common topics in a manual categorization of Android application development issues on SO. Using affinity diagramming, we identified six domains, seven security aspects, six drivers, and seven challenge categories.

F. *Limitations*

We only collected posts from Q&A sites in the SE network for the study. We did not consider other Q&A sites, forums, or communities where questions about system hardening might be discussed, such as Reddit or Twitter. We note that since SE provides a public forum where anyone can sign up and ask questions, participants in discussions may be relatively uninitiated in the topic. However, we argue that this is an acceptable trade-off, as SE forms the primary discussion platforms for developers and administrators with large communities. Moreover, the posts on Q&A sites from the SE network have been used as a data source by researchers of various disciplines in the past and have generated many relevant and valuable insights in the past [9, 18, 63, 3, 55, 53, 54, 31].

Our dataset only contains SE posts tagged with *hardening* or having *hardening* or *harden* in their title. Hence, we could have missed posts that were related in content to hardening but were not tagged or named accordingly.

Our analysis only encompasses questions asked in SE posts. Assuming that system operators ask questions on SE only when they seek and cannot find a solution to an issue, we have only insights into publicly discussed issues. However, we might miss problems that are not discussed, e. g., questions that seem too trivial and were therefore not asked at all, or those that system operators could solve themselves without creating a post.

G. *Ethical Considerations*

This study uses only publicly available data from Stack Exchange (SE) platforms. The data was collected in accordance with the platforms' terms of service, which explicitly encourage research use and license content under a Creative Commons Attribution-ShareAlike (CC BY-SA) license [45]. Since on every SE page the footer states *"[. . .] user contributions licensed under CC BY-SA."* ([43]), by posting on the page, users are making the informed decision to distribute their content under a CC BY-SA license. As required, all cited posts are attributed with direct links to the original sources (cf. Table III) [6].

Since the study relies exclusively on publicly accessible content and does not involve interaction with users, intervention, or collection of non-public information, it does not constitute human subjects research. Therefore, we did not consult an Institutional Review Board (IRB) or equivalent ethics committee before conducting our study. However, before publishing our data, we consulted with our organization's ethics office, which approved its publication.

Since only practical questions are typically asked on SE and sensitive information is not shared, there is little risk of users unintentionally sharing personal information. Additionally, this research does not evaluate or judge users based on their questions, nor does it frame them unfavorably. Nevertheless, potential privacy risks were considered. The analysis focuses on aggregated patterns rather than individual users, and no attempts were made to identify, track, or profile posters. No efforts were made to track users across posts or platforms, nor to infer identities beyond what is explicitly visible in the original content. Furthermore, no additional personal data was collected or combined with the dataset from external sources. Data handling and reporting followed principles of data minimization, striking a balance between transparency and respect for user privacy.

IV. Results

This section presents our results, beginning with a description of the dataset and a discussion of the found domains. We furthermore report on discussed security aspects (Section IV-A), drivers for asking questions (Section IV-B), and the system hardening challenges we identified (Section IV-C). Since a post on SE may contain several (sub-)questions, the total number of coded observations does not equal the total number of posts in the dataset.

While we report numbers throughout this paper, we note that this is exploratory qualitative research and the numbers should therefore not be interpreted as quantitative statistical results. Instead, they are intended to give an impression of the weight of a theme. We report both the absolute count and the relative percentage for each domain, security aspect, and challenge across the entire dataset.

About the Dataset. Our final dataset includes 316 posts from between June 13, 2017, and June 13, 2023. As of June 13, 2023 194 of the posts were responded to, from which 96 had an accepted answer [49]. 122 posts were unanswered, excluding comments. We found 24 *closed* posts. Most commonly, the post was closed because the question was already answered in another post (8). We included them in our analysis and eventually added their linked post to the dataset. Other reasons for closing the posts were *needs more focus* (6), *not suitable for this site* (5), *needs details or clarity* (3), and *opinion-based* (3).

Context of Usage. We have identified 72 posts with a professional context and only 20 with a private one. We were unable to assign the context for the other posts.

Domains. System hardening topics were discussed broadly across various domains, including operating systems, server applications, client applications, cloud-related posts, virtualization-related posts, and posts regarding the supply chain. Table II accompanies each domain with its corresponding description and the frequency count indicating how often it has been observed.

A. Security Aspects

The security aspect indicates to which security sub-area a post belongs and helps to understand which security aspects are challenging in system hardening. Posts covered the seven security aspects *Deployment*, *Access Control*, *Guidelines and Benchmarks*, *Networking*, *Security Tooling*, *Malware* and *Cryptography*. Furthermore, eight posts were left without a security aspect assigned. This occurred due to the broad nature of these posts.

Deployment (91, 28.8%). This category contains posts dealing with deploying applications, services, systems, and specific configurations. Comprehensive deployment care is critical for system hardening, e. g., to prevent the creation of attack vectors through insecure initial configurations. We found that operating systems were the most common domain (38), followed by server applications (35). Web servers were discussed, especially within server applications (23). Deploying systems in the cloud (13) or on virtualization hypervisors (12) also kept users busy. Another range of posts on virtualization deployment addressed container systems, such as Docker, or their orchestration (9). The software supply chain, especially package management (8), was another recent topic. Over one-third of the posts came up in a professional context (31).

Access Control (88, 27.8%). One-fourth of the posts concerned access control challenges. The majority were in the domain of operating systems (51). Permissions were the most common access control issue within operating systems (27), and Linux was most frequently mentioned as the operating system used (39). Access control was a significant security aspect within the domain of client applications (11). Browsers were the most frequently mentioned client applications for access control.

Guidelines and Benchmarks (60, 19%). The importance of system hardening guidelines and benchmarks, e. g., due to regulatory requirements, and related challenges for system operators are also reflected in our data. Those serve two primary purposes: first, to ensure compliance with a standard such as the Payment Card Industry Data Security Standard (PCI-DSS) [35]; and second, to outline best practices for hardening specific systems or services, as exemplified by the CIS benchmark. The CIS benchmark describes itself as *"prescriptive configuration recommendations [...][that] represent the consensus-based effort of cybersecurity experts"* ([12]). We found 32 posts that address CIS benchmarks. These guidelines and benchmarks primarily pertain to various operating systems and applications, representing their main focus domains. Operating systems accounted for 29 posts, while server applications constituted 22 posts.

Networking (44, 13.9%). The security aspect of networking contains all posts related to communication between systems or services over a network, including regulation and firewall filtering. The predominant domain in networking was server applications, with 20 occurrences. In contrast to the other security aspects instead of web servers, the posts mainly addressed SSH servers (8), file servers (5) and mail servers (3).

TABLE II: Domains and their occurrence in the dataset.

Domain	Description	Observations	
Operating System	Posts on configuring or administering operating systems and services, including Linux, Windows, macOS, and Android	137	(43.4%)
Server Application	Posts about server applications, such as Web-, Mail-, and File-servers	92	(29.1%)
Cloud	Posts on managing and configuring cloud providers like AWS, Azure, and GCP	24	(7.6%)
Client Application	Posts about client and workstation applications, such as browsers and Office	22	(7%)
Virtualization	Posts on virtualization technologies, such as virtual machines and containers	22	(7%)
Supply Chain	Supply chain related posts, like package management and configuration management (e. g., Ansible)	16	(5.1%)

Firewalling comprised 19 of the posts. Furthermore, 11 posts specifically addressed firewalling on Linux systems. The primary firewall used under Linux, with seven occurrences, was iptables [33] or its successor, nftables [33]. For Windows, three posts related to firewalls.

