usenix
THE ADVANCED
COMPUTING SYSTEMS
ASSOCIATION

System Administrators Prefer Command Line Interfaces, Don't They? An Exploratory Study of Firewall Interfaces

Artem Voronkov, Leonardo A. Martucci, and Stefan Lindskog, *Karlstad University*

https://www.usenix.org/conference/soups2019/presentation/voronkov

This paper is included in the Proceedings of the
Fifteenth Symposium on Usable Privacy and Security.

August 12–13, 2019 • Santa Clara, CA, USA

ISBN 978-1-939133-05-2

System Administrators Prefer Command Line Interfaces, Don't They?
An Exploratory Study of Firewall Interfaces

Artem Voronkov
Karlstad University

Leonardo A. Martucci
Karlstad University

Stefan Lindskog
Karlstad University

Abstract

A graphical user interface (GUI) represents the most common option for interacting with computer systems. However, according to the literature system administrators often favor command line interfaces (CLIs). The goal of our work is to investigate which interfaces system administrators prefer, and which they actually utilize in their daily tasks. We collected experiences and opinions from 300 system administrators with the help of an online survey. All our respondents are system administrators, who work or have worked with firewalls. Our results show that only 32% of the respondents prefer CLIs for managing firewalls, while the corresponding figure is 60% for GUIs. We report the mentioned strengths and limitations of each interface and the tasks for which they are utilized by the system administrators. Based on these results, we provide design recommendations for firewall interfaces.

1 Introduction

Firewalls are systems designed to regulate network traffic, and are often the first line of defense in computer networks. The maintenance and configuration of firewalls is the responsibility of system administrators. System administrators have multiple methods available to interact with firewalls, e.g. via a command line interface (CLI), graphical user interface (GUI), or application programming interface (API). Although visualization offers an effective approach to exploring and managing data, the use of GUIs by system administrators is not taken for granted. According to the literature, the main instrument for system administrators is the CLI [2, 9, 18].

In this paper, we examine how system administrators interact with firewalls. The goal of our study is to gain a better understanding of the following questions:

Q1: What firewall interfaces do system administrators use?

Q2: What firewall interfaces do they prefer?

Additionally, we want to gain insights into which of the interfaces are beneficial for which tasks, and what strengths and limitations they have. To answer our research questions, we surveyed 300 system administrators and collected their experiences and opinions of utilized firewall interfaces through an online survey.

Unexpectedly, our results show that 70% of the system administrators work primarily with firewall GUIs, with 60% preferring GUIs as a main instrument. The system administrators mainly choose GUIs because they provide better visual representations of data, are easier to create and modify rules with, and are convenient for occasional use. Relatively few system administrators utilize a CLI as their primary or preferred firewall interface: 24% and 32%, respectively. According to our respondents, the main reasons for choosing command line interfaces are their flexibility, efficiency of use, superior functionality, and performance; aspects in which GUIs are deficient.

The contributions of our work are summarized as follows:

- We conduct an online study on the preferences of system administrators regarding firewall interfaces, with 300 volunteer participants.
- Using the gathered data, we classify and report the main strengths and limitations of CLIs and GUIs.
- We provide insights into tasks in which utilizing a CLI or GUI is advantageous for system administrators.
- We provide some recommendations for designers and developers of firewall interfaces, taking into account the main problems of the two interfaces.

USENIX Symposium on Usable Privacy and Security (SOUPS) 2019.
August 11–13, 2019, Santa Clara, CA, USA.

The remainder of this paper presents a review of work related to our study in Section 2, describes our research methodology in Section 3, and presents the results in Section 4. A discussion of the findings, limitations, and our design recommendations is presented in Section 5. Finally, concluding remarks are provided in Section 6.

2 Related Work

Despite the fact that GUIs are known to be convenient for the presentation of large amounts of information, their use is limited in the field of system configuration, as noted by Mahendiran et al. [10].

Botta et al. [2] and Haber and Bailey [9] reported the results of two independent ethnographic studies describing the routines and activities of system administrators. Haber and Bailey followed the daily work of three system administrators, and reported their preference of CLIs over GUIs owing to their speed, scalability, reliability, transparency and trustworthiness. These findings are in line with the interviews of Botta et al., involving a dozen IT professionals who reported being more comfortable with CLIs than GUIs, especially because of their versatility. Botta et al. also highlighted the reliability problem of GUIs that *"write configuration files that sometimes do not take effect"* and *"write unnecessary, noisy markup into configuration files."*

For a study with 101 participants, Takayama and Kandogan [18] reported that 65% of the participants were primarily CLI users, because CLIs are considered to be more reliable, fast, robust, trustworthy, and accurate. Furthermore, the authors pointed out that trust is critical in the adoption of a technology.

However, system administrators require graphical tools that can facilitate their daily work and make it less error-prone [10]. This is especially relevant for security system administrators, as their work has been demonstrated to be more complex [6].

Recent research has sought to leverage the benefits of information visualization in designing interfaces for network security. Shiravi et al. [15] presented a survey of visualization systems in network security in general, while Voronkov et al. [21] reviewed papers specifically concerning firewalls. The authors of both papers identified limitations of existing visualization techniques and suggested future research directions.

Xu et al. [22] argued that *"system configuration becomes a new human–computer interaction (HCI) problem,"* and that *"classic interface design principles are not sufficient for system configuration."* A variety of research studies [9, 19, 20] have attempted to address these problems and suggest appropriate design principles for system configuration.

Although interface preferences of system administrators have been studied in the literature, the present work represents the first large-scale study investigating firewall interfaces, with 300 participants. Furthermore, we aim to investigate whether there have been changes in preferences, as it has been over 10 years since the studies of Botta et al. [2], Haber and Bailey [9], and Takayama and Kandogan [18] were published. Another important aspect of our work is the qualitative analysis of participants' comments regarding the strengths and limitations of firewall interfaces, as well as tasks in which these interfaces are superior.

3 Methodology

We collected both quantitative and qualitative data on the interactions between system administrators and firewall interfaces through an online survey ($N = 300$). In this section, the methodology and demographics of the participants are described, while the remainder of the quantitative data and qualitative results are presented in Section 4.

3.1 Survey Details

We collected the data through an online survey, which ran for six weeks from April to June 2018.[1] The survey utilized skip logic (also known as branch logic or conditional branching) and consisted of up to 14 questions, four of which were open-ended. The close-ended questions required an answer and we also encouraged the participants to answer the open-ended questions, although these were not mandatory.

The survey consisted of two parts. In the first, we asked the participants about the following aspects of their interactions with firewalls:

- How much time on average they spend working with firewalls.
- Which firewall interface they mainly work with, and which interface they prefer.
- Which tasks are easier with which firewall interface.
- What strengths and limitations those interfaces have.

Only general questions about firewall interfaces were asked in the survey. No questions about specific vendor solutions were included. In the second part of the survey, demographically related questions were asked, such as on age, gender, and expertise.

We kept the survey short to minimize respondent fatigue. The survey took an average of 177 seconds ($SD = 106$, $M = 148$, $Q1 = 101$, and $Q3 = 228$ seconds) of the participants' time to be answered.

Prior to dissemination, the survey was pre-tested with six users. Based on their feedback, a few questions were slightly altered to eliminate some ambiguity in the wording, although no significant changes were necessary. For wider coverage, the survey was translated from the original (English) language into three others (Portuguese, Russian, and Swedish) by bilingual speakers.

---

[1] The survey is available at `https://www.soscisurvey.de/firewall_interfaces/`

3.2 Recruitment and Participants

The participants for the study were recruited using various channels:

1. System administrators' forums. The "Sysadmin" subreddit yielded the majority of our participants.[2] Another contributor was the SysAdmins.ru forum.[3]

2. System administrators' mailing lists. We contacted several system administrators from our professional networks and asked them to distribute the survey via system administrator mailing lists of which they are members.