Security Tooling (32, 10.1%). Posts dealt with tools such as Intrusion Detections Systems (IDS), Intrusion Prevention Systems (IPS), audit tools, or monitoring tools. The domains of security tools are diverse, and we were unable to identify any particular trends. The majority of posts focused on audit (8) or monitoring tools and logging tools (7). These tools were related to guidelines and benchmarks (11), since a goal of their use is either to audit compliance with them or to apply it automatically.

Malware (10, 3.2%). Malware played a relatively minor role in our dataset. These posts primarily focused on the potential risks of malware infections, with some addressing remedial measures for compromised systems. Web servers (6) were the predominant domain. Three of these posts highlighted instances in which WordPress served as the entry point for malware to gain access to the system. One focused on malware's operating mechanisms, and another on whether malware concerns Linux users.

Cryptography (9, 2.8%). A few posts were cryptography-related. Cryptographic cipher suites were the topic in nearly all posts (7), ranging from selecting the best cipher suites to removing deprecated ones, to configuring them. Questions focused on server applications, including SSH-, web-, and mail servers. These posts included additional artifacts, like code snippets from configuration files (6) or links to external websites or documents (6).

> **Summary: Security Aspects**. We identified deployment and access control as essential areas for system hardening. Guidelines emerged as essential, whereas networking, security tooling, malware, and cryptography were less critical.

B. Drivers

Overall, we found seven different drivers that motivated users to seek help on SE. Drivers included fear of attacks, external factors, configuration purposes, privacy, automation, and updates and migrations. We could not identify drivers for 50 posts. System hardening is always performed to enhance the protection of a system or service, thereby preventing potential attacks. Hence, we focused on coding explicitly mentioned drivers.

Fear of Attacks (72, 22.8%). The fear of attacks was the most common driver in our dataset. Past security incidents (18) were most common. Users sought assistance and guidance to address vulnerabilities, remediate past attacks, and mitigate the risk of future attacks: *"My WP site just got hacked for the third time even after following WP hardening guidelines [...] How can I prevent future attacks?"* (231046) Additionally, system operators often mentioned security and audit scans (14) to proactively identify and address potential vulnerabilities. *"We have an Apache [...] in production. The security Audit team found few vulnerabilities lately which needs to be fixed."* (56143561) In related posts, system operators asked to protect their systems against vulnerability scans. *"I am trying to change the default path of the WP default directories such as* wp-content, wp-include *etc to avoid* wpscan*"* (54459236) System operators were also driven to ask questions by various minor types of attacks, including data tampering, privilege escalations, and side-channel attacks.

External (53, 16.8%). External drivers emerged as the second most prominent motivation. Users were motivated by various external reasons to enhance the security of their systems. Compliance with regulations and adherence to guidelines and standards were central reasons for the extrinsically motivated implementation of security measures (46). *"In our company, we want to configure our Windows-based infrastructure compliant to the IASE SCAP specifications, e.g., the Microsoft Windows Server 2016 STIG Benchmark."* (941192) or *"[...] are [there] other aspects of the container that need to be hardened in the Dockerfile to ensure the container is CIS compliant?"* (204026)

We identified further external drivers. Distrust in government served as a driver for system operators, prompting them to enhance their system security proactively, e. g., to protect against intelligence agencies. *"What would be the best practices for securing a single-purpose Windows laptop against a determined foreign intelligence agency from tampering with data on the machine?"* (191469) Other external drivers were user pressure, reading about system hardening best practices, or attending security courses or lectures.

Configuration (40, 12.7%). Configuration revolved i. e.,around implementing security principles, such as least privilege (16), or sandboxing and isolation (6). For applying the least privilege principle, users asked to restrict user privileges, as in *"[...] my requirement is that these root users not have access to data which is located under certain directories."* (222616) or limit the permissions of processes, as in *"The idea is to block everything and allow only what is actually known to be used by the server services."* (1090794).

Other posts focused on reducing the system's attack surface. Therefore, users sought to decrease risk by removing unused software and services or disabling unnecessary features (11). *"how to [...] disable Apache when only using Tomcat?"* (969212) or *"look at attack surface, and remove/deny/disable ereything that an attacker could use to escape the jail or pivot to other networked devices."* (219981)

Additionally, (6) posts discussed inherent distrust in default settings. System operators were driven by a distrust of default settings that could leave their systems vulnerable to attacks, i. e.,*"Microsoft isn't shipping Windows Defender with the strongest settings."* (83606)

Privacy (27, 8.5%). The privacy driver primarily revolved around protecting and hiding sensitive information (19). System operators focused on safeguarding different types of data. This was mainly data an attacker could use to gather details about the target system or network, such as details on software version information. *"I would like to modify Win OS banner to defeat OS detection from scanning tools like* nmap *for example."* (182508) or,

> *"My organization wants to restrict all the plug-ins/tools like Netcraft and Builtwith to detect all the server-side technologies for security reasons like platform, operating system name and version, web server name and version*" — 195359

They also prioritized securing documents and application data, including API interfaces and logs. *"I want to make it as difficult as possible to extract information from my hard drive if stolen or lost as there are sensitive documents and details on it."* (406843) or *"I'd prefer to enable as much logging as possible but secure the access to the logs."* (215398) By implementing privacy-enhancing measures, system operators aimed to ensure the confidentiality and integrity of sensitive information.

Automation (16, 5.1%). The automation driver stems from the desire to automate the hardening process, thereby saving time and effort. Furthermore, automation enables the consistent and efficient deployment of security configurations across multiple systems, reducing the risk of human error and ensuring adherence to security standards. Many system operators seek to automate the implementation of security measures by searching for or developing custom hardening scripts. *"I'm trying to write a hardening script to remove the cron directory in an alpine Linux based docker image."* (955208) Another approach for automatization is utilizing configuration management tools like Ansible *"I want to be able to modify specific local policies on my WS 2019. I've tried to use the win_security_policy module from ansible [...]"* (69471721). These automation efforts often rely on guidelines and benchmarks as their foundation. *"Anyone has any Ansible or other scripts to perform CIS hardening level on the above spec?"* (68112625)

Update and Migration (13, 4.1%). After migrations, changes in the system environment can render existing hardening instructions ineffective or outdated, requiring system operators to adapt and modify their hardening measures accordingly.

> *"I'm working on a hardening task of RHEL 8. The step now is set umask Daemon, I've tried to find /etc/sysconfig.init file to add umask 027 but it's not exist likes RHEL 7. Where can I config this umask on RHEL 8?*" — 73808977

Similarly, system updates aimed at improving security can sometimes cause unexpected issues or conflicts that affect the functionality of specific components. *"[...] everything has been working fine up until a while ago after an update [...]"* (645077)

> **Summary: Drivers**. Attack prevention or remediation was the most common driver in our dataset. The second most pressing driver was compliance, e. g., to hardening standards like the CIS benchmark.