Of 516 participants that started our online survey, 303 completed it (ca. 59% completion rate). After the quality check, three participants were removed as they filled out nonsensical answers. Table 1 summarizes the demographics of the remaining 300 participants. Our sample is heavily skewed owing to specificity of the target audience (the percentage of female system administrators is known to be very low [1]) and recruitment method. A majority of the participants (approximately 80%) were recruited via the "Sysadmin" subreddit, which led to the sample being more male (only 7.5% of the subreddit members are female [3]) and younger than the general population, owing to the demographics of Reddit users [14]. All participants were volunteers, and no financial compensation was offered.

3.3 Survey Data Analysis

The data were analyzed using a content analysis approach. With this approach, it is possible to analyze data qualitatively at the same time as quantifying it [8].

Two of the authors worked independently and coded participants' responses to the open-ended questions using an initial (open) coding approach [13]. Two coding procedures were performed: one before and one after the final codebook. We utilized NVivo for all coding.[4] NVivo helped us to organize and analyze the qualitative data, i.e. open-ended survey responses. NVivo provides methods to automatically or manually code the data. We used manual coding only, which comprises three approaches: 1) select and code content, 2) drag and drop selected content, and 3) in vivo coding.

After the authors completed the first coding procedure, they met, discussed their codes, consolidated them, and formed a final codebook, which consisted of 230 codes (see Section 6). Using the final codebook during the second coding procedure, 1570 coding references were identified. It is worth mentioning that each answer from a participant can have several different codes associated with it, but at most one instance of a single code.

Table 1: Participant demographics ($N = 300$).

	Metric	Participants
Age	18-24	34 (11.3%)
	25-34	142 (47.3%)
	35-44	86 (28.7%)
	45-54	25 (8.3%)
	55-64	9 (3.0%)
	Prefer not to answer	4 (1.3%)
Gender	Female	3 (1.0%)
	Male	285 (95.0%)
	Other	1 (0.3%)
	Prefer not to answer	11 (3.7%)
Time per week (on average) spent on managing firewalls	<1 hour/week	106 (35.3%)
	1-4 hours/week	117 (39.0%)
	5-8 hours/week	35 (11.7%)
	9-12 hours/week	11 (3.7%)
	13+ hours/week	21 (7.0%)
	Do not directly manage firewalls	10 (3.3%)
Experience as system administrator	<1 year	6 (2.0%)
	1-3 years	46 (15.3%)
	4-6 years	64 (21.3%)
	7-9 years	39 (13.0%)
	10+ years	145 (48.3%)
Proficiency with firewalls	Basic knowledge	20 (6.7%)
	Intermediate	114 (38.0%)
	Advanced	114 (38.0%)
	Expert	52 (17.3%)
Language	English	256 (85.3%)
	Portuguese	7 (2.3%)
	Russian	21 (7.0%)
	Swedish	16 (5.3%)

The Cohen's kappa inter-rater reliability value for the final codes was 0.79, indicating an excellent agreement between the coders [4]. The cases in which the coders varied in the final codes were resolved by the first author, who examined respondents' answers and assigned the most appropriate code.

3.4 Ethical Considerations

The survey was conducted in accordance with the Swedish Ethical Review Act [16] and the Good Research Practice guidelines from the Swedish Research Council [17]. No sensitive personal data were collected and no mental or physical interventions took place. Therefore, no explicit ethical approval was required for this study. The following precautions were taken into consideration to ensure that the participants were treated ethically and with respect:

- The participants provided informed consent before starting the survey. The informed consent form stated the purpose of the study, its approximate duration, our commitment to confidentiality, and their rights as participants,

---

[2] `https://www.reddit.com/r/sysadmin/`

[3] `https://sysadmins.ru/`

[4] `https://www.qsrinternational.com/nvivo/home`

including the right to withdraw from the study at any point in time.

- Only (the minimal) necessary personal data (see Table 1) were collected.
- No sensitive personal data were collected.