C. Challenges in System Hardening

This section presents the categories for characterizing the seven system hardening challenges we identified. We connect them accordingly, as they can relate to challenges, domains, security aspects, and drivers. Our findings range from *Curiosity* about needing explanations at a higher level to requesting *Assistance* with implementing or executing a specific task and *Troubleshooting*.

Question Triggers in System Hardening. We identified three triggers for asking questions on system hardening, namely (1) *Curiosity*, (2) *Assistance*, and (3) *Troubleshooting*, as shown in Figure 3. This is likely due to the overall hardening process, which begins with general questions that arise during the search for information on hardening measures, followed by more specific questions about implementing those measures, and concludes with highly concrete questions about encountered errors and problems. These triggers are characterized by a decreasing level of abstraction, i. e., *Troubleshooting* questions are typically very concrete compared to more general questions triggered by *Curiosity*. We also added the identified challenges to Figure 3 to indicate which stage a challenge typically occurs.

The first trigger, *Curiosity*, describes questions that arise during the discovery phase: users are curious about a topic and tend to ask higher-level, theoretical, and abstract questions. These users' challenges can be described as a lack of general knowledge or difficulty in searching for resources to gather initial information. The second trigger, *Assistance*, refers to questions that are more concretely related to a specific topic or software. Questioners usually already have a vague to concrete idea of a hardening measure they want to apply, but ask a related question because they need help with it. The challenges associated with this trigger include discussing and weighing

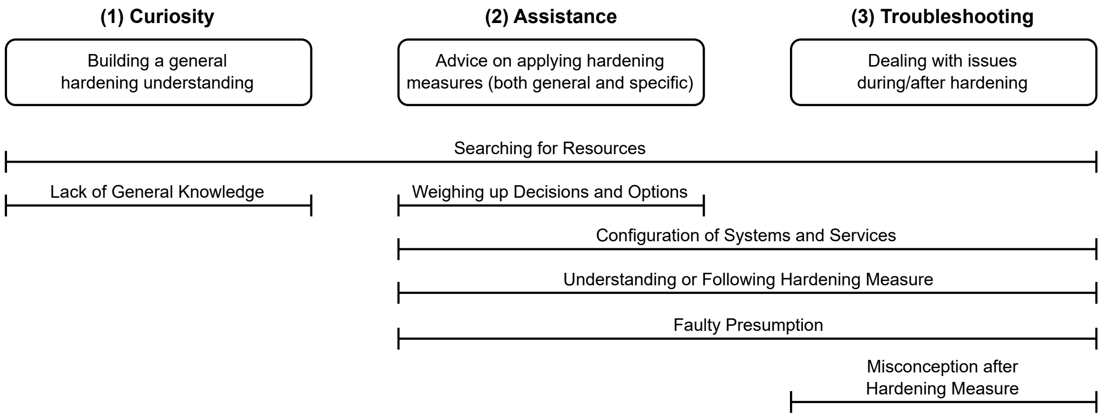

(1) Curiosity
Building a general hardening understanding
(2) Assistance
Advice on applying hardening measures (both general and specific)
(3) Troubleshooting
Dealing with issues during/after hardening
Searching for Resources
Lack of General Knowledge
Weighing up Decisions and Options
Configuration of Systems and Services
Understanding or Following Hardening Measure
Faulty Presumption
Misconception after Hardening Measure

Fig. 3: An overview of question triggers and associated hardening challenges.

options, writing configuration files, and executing the proper steps to apply hardening measures. Hence, it encompasses all questions related to the implementation of hardening measures. The third and last trigger, *Troubleshooting*, consists of questions where a hardening action may have failed. For example, system operators may become stuck when their hardening measure fails to work as expected due to an error or unexpected event.

Co-Occurences. Below, we illustrate the relationships between challenges, domains, and security aspects in our dataset. Figure 4a shows the normalized co-occurrence of security aspects within the challenges, while Figure 4b shows the normalized co-occurrence of challenges within the security aspects. Furthermore, Figure 5 shows the normalized co-occurrence of domains within the challenges. The figures illustrate the relative frequencies at which specific security aspects occur in each challenge. By normalizing the data across the domains, the figure enables a comparative analysis of the prevalence and distribution of security aspects, highlighting the relative importance and prominence of each aspect within the different domains. Below, we report the figure's insights in detail.

Searching for Resources (46, 14.6%). The challenge of searching for resources is unique in that it can occur across all three triggers of system hardening (cf. Figure 3). This challenge involves identifying relevant and reliable resources on system hardening, which is crucial before, during, and after the implementation of hardening measures. These resources encompass both abstract elements, such as formal guidelines and best practices, and concrete components, like system images, tools, and specialized software. The posts often seek resources on specific topics, e. g., Windows Defender: *"[Is] someone knowing a Tool/Software, which is recommended for hardening Windows Defender. [sic!]"* (83606)

Figure 4a indicates that the security aspect of *Deployment* was most commonly associated with this challenge (17). These posts were predominantly related to the *Virtualization* domain, as depicted in Figure 5, e. g., for searching Docker images, like *"Is there any service that provides certified, security-hardened Docker images for common platforms like Python, PHP, Node, Java, etc. with 0 major/critical CVEs. [sic!]"* (195845).

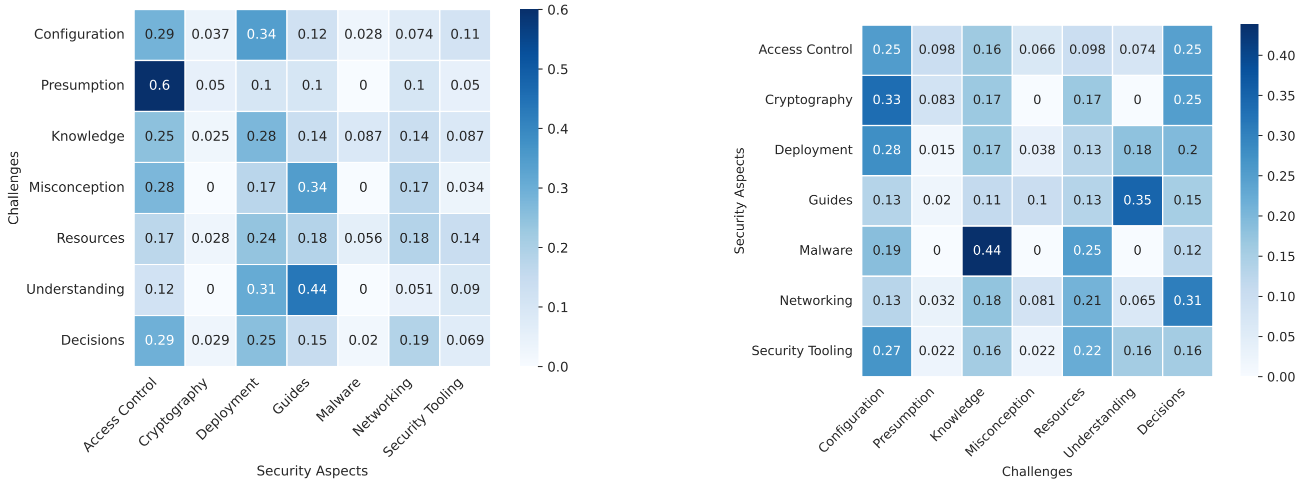
System operators also faced challenges when searching for resources concerning guidelines and benchmarks (13) in combination with security tooling. They typically sought tools to implement security measures outlined in a guide or to perform system audits to comply with guidelines. Examples of such posts include: *"Is there any way to attain a bash script that would allow me to automate the installation of CIS security policy to existing Oracle Linux 8?"* (75371459) or *"I'm trying to find any open source tool or scripts available for direct use to audit the Windows 2019 system against the CIS benchmarks [...]"* (70463445). Therefore, the driver being compliant to some standard (11) was a significant reason to search for resources.

Lack of General Knowledge (67, 21.2%). Some questions seemed to be triggered by *Curiosity* about system hardening (Figure 3) and aim to build a general understanding of system hardening. More than half of the posts were about the domain operating system, mostly covering Linux/Unix. Therefore, many posts were about kernel features and default Unix commands. However, components of the operating system, such as the file system, also frequently appeared. Therefore, specific knowledge of operating systems seems to be a significant challenge for hardening computer systems. Deployment is the *Security Aspect* about which the most questions were asked (cf. Figure 4a), closely followed by *Access Control*. This also aligns with the predominant domain operating system, as many questions addressed specific configuration details.

> *"I am hardening CentOS/RHEL 7.6. The hardening documents recommend disabling the automounter, "unless it is necessary." Why is autofs such a problem? One of the benefits of networking is a shared file system. What other alternatives are there?"* — 210589

In addition, this challenge also contains most posts without an identifiable *Security Aspect* or *Driver*.

Weighing up Decisions and Options (73, 23.1%). The challenge of *Weighing up decisions and options* occurs with



(a) Row-wise normalized co-occurrence of challenges and security aspects.

(b) Row-wise normalized co-occurrence of security aspects and challenges.

Fig. 4: Visualization of the relative frequencies and associations between Security Aspects and Challenges and vice versa using row-wise normalization, to give insights into their co-occurrence dynamics within the dataset. Challenge abbreviations: *Configuration:* Configuration of Systems and Services, *Presumption:* Faulty Presumption, *Knowledge:* Lack of General Knowledge, *Misconception:* Misconception after Hardening Measure, *Resources:* Searching for Resources, *Understanding:* Understanding or Following Hardening Measures, *Decisions:* Weighing up Decisions and Options, *Guides:* Guidelines and Benchmarks.

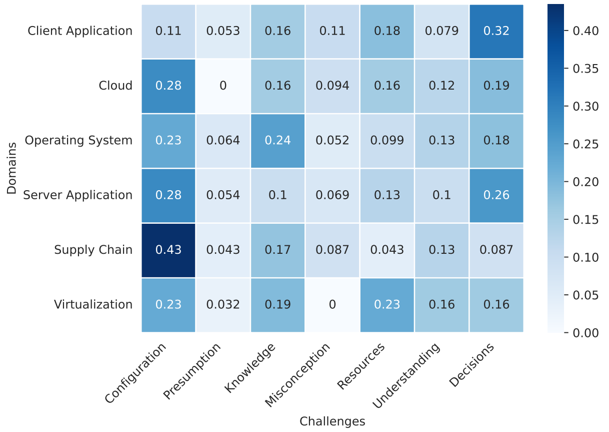


Fig. 5: Row-wise normalized co-occurrence of Domains and Challenges

questions that were triggered due to requesting *Assistance* in system hardening (Figure 3). Users often expressed an understanding of their goal but required assistance to make specific decisions or choose between multiple options.

This challenge covers diverse domains. Most posts covered access control. The most significant uncertainty in decision-making concerned permissions.

> *“We are running a Spring Boot application that we start up with a simple* ’java -jar jarFile’*, and the image is built using maven’s dockerfile-maven-plugin. With that being said, should I be changing the user to an unprivileged user before running that[...]?”* — 57731428

Furthermore, networking, especially firewalls, appeared frequently (Figure 4a). Here, alternative firewalls were sought, or the question arose whether additional firewall rules would increase security: *“I wonder if it is useful also to set the policies to DROP for mangle, raw, and security tables (not nat table because it does not work) to more secure the server?”* (487876) These findings align with the driver *Fear of Attacks* mentioned in 25 posts. In 14 cases, this fear of attacks resulted from previous security incidents.

Configuration of Systems and Services (79, 25%). This challenge involves writing or finding configurations for hardening operating systems, network components, and services. Therefore, this challenge can be triggered by the need for *Assistance* when the questioner asks for help before or during implementation. Moreover, this challenge also arose in *Troubleshooting* posts when errors occurred, e. g., due to an incorrect configuration.

The domain with the most challenges in this category was server applications (Figure 5). Web servers were by far the most common. Deployment was the most common security aspect covered in 34 posts (Figure 4a):

> *“How would I completely disable Apache, since we aren’t using it at all? If I do the above would it impact tomcat in any way? (I’m assuming not). How would I alternatively keep apache httpd running and just redirect all requests to tomcat? What files should I put these redirect rules in? httpd/conf.d/redirect.conf [. . .]???”* — 969212

The fear of attacks often drove questioners. Being attacked, such as following automated network scans for vulnerabilities, was of particular concern to the questioners:

> *"I am trying to change the default path of the WP default directories such as wp-content, wp-include etc to avoid wpscan. I have tried using plugin would it possible to perform the same using manual techniques. I am using apache as a web server. An example, I have tried:*
> RewriteRule ^cms_plugins/(.+) /wordpress/wp-content/plugins/$1 [L,QSA]" — 54459236

Another driver was the configuration of services to reduce the attack surface or to implement the least-privilege principle.

Understanding or Following Hardening Measure (43, 13.6%). This challenge was triggered by posts related to both *Assistance* and *Troubleshooting*, in which questioners encounter difficulty in comprehending specific hardening measures despite having prior knowledge of their intended implementation. Typically, queries about operating systems or server applications were addressed in this.

This challenge is noteworthy for the high prevalence of security guidelines and benchmarks across numerous posts, as well as for the inclusion of external references or links in 35 of these posts (Figure 4a). Most frequently, posts mentioned the CIS benchmark, which often resulted in errors requiring further clarification and assistance:

> *"I'm hardening fedora OS following the CIS benchmark for fedora 28. In one of the remediations, the Benchmark provides a script that modifies the files* system-auth *and* password-auth*. When I apply the changes with* authselect apply-changes *I get an error because the files were modified. Supposedly I can modify these files, but I'm not understanding how to commit the changes. I've been searching about this but stilling stuck. [sic!]*" — 1018828

Furthermore, certain users acknowledged their need for more security proficiency. Users mainly were extrinsically motivated due to reaching compliance with a standard – e.g. the CIS benchmark: *"First of all I would like to say I'm not a Linux/Solaris guy, but just assigned task to look at 1 particular item in the hardening checklist, so thinking to seek help here to understand more."* (899084)

Faulty Presumption (17, 5.4%). This challenge contains posts triggered by some incorrect assumptions. Users requested for *Assistance*, or *Troubleshooting*.