4 Results

We describe the survey results by providing both quantitative and qualitative data in Sections 4.1–4.2. In Section 4.3 we report on the suitability of firewall CLIs and GUIs for different tasks.

4.1 Quantitative Data

Seventy percent of the participants in our survey are primarily firewall GUI users, and 60% prefer GUIs to text-based interfaces when having to deal with a firewall (see Table 2). Approximately a quarter of the polled system administrators primarily work with textual interfaces (24% for CLI and 2% for API), and slightly over one third prefer to use these as their main interface: 32% and 4% for CLIs and APIs, respectively. The option `Other` indicates system administrators that use either a combination of the aforementioned interfaces or another type of firewall interface.

Based on our data, there may be a connection between a system administrator's proficiency with firewalls and the interface that they prefer to utilize. Table 3 shows that the stronger the firewall expertise of respondents, the lower the likelihood of utilizing GUIs. Seventy percent of the system administrators with a basic knowledge of firewalls prefer GUIs to any other interface, while this holds true for only 54% of firewall experts.

4.2 Qualitative Data

The thoughts and opinions of the system administrators received during our online survey were coded and grouped according to the following principles: 1) the type of interface: CLI, GUI, API, or other; and 2) the type of comment: positive, negative, or neutral. For the convenience of presenting the strengths and limitations of the interfaces, we categorized the codes as follows:

- We began classifying our codes according to the 10 usability heuristics introduced by Nielsen [12] (see Table 4).
- Because not all codes concerned usability, some of them did not fall into any of the 10 categories, and were further classified according to the ISO/IEC 25010 [5], a standard that defines systems and software quality models (see Figure 1). This includes aspects that are not covered by Nielsen's usability heuristics, such as security and reliability. Regarding usability, the ISO standard comprises appropriateness recognizability, learnability, operability, and accessibility, aspects that are not covered by Nielsen's usability heuristics.
- All remaining codes fell within the `Other` category.

Because the number of respondents who work with APIs or other interfaces is relatively small, we do not report the corresponding results in this paper.

The strengths and limitations of CLIs and GUIs (see Figures 2–5) are discussed in further detail in Sections 4.2.1–4.2.4. In each subsection, we examine the categories that cover 80% of all coding references, starting from the most popular. Note that subsections have different total numbers of coding references, and not all codes in each category are discussed in detail. For convenience, codes are highlighted in bold.

4.2.1 CLI Strengths

According to our respondents, CLIs have a number of strengths (the total number of coding references is 319):

1. Flexibility and efficiency of use (106 coding references; 33.2%). Several respondents (64 coding references) noted the possibility of **automation** as a strength of CLIs: *"CLIs are good targets for automation, even if the only thing you can do is bash scripting."* **User efficiency** was mentioned 42 times. One respondent stated: *"CLIs have a high signal-to-noise ratio, and are therefore preferable to everything else."*
2. Functional suitability (62 coding references; 19.4%). The **superior functionality** of CLIs was mentioned 37 times by system administrators: *"100% coverage of all firewall functionality supported by the OS kernel, unlike GUIs and APIs."* Other useful features of the interface, such as the **ability to work offline** and **ease of search**, were stated 16 times.
3. Usability (30 coding references; 9.4%). According to 12 system administrators, the user has **full control** with a firewall CLI: *"I do not see any reasonable way to be sure a firewall is doing the right thing without using a CLI."* Seven other respondents stated the advantages of managing a firewall with a CLI: *"Properly used, CLI is by far the best method to manage any system."*
4. Performance efficiency (22 coding references; 6.9%). The system administrators noted the superior **speed of operation** of CLIs (22 coding references), commenting *"[CLI] uses zero system resources"* and *"it is faster and does not take five minutes to load."*

Table 2: Relations between primary and preferred firewall interfaces based on the answers from our survey.

		Preferred interface				Total
		CLI	GUI	API	Other	
Primary interface	CLI	61	7	3	2	73 (24.3%)
	GUI	30	169	4	6	209 (69.7%)
	API	0	1	4	0	5 (1.7%)
	Other	4	2	0	7	13 (4.3%)
Total		95 (31.7%)	179 (59.7%)	11 (3.6%)	15 (5.0%)	300 (100.0%)

Table 3: Relations between firewall proficiency and preferred firewall interfaces based on the answers from our survey.

		Preferred interface				Total
		CLI	GUI	API	Other	
Proficiency	Basic knowledge	5	14	0	1	20 (6.7%)
	Intermediate	30	72	2	10	114 (38%)
	Advanced	43	65	5	1	114 (38%)
	Expert	17	28	4	3	52 (17.3%)
Total		95 (31.7%)	179 (59.7%)	11 (3.6%)	15 (5.0%)	300 (100.0%)

5. Visibility of system status (21 coding references; 6.6%). **Transparency** was mentioned 21 times as an important positive characteristic of CLIs: *"With a CLI, you know exactly what the firewall is doing."*

6. Reliability (16 coding references; 5.0%). Our respondents highlighted some strengths of CLIs, such as: **reliability**: *"... there is a lower incidence of random issues with the UI"*; **high availability**: *"I can do the same task via an SSH connection or even a KVM if the whole network is down. I can do that via a smartphone if I must."*; and ease of **configuration backup**: *"Backing up and restoring configurations easily through text files."*

4.2.2 CLI Limitations

The main CLI limitations noted by our respondents are the following (the total number of coding references is 86):

1. Match between system and real world (22 coding references; 25.6%). The main problem, which was referenced 19 times, is a **long learning curve**. System administrators shared that *"CLI may be scary/overwhelming for a beginner/untrained user"* and *"There is typically a slightly higher learning curve associated with CLI, which can often be discouraging to unexperienced users."*

2. Usability (22 coding references; 25.6%). There are two codes that were referenced more than any others: CLIs are **not easy to use** (8 times) and **inconvenient data representation** (7 times). Two respondents stated: *"The CLI is not capable of representing all the firewall rule data in a clean and easy-to-read format"* and *"CLIs are terrible at generating visual information that is comprehensible by non-experts..."* Regarding the ease of use, one system administrator wrote that *"ease of use is a definite issue [of CLI]."*

3. Recognition rather than recall (10 coding references; 11.7%). The facts that CLIs are **less intuitive** and **less educational** were mentioned seven and three times, respectively. CLIs *"may be less intuitive than other interfaces"* and *"you cannot click your way around it in an attempt to* figure it out."

4. Error prevention (8 coding references; 9.3%). CLIs are **prone to errors**, both typographical and logical, and that fact was named 8 times by the respondents. One system administrator wrote that it is *"much easier to cause catastrophic failure quickly and effectively"* with a CLI.

5. Functional suitability (8 coding references; 9.3%). The absence of some auxiliary functionality was noted by eight respondents. A CLI *"has no Ctrl+F [searching] feature."*

4.2.3 GUI Strengths

GUIs have several strengths (the total number of coding references is 586):

1. Usability (236 coding references; 40.3%). In general, GUIs are known to be user-friendly. Visual representations of data provide a **better understanding and/or overview of configuration** according to 124 coding references. One respondent shared with us that *"it [GUI] allows me to have a better understanding of a firewall's configuration while having that information displayed in a more organized manner when compared to a CLI."* The system administrators also stated that GUIs are **easy**

Table 4: Nielsen's usability heuristics [12].