Thematically, these issues primarily addressed operating system challenges (11 posts). Access control was most frequent in 12 questions (Figure 4a). Often users misunderstood how certain technologies worked and how they could be implemented, like in this SE post:

> *"How to protect gnome-terminal or any shell with a password and maybe something like recaptcha... It could consult a shadowed password database or require the user to log in like tty*" — 1460813

Users, like the one above, were often driven by the fear of attacks.

In this challenge, many questioners (7) indicated a professional background.

Misconception after Hardening Measure (19, 6%). This challenge addresses issues where users do not understand what a hardening action has done and why errors occur. Posts were triggered by *Troubleshooting* due to something unexpected occurring after the questioner tried to apply a hardening measure.

Most challenges occurred in the operating systems domain. Windows has an above-average share of 33.3%. With 10 questions, the most relevant security aspect was guidelines and benchmarks (Figure 5). This indicates that many guidelines are implemented blindly, without being aware of what individual actions do:

> *"I am enforcing a hardening policy on my organization's workstations. One of the policies I removed [...] is called "Allow system to be shut down without having to log on". Users started to complain and asked us to re-enable this policy, and I tend to agree. Can you think of a good reason why to disable?*" — 199246

The quote shows that hardening recommendations for server systems have been applied to client systems. Another security aspect that came up more often than average was firewalling under networking. Users implemented firewall rules without properly understanding their consequences and then tried to investigate root causes for errors, such as:

> *"As part of a "Hardening" task, I need to run*
> iptables -P INPUT DROP
> iptables -P OUTPUT DROP
> iptables -P FORWARD DROP
> *on our servers. Normally we would run this command and then run to implement the new policy. However, as soon as I ran* iptables -P OUTPUT DROP *my SSH disconnected. Is this due to the OS being RHEL? How do I configure this machine to allow my IP address through?*" — 1027188

The most frequent driver was being compliant to some standard. This illustrates that system operators may not fully understand guidelines and benchmarks such as CIS, and consequently struggle to apply hardening measures effectively – ultimately resulting in unexpected problems that occur after implementation attempts.

> **Summary: System Hardening Challenges**. The two major challenges were configuring systems and services and weighing up hardening decisions and options. Overall, some questioners have clear hardening objectives but need help accomplishing them.

V. Discussion

This section deepens the interpretation of our findings by situating them within prior research on system hardening, reflecting on their implications, and explicitly grounding our recommendations in the challenges identified in the results (Section IV).

A. Positioning the Findings in Prior Work

Our results both confirm and extend existing research on system hardening and human-centered security. Prior studies have repeatedly highlighted misconfigurations as a dominant source of security incidents [17, 26]. Our analysis confirms these findings, but further refines them by showing that misconfigurations are not isolated technical errors. Instead, they frequently emerge at the intersection of deployment complexity, access control decisions, and the application of hardening guidelines.

Unlike technical hardening research that focuses on prescriptive solutions such as kernel hardening or container isolation [1, 4], our findings show that system operators often struggle to apply such measures meaningfully. In particular, the identified challenges *Understanding or Following Hardening Measure*, *Faulty Presumption*, and *Misconception after Hardening Measure* highlight the need for in-depth knowledge of the measures involved in system hardening. These challenges also reveal that system operators often make false assumptions and are surprised by the results of the measures they implement.

Our work also complements prior qualitative insights from interviews and surveys [17, 24] by drawing on naturally occurring issues on Stack Exchange. This perspective illustrates not only what operators struggle with but also what they actively seek help with in practice. Notably, our results show a stronger emphasis on compliance-driven hardening than previously reported, suggesting a shift from security-as-risk-management toward security-as-obligation.

B. Interpreting the Identified Challenges

A key contribution of our work is distinguishing between different categories of system hardening challenges and relating them to hardening process stages. While prior work often treats hardening difficulties as a monolithic problem, our results show that challenges span conceptual understanding, concrete implementation, and post-deployment troubleshooting.

For example, challenges related to *Lack of General Knowledge* and *Searching for Resources* primarily arise during early exploratory phases. These challenges suggest that system hardening is frequently initiated without a coherent mental model of threats, mitigations, or trade-offs. In contrast, the challenges *Configuration of Systems and Services* and *Weighing up Decisions and Options* dominate later stages and are closely tied to access control and deployment tasks, particularly in operating systems and server applications.

The prominence of *Misconception after Hardening Measure* is especially noteworthy. These cases illustrate that system operators often apply recommendations from benchmarks such as CIS without fully understanding their scope or assumptions, leading to unexpected system behavior.

C. Grounding Recommendations in the Empirical Findings

We base our recommendations directly on the challenges identified in the analysis and are intended to mitigate specific, recurring system hardening challenges.

Secure Default Settings. The challenges *Configuration of Systems and Services*, *Faulty Presumptions*, and *Misconception after Hardening Measure* show that many questions arise from insecure or ambiguous defaults, requiring operators to make complex decisions early in the hardening process, often without sufficient expertise. Overall, our findings suggest a need for enhanced security, often because default security settings were deemed sufficient. Hence, we recommend that applications should be pre-configured with strong, secure defaults. For instance, we found that the default cipher suites do not align with best practices for SSH servers and TLS deployments on web servers, such as Apache [19]. Consequently, SE users asked for support to disable outdated and insecure TLS cipher suites or, even worse, might be completely unaware of this security problem. Providing secure default settings could prevent these pitfalls. If needed, these secure defaults can still be explicitly changed and downgraded, e. g., to ensure compatibility with legacy systems. Moreover, applications could ship with different configurations tailored to various security levels, like in the Mozilla SSL Configuration Generator [19]. The advantage of secure defaults is that they are proactive and require little to no user interaction or security expertise. A proactive and user-friendly approach to system hardening can be achieved by emphasizing secure defaults and providing customization options. By shifting security-critical decisions into well-designed defaults, systems can reduce the need for ad-hoc hardening and lower the risk of misconfiguration.

Documentation. Our findings imply that the availability and accessibility of comprehensive and easy-to-use documentation pose significant challenges. This aligns with prior research on documentation in secure software development [2]. The posts related to the challenges *Searching for Resources* and *Understanding or Following Hardening Measure* indicate that operators frequently rely on external benchmarks because vendor documentation lacks actionable, context-aware guidance Since they are often inaccessible or costly, relying on external resources, such as the CIS benchmark [12], adds another layer of complexity to system hardening efforts in many cases. Hence, we also recommend enhancing the documentation by providing comprehensive, user-friendly guides with actionable recommendations to significantly support system operators in their pursuit of effective system hardening practices.

Security Operation Champions. Most posts related to the challenges *Understanding or Following Hardening Measure* and *Misconception after Hardening Measure* arose in compliance-driven environments, with guidelines and benchmarks as the predominant topics. The presence of recurring misconceptions and blind application of benchmarks suggests that organizational knowledge and expertise are unevenly distributed. Embedding specialized roles that mediate between abstract standards and concrete system contexts can help prevent the propagation of misunderstandings and improve long-term hardening practices. Similar to the role of a security champion in software development teams [60], we therefore recommend a dedicated security operation champion role in system operator teams to support system hardening. They could promote a proactive approach to system hardening and prioritize security considerations throughout the process. These dedicated individuals should have in-depth knowledge of system vulnerabilities and hardening techniques, allowing them to guide and educate other system operators. This may enable organizations to establish a culture of continuous improvement in system hardening, effectively mitigating risks and enhancing their overall security posture.

D. Implications

Our findings indicate that many system hardening failures are not primarily caused by missing security mechanisms, but by usability challenges in applying, interpreting, and maintaining existing ones. This has several implications.

For research, the results highlight the need to place stronger emphasis on the usability of system hardening practices, tools, and guidelines. In particular, research should examine how system operators understand hardening recommendations, how they reason about trade-offs and side effects, and how interfaces and documentation support—or hinder—correct decision-making. Addressing these usability aspects is critical to reducing misconfigurations, misconceptions, and unintended consequences observed in practice.

For practice, the dominance of compliance as a driver indicates that hardening is often performed reactively and under external pressure. This can encourage checklist-based application of benchmarks without a clear understanding of their assumptions or side effects, increasing the risk of fragile configurations. Organizations should treat standards as starting points and invest in internal expertise and review processes to adapt recommendations to their specific system contexts.

Finally, for guideline and tool designers, the prevalence of challenges related to misunderstandings and unintended consequences suggests that current hardening benchmarks are not sufficiently clear. More transparent rationale, scope, and trade-offs could reduce misconceptions and support more sustainable hardening practices.

VI. Conclusion

In this paper, we analyzed 316 Stack Exchange posts related to system hardening. Our qualitative analysis provides a structured overview of the hardening domains, security aspects, and drivers in our dataset. Furthermore, we identified common challenges, revealing where system operators most frequently struggle in practice.

Our findings show that system hardening is dominated by the security aspects deployment and access control, particularly in the domains operating systems and server applications. While fear of attacks motivates some hardening efforts, compliance with external standards and benchmarks emerged as a major driver. However, the widespread use of such guidelines is accompanied by substantial difficulties in understanding, applying, and maintaining hardening measures, often leading to misconfigurations, misconceptions, and unintended side effects.

The results underscore the need for secure default settings, improved system documentation, and user-centric approaches, with security operations champions, to enhance system security.

Acknowledgments

We want to thank the anonymous reviewers for their valuable feedback and for helping us improve this paper. The authors from Hanover would also like to thank our office's feel-good manager, Cleo, for being the cutest office dog and for always providing emotional support.

References

[1] Muhammad Abubakar, Adil Ahmad, Pedro Fonseca, and Dongyan Xu. 2021. SHARD: Fine-Grained Kernel Specialization with Context-Aware Hardening. In *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*. Michael Bailey and Rachel Greenstadt, (Eds.) USENIX Association, 2435–2452.

[2] Yasemin Acar, Michael Backes, Sascha Fahl, Doowon Kim, Michelle L Mazurek, and Christian Stransky. 2016. You Get Where You're Looking For: The Impact of Information Sources on Code Security. In *Proc. 37th IEEE Symposium on Security and Privacy (SP'16)*. IEEE.

[3] Md Ahasanuzzaman, Muhammad Asaduzzaman, Chanchal K. Roy, and Kevin A. Schneider. 2018. Classifying stack overflow posts on API issues. In *2018 IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. (Mar. 2018), 244–254.

[4] Amith Raj MP, Ashok Kumar, Sahithya J Pai, and Ashika Gopal. 2016. Enhancing security of Docker using Linux hardening techniques. In *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*. IEEE, Bangalore, India, 94–99.

[5] Pascal Marc André, Quentin Stiévenart, and Mohammad Ghafari. 2022. Developers Struggle with Authentication in Blazor WebAssembly. Publisher: arXiv Version Number: 1.

[6] Jeff Atwood. 2009. Attribution Required. https://stackoverflow.blog/2009/06/25/attribution-required/. Accessed: 2023-05-05. (2009).

[7] Hugh Beyer and Karen Holtzblatt. 1997. *Contextual Design: Defining Customer-Centered Systems*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[8] Stefanie Beyer, Christian Macho, Martin Pinzger, and Massimiliano Di Penta. 2018. Automatically classifying posts into question categories on stack overflow. en. In *Proceedings of the 26th Conference on Program Comprehension*. ACM, Gothenburg Sweden, (May 2018), 211–221.

[9] Stefanie Beyer and Martin Pinzger. 2014. A Manual Categorization of Android App Development Issues on Stack Overflow. In *2014 IEEE International Conference on Software Maintenance and Evolution*, 531–535.

[10] Melanie Birks and Jane Mills. 2011. *Grounded Theory: A Practical Guide*. (Jan. 2011).

[11] Dorian Burihabwa, Pascal Felber, Hugues Mercier, and Valerio Schiavoni. 2018. SGX-FS: Hardening a File System in User-Space with Intel SGX. In *2018 IEEE International Conference on Cloud Computing Technology and Science, CloudCom 2018, Nicosia, Cyprus, December 10-13, 2018*. IEEE Computer Society, 67–72.

[12] Center for Internet Security (CIS). [n. d.] CIS Benchmarks List. https://www.cisecurity.org/cis-benchmarks. Accessed: 2023-05-05. ().

[13] Kathy Charmaz. 2014. *Constructing Grounded Theory*. SAGE Publications.

[14] Juliet Corbin and Anselm Strauss. 1990. Grounded theory research: Procedures, canons and evaluative criteria. *Zeitschrift für Soziologie*, 19, 6, 418–427.

[15] cyware. 2021. Cosmolog Kozmetik Leaks Half a Million Customers' Data Due to Cloud Misconfiguration. https://cyware.com/news/cosmolog-kozmetik-leaks-half-a-million-customers-data-due-to-cloud-misconfiguration-362e7416. Accessed: 2023-05-05. (2021).

[16] Jason Dahlstrom, Jim Brock, Mekedem Tenaw, Matthew Shaver, and Stephen Taylor. 2019. Hardening Containers for Cross-Domain Applications. In *MILCOM 2019 - 2019 IEEE Military Communications Conference (MILCOM)*. IEEE, Norfolk, VA, USA, (Nov. 2019), 1–6.

[17] Constanze Dietrich, Katharina Krombholz, Kevin Borgolte, and Tobias Fiebig. 2018. Investigating system operators' perspective on security misconfigurations. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 1272–1289.

[18] Felix Fischer, Konstantin Böttinger, Huang Xiao, Christian Stransky, Yasemin Acar, Michael Backes, and Sascha Fahl. 2017. Stack Overflow Considered Harmful? The Impact of Copy&Paste on Android Application Security. In *Proc. 38th IEEE Symposium on Security and Privacy (SP'17)*. IEEE.

[19] Mozilla Foundation. [n. d.] moz://a SSL Configuration Generator. https://ssl-config.mozilla.org/. Accessed: 2023-05-05. ().

[20] Ambika P. H and G. Sujatha. 2024. System Hardening using CIS Benchmarks. In *2024 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*. (May 2024), 1–6.