Heuristics	**Short explanation**
Visibility of system status	The system should always keep users informed about what is going on.
Match between system and real world	The system should speak the users' language. Information should appear in a natural and logical order.
User control and freedom	Users need clearly marked emergency exits. The system should support undo and redo.
Consistency and standards	Users should know whether different words, situations, or actions mean the same thing. The system should follow platform conventions.
Error prevention	The system should eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.
Recognition rather than recall	The system should minimize the user's memory load. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
Flexibility and efficiency of use	The system should have accelerators that can speed up interactions for expert users so that it can cater to both inexperienced and experienced users.
Aesthetic and minimalist design	Dialogues should not contain information that is irrelevant or rarely needed.
Help users recognize, diagnose, and recover from errors (we refer to this as assistance with errors)	The system should explain error messages in plain language, precisely indicate the problem, and constructively suggest a solution.
Help and documentation	Any system information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

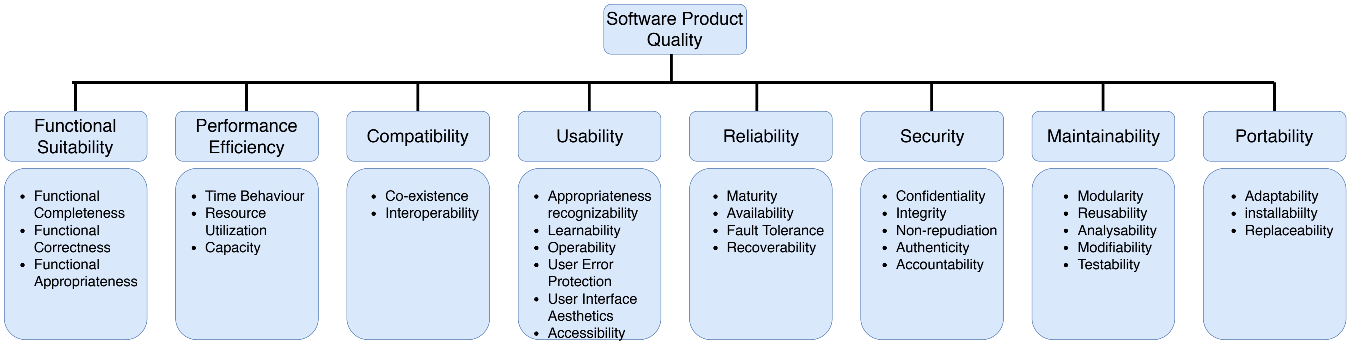

Figure 1: The software quality model ISO/IEC 25010 [5].

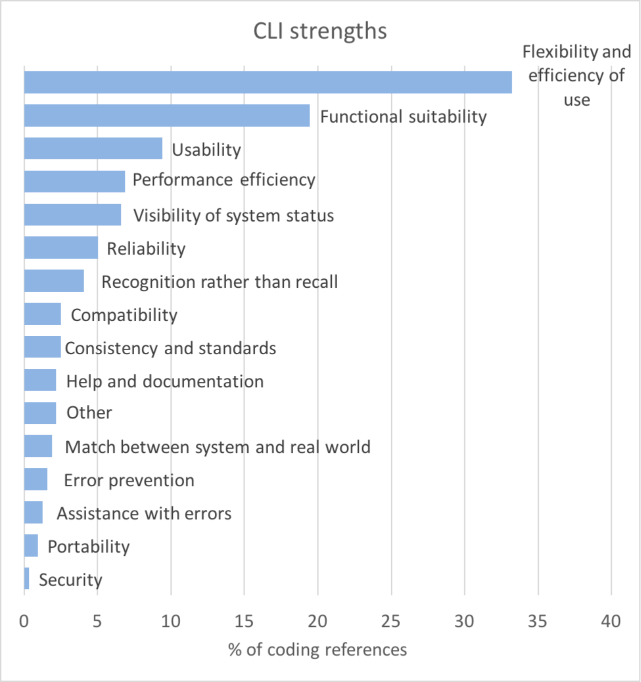

Figure 2: Classification of CLI strengths mentioned by our respondents. The total number of coding references is 319.