[21] Nasif Imtiaz, Akond Rahman, Effat Farhana, and Laurie Williams. 2019. Challenges with Responding to Static Analysis Tool Alerts. In *2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)*. IEEE, Montreal, QC, Canada, (May 2019), 245–249.

[22] Intel Corporation. [n. d.] What Is System Hardening? https://www.intel.com/content/www/us/en/business/enterprise-computers/resources/system-hardening.html. Accessed: 2023-05-05. ().

[23] Katharina Krombholz, Wilfried Mayer, Martin Schmiedecker, and Edgar Weippl. 2017. "I Have No Idea What I'm Doing" - On the Usability of Deploying HTTPS. In *26th USENIX Security Symposium (USENIX Security 17)*. USENIX Association, Vancouver, BC, (Aug. 2017), 1339–1356.

[24] Katharina Krombholz, Wilfried Mayer, Martin Schmiedecker, and Edgar R. Weippl. 2017. "i have no idea what i'm doing" - on the usability of deploying HTTPS. In *26th USENIX Security Symposium, USENIX Security 2017, Vancouver, BC, Canada, August 16-18, 2017*. Engin Kirda and Thomas Ristenpart, (Eds.) USENIX Association, 1339–1356.

[25] Frank Li, Lisa Rogers, Arunesh Mathur, Nathan Malkin, and Marshini Chetty. 2019. Keepers of the Machines: Examining How System Administrators Manage Software Updates For Multiple Machines. In *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*. USENIX Association, Santa Clara, CA, (Aug. 2019), 273–288.

[26] Frank Li, Lisa Rogers, Arunesh Mathur, Nathan Malkin, and Marshini Chetty. 2019. Keepers of the machines: examining how system administrators manage software updates for multiple machines. In *Fifteenth Symposium on Usable Privacy and Security ({SOUPS} 2019)*.

[27] Zhenpeng Lin, Yueqi Chen, Yuhang Wu, Dongliang Mu, Chensheng Yu, Xinyu Xing, and Kang Li. 2022. GREBE: Unveiling Exploitation Potential for Linux Kernel Bugs. In *2022 IEEE Symposium on Security and Privacy (SP)*. ISSN: 2375-1207. (May 2022), 2078–2095.

[28] Tamara Lopez, Thein Tun, Arosha Bandara, Levine Mark, Bashar Nuseibeh, and Helen Sharp. 2019. An Anatomy of Security Conversations in Stack Overflow. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*. (May 2019), 31–40.

[29] Alexandra Mai, Oliver Schedler, Edgar Weippl, and Katharina Krombholz. 2022. Are HTTPS Configurations Still a Challenge?: Validating Theories of Administrators' Difficulties with TLS Configurations. In Springer-Verlag, Berlin, Heidelberg, 173–193.

[30] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. 2019. Reliability and inter-rater reliability in qualitative research: norms and guidelines for cscw and hci practice. *Proceedings of the ACM on Human-Computer Interaction*, 3, CSCW, 1–23.

[31] Na Meng, Stefan Nagy, Danfeng (Daphne) Yao, Wenjie Zhuang, and Gustavo Arango Argoty. 2018. Secure coding practices in Java: challenges and vulnerabilities. en. In *Proceedings of the 40th International Conference on Software Engineering*. ACM, Gothenburg Sweden, (May 2018), 372–383.

[32] MSRC. 2023. Results of Major Technical Investigations for Storm-0558 Key Acquisition. https://msrc.microsoft.com/blog/2023/09/results-of-major-technical-investigations-for-storm-0558-key-acquisition/. Accessed: 2023-09-14. (2023).

[33] netfilter.org. [n. d.] Netfilter – firewalling, NAT, and packet mangling for Linux. https://www.netfilter.org/. Accessed: 2023-05-05. ().

[34] Wojciech Ozga, Rasha Faqeh, Do Le Quoc, Franz Gregor, Silvio Dragone, and Christof Fetzer. 2022. CHORS: hardening high-assurance security systems with trusted computing. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing* (SAC '22). Association for Computing Machinery, New York, NY, USA, (May 2022), 1626–1635.

[35] PCI Security Standards Council (PCI SSC). [n. d.] System Hardening Standards for Complying with PCI DSS. https://www.pcidssguide.com/system-hardening-standards-for-complying-with-pci-dss/. Accessed: 2023-05-05. ().

[36] Akond Rahman and Laurie Williams. 2019. A bird's eye view of knowledge needs related to penetration testing. In *Proceedings of the 6th Annual Symposium on Hot Topics in the Science of Security* (HotSoS '19). Association for Computing Machinery, New York, NY, USA, (Apr. 2019), 1–2.

[37] rapid7. 2022. 2022 Cloud Misconfigurations Report. https://www.rapid7.com/info/cloud-misconfigurations-research-report/. Accessed: 2023-05-05. (2022).

[38] Tina Rose and Xiaobo Zhou. 2020. System Hardening for Infrastructure as a Service (IaaS). In *2020 IEEE Systems Security Symposium (SSS)*. IEEE, Crystal City, VA, USA, (July 2020), 1–7.

[39] Rajeshkumar Sasidharan. 2022. A Case Study to Implement Windows System Hardening using CIS Controls. *International Journal of Computer Trends and Technology*, 70, (July 2022), 1–7.

[40] security.stackexchange.com. [n. d.] About hardening - Tag Info. https://security.stackexchange.com/tags/hardening/info. Accessed: 2023-05-05. ().

[41] Wadlkur Kurniawan Sedano and Muhammad Salman. 2021. Auditing Linux Operating System with Center for Internet Security (CIS) Standard. In *2021 International Conference on Information Technology (ICIT)*. (July 2021), 466–471.

[42] Stack Exchange. 2022. Stack Exchange API. https://api.stackexchange.com/. Accessed: 2023-05-05. (2022).

[43] Stack Exchange. 2022. Stackexchange Website. https://stackexchange.com/. Accessed: 2023-05-05. (2022).

[44] Stackexchange.com. [n. d.] Information Security. https://security.stackexchange.com/. Accessed: 2023-05-05. ().

[45] Stackexchange.com. [n. d.] Terms of Service. https://stackoverflow.com/legal/terms-of-service/#licensing. Accessed: 2023-05-05. ().

[46] Stackexchange.com. [n. d.] The Great Outdoors. https://outdoors.stackexchange.com/. Accessed: 2023-05-05. ().

[47] Stackexchange.com. [n. d.] What are tags, and how should I use them? https://meta.stackexchange.com/help/tagging. Accessed: 2023-05-05. ().

[48] Stackexchange.com. [n. d.] Why are some questions marked as duplicate? https://stackoverflow.com/help/duplicates. Accessed: 2023-05-05. ().

[49] Stackoverflow.com. [n. d.] What does it mean when an answer is äccepted?̈ https://stackoverflow.com/help/accepted-answer. Accessed: 2023-05-05. ().

[50] Anselm Strauss and Juliet M Corbin. 1997. *Grounded theory in practice*. SAGE Publications.

[51] Sumeet Wadhwani. 2022. Misconfigured Azure Blob Storage Exposed the Data of 65K Companies and 548K Users. https://www.spiceworks.com/it-security/cloud-security/news/microsoft-azure-cloud-misconfiguration/. (2022).

[52] Yuqiong Sun, David Safford, Mimi Zohar, Dimitrios Pendarakis, Zhongshu Gu, and Trent Jaeger. 2018. Security Namespace: Making Linux Security Frameworks Available to Containers. en. In 1423–1439.

[53] Mohammad Tahaei, Julia Bernd, and Awais Rashid. 2022. Privacy, Permissions, and the Health App Ecosystem: A Stack Overflow Exploration. In *EuroUSEC 2022: European Symposium on Usable Security, Karlsruhe, Germany, September 29 - 30, 2022*. ACM, 117–130.

[54] Mohammad Tahaei, Tianshi Li, and Kami Vaniea. 2022. Understanding Privacy-Related Advice on Stack Overflow. *Proc. Priv. Enhancing Technol.*, 2022, 2, 114–131.

[55] Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. 2020. Understanding Privacy-Related Questions on Stack Overflow. en. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, Honolulu HI USA, (Apr. 2020), 1–14.

[56] The Hacker News. 2023. LastPass Hack: Engineer's Failure to Update Plex Software Led to Massive Data Breach. https://thehackernews.

com/2023/03/lastpass-hack-engineers-failure-to.html. Accessed: 2023-05-05. (2023).

[57] Dave Jing Tian, Grant Hernandez, Joseph I. Choi, Vanessa Frost, Peter C. Johnson, and Kevin R. B. Butler. 2019. LBM: A Security Framework for Peripherals within the Linux Kernel. In *2019 IEEE Symposium on Security and Privacy (SP)*. ISSN: 2375-1207. (May 2019), 967–984.

[58] Cathy Urquhart. 2013. *Grounded Theory for Qualitative Research: A Practical Guide*. (Jan. 2013).

[59] Huibo Wang, Erick Bauman, Vishal Karande, Zhiqiang Lin, Yueqiang Cheng, and Yinqian Zhang. 2019. Running Language Interpreters Inside SGX: A Lightweight, Legacy-Compatible Script Code Hardening Approach. In *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security, AsiaCCS 2019, Auckland, New Zealand, July 09-12, 2019*. Steven D. Galbraith, Giovanni Russello, Willy Susilo, Dieter Gollmann, Engin Kirda, and Zhenkai Liang, (Eds.) ACM, 114–121.

[60] Charles Weir, Ben Hermann, and Sascha Fahl. 2020. From Needs to Actions to Secure Apps? The Effect of Requirements and Developer Practices on App Security. In *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, (Aug. 2020), 289–305.

[61] Chris Wright, Crispin Cowan, Stephen Smalley, James Morris, and Greg Kroah-Hartman. 2002. Linux Security Modules: General Security Support for the Linux Kernel. en. In.

[62] Tianyin Xu, Han Min Naing, Le Lu, and Yuanyuan Zhou. 2017. How Do System Administrators Resolve Access-Denied Issues in the Real World? en. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, Denver Colorado USA, (May 2017), 348–361.

[63] Tianyi Zhang, Ganesha Upadhyaya, Anastasia Reinhardt, Hridesh Rajan, and Miryung Kim. 2018. Are code examples on an online Q&A forum reliable? a study of API misuse on stack overflow. In *Proceedings of the 40th International Conference on Software Engineering* (ICSE '18). Association for Computing Machinery, New York, NY, USA, (May 2018), 886–896.

Appendix

TABLE III: Stack Exchange posts mentioned in this paper.

ID	Title	URL
83606	A recommended Tool/Software for hardening Windows Defender	https://softwarerecs.stackexchange.com/questions/83606
113063	How to disable services on DietPi (debian linux) and harden the security?	https://raspberrypi.stackexchange.com/questions/113063
182508	Modify Win OS banner to avoid OS detection	https://security.stackexchange.com/questions/182508
191469	Securing a Laptop from a Foreign Intelligence Agency	https://security.stackexchange.com/questions/191469
195359	How to restrict plugins/tools like Netcraft and Builtwith to detect server side technologies?	https://security.stackexchange.com/questions/195359
195845	Where to find the security hardened docker images	https://security.stackexchange.com/questions/195845
199246	WIN 10 hardening: Importance of "Allow system to be shut down without having to log on" policy	https://security.stackexchange.com/questions/199246
204026	CIS hardening of alpine based docker container	https://security.stackexchange.com/questions/204026
210589	Why is autofs insecure?	https://security.stackexchange.com/questions/210589
215398	How should you configure PowerShell logs permissions?	https://security.stackexchange.com/questions/215398
219981	Tightly locking down a FreeBSD jail	https://security.stackexchange.com/questions/219981
222616	Restrict privileged users from accessing certain directories on Linux servers with Grsecurity?	https://security.stackexchange.com/questions/222616
231046	My WP site just got hacked for the third time even after following WP hardening guidelines	https://security.stackexchange.com/questions/231046
406843	Extreme hardening of my Macbook pro from physical hacking	https://apple.stackexchange.com/questions/406843
487876	Is it useful to set the policies to DROP for all tables in Iptables?	https://unix.stackexchange.com/questions/487876
645077	hidepid=2 stopped working after an update. Kernel don't suppport "per-mount point"?	https://unix.stackexchange.com/questions/645077
899084	Solaris 11 Auditing, audit_control file cannot be found	https://serverfault.com/questions/899084
941192	Windows 10: Kerberos settings not found	https://serverfault.com/questions/941192
955208	Attempting to delete cron directory in docker gives "Invalid argument"	https://serverfault.com/questions/955208
969212	Apache HTTPD and Tomcat - how to harden and/or disable Apache when only using Tomcat?	https://serverfault.com/questions/969212
1018828	Editing Authselect files	https://serverfault.com/questions/1018828
1027188	iptables policy & saving in RHEL	https://serverfault.com/questions/1027188
1090794	Network Security: Hardening IPv6 on Ubuntu Server?	https://serverfault.com/questions/1090794
1361197	Accounts Expired after CIS Hardening on Ubuntu 20.04 - Workstation Level 1	https://askubuntu.com/questions/1361197
1460813	gnome-terminal/shell hardening security	https://askubuntu.com/questions/1460813
1720993	Network share access denied after STIG/CIS hardening in windows	https://superuser.com/questions/1720993
54459236	Wordpress Default Directory Change	https://stackoverflow.com/questions/54459236
56143561	How to Harden Apache against security vulnerabilities	https://stackoverflow.com/questions/56143561
57731428	How do I prevent root access to my docker container	https://stackoverflow.com/questions/57731428
58132270	Will Memory Tagging Extension be implemented in x86?	https://stackoverflow.com/questions/58132270
68112625	CIS hardening script for windows 2016 server in GCP	https://stackoverflow.com/questions/68112625
69471721	How do you use the win security policy module for something in the local policies section using Ansible?	https://stackoverflow.com/questions/69471721
70463445	Windows 2019 CIS benchmark audit tool/script	https://stackoverflow.com/questions/70463445
73808977	Daemon Umask in RHEL 8	https://stackoverflow.com/questions/73808977
75371459	Oracle Linux 8 hardening with CIS security policy	https://stackoverflow.com/questions/75371459