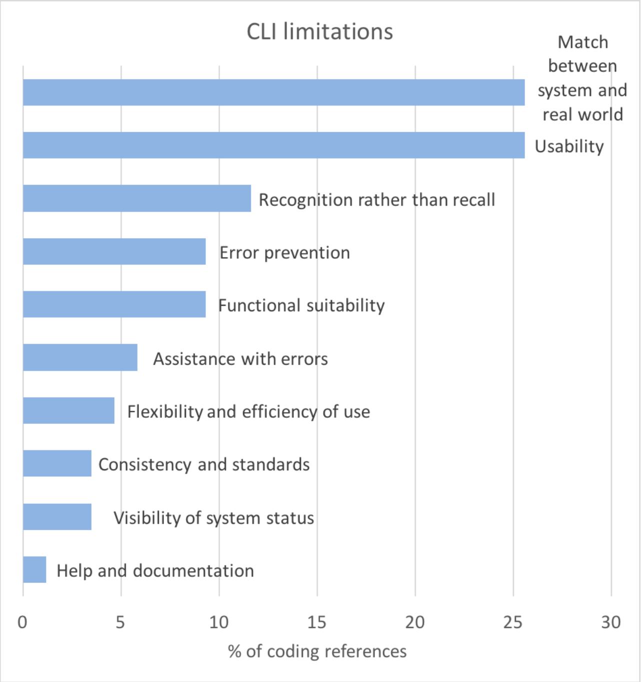

Figure 3: Classification of CLI limitations mentioned by our respondents. The total number of coding references is 86.

to use (49 times), **easy to manage and modify rules** with (19 times), good for **creating rules and policies** (16 times) and **good for people that struggle to work with text** (six times).

2. Functional suitability (120 coding references; 20.5%). The system administrators wrote that GUIs are excellent for a variety of tasks, such as **monitoring** (17 coding references), **reporting** (nine coding references), and **logging** (five coding references). Another strong aspect of GUIs is an **ease of displaying additional information** (20 coding references), such as graphs and statistics.

3. Recognition rather than recall (83 coding references; 14.2%). Being easy to navigate, GUIs are an irreplaceable tool that is **good for occasional use** (44 coding references). A system administrator shared: *"Because of my responsibilities as a general sysadmin [system administrator], management of the firewall takes up only a small part of my time, and having using the GUI for management means that I do not have to remember CLI commands."*

4. Flexibility and efficiency of use (45 coding references; 7.7%). **User efficiency** was named 20 times as a strength of GUIs: *"Makes it faster than using CLI to edit basic things on a firewall ...," "It just gives me a quicker and more visual grasp on what I am doing. Point, click, move on..."*

4.2.4 GUI Limitations

Although the majority of our respondents prefer to use GUIs to other alternatives, several serious limitations of GUIs were named (the total number of coding references is 406):

1. Flexibility and efficiency of use (125 coding references; 30.8%). A **lack of automation** and **user inefficiency** in general when working with the interface were stated 102 times. This makes GUIs less useful for experts. One system administrator wrote: *"We are at a very bad time for GUI firewalls, because experts are the only ones who can effectively scale the workloads demanded of the modern IT infrastructure, and GUIs are almost useless for most experts in that regard."*

2. Functional suitability (56 coding references; 13.8%). According to 56 participants, the **reduced functionality** of GUIs is a serious issue: *"[GUI is] missing a lot of features/settings, so that you have to use CLI to make changes."*

3. Matching between the system and real world (38 coding references; 9.4%). Because a GUIs represents an **additional layer of abstraction**, the user may lack a deeper understanding of their actions (30 coding references). A system administrator formulated a drawback of GUIs as *"a lack of knowledge for the underlying system you are working on."* Another problem, named six times, is that GUIs may **generate less understandable configu-**

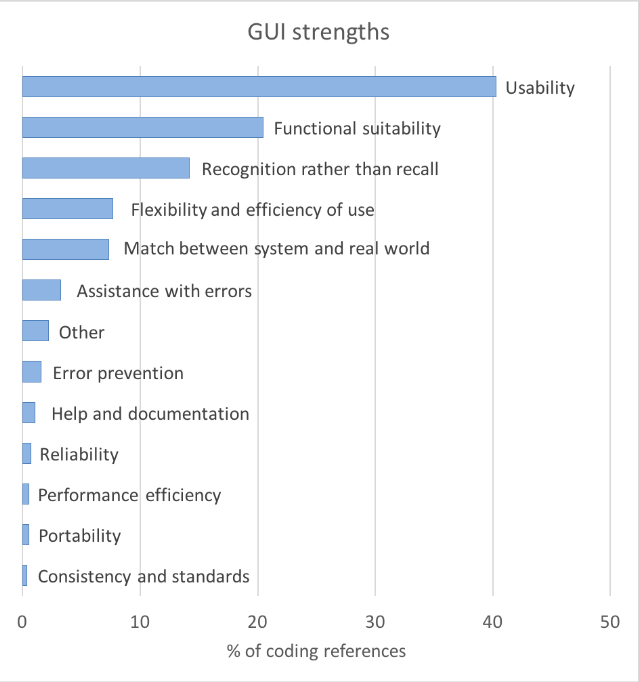

Figure 4: Classification of GUI strengths mentioned by our respondents. Total number of coding references is 586.

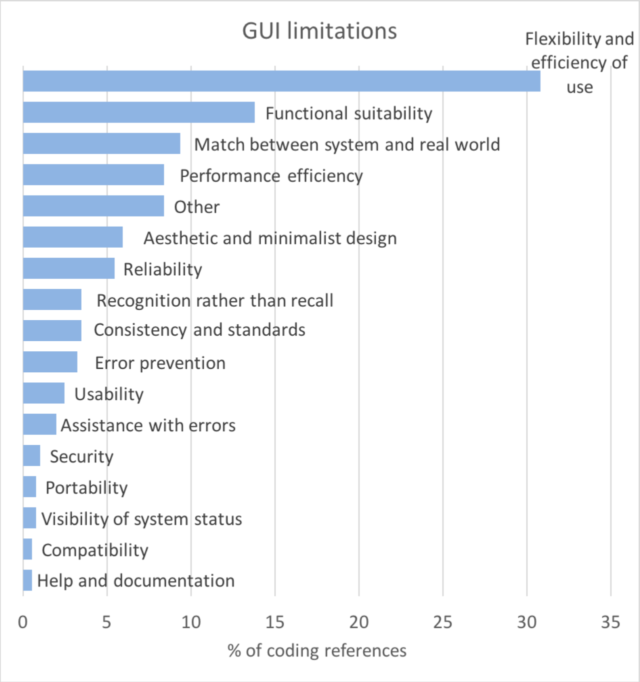

Figure 5: Classification of GUI limitations mentioned by our respondents. Total number of coding references is 406.

ration files: *“GUIs do not always generate configs that make logical/visual sense to a human.”*

4. Performance efficiency (34 coding references; 8.4%). GUIs are highly demanding in terms of system resources, and for this reason are usually very **slow** (34 references): *“GUIs take more overhead to display and run, which may draw away from a firewall’s processing power.”*

5. Other (34 coding references; 8.4%). The system administrators stated a number of problems. The facts that GUIs **require additional equipment or software** and are **platform or browser dependent** were mentioned 12 times each. Two participants shared that *“A software client can be needed, which may not always be accessible...”* and *“Depending on browser it can be a horrible experience (slow, unresponsive, thus can cause issues with clicks being registered late or not at all).”* Several additional issues were mentioned by the system administrators, such as GUIs being **difficult to document** (five references): *“Unlike CLIs, documenting a GUI is mostly useless and defeats most of the purpose of a GUI,”* and **unavailable for particular firewalls** (three references): *“I currently do not have a firewall that supports GUI . . . .”*

6. Aesthetic and minimalist design (24 coding references; 5.9%). The respondents encountered **badly designed** GUIs (16 references): *“. . . some [GUIs] are horribly designed so it is hard to figure out how to do what you want to do.”* Eight system administrators noted the problem of an **interface beauty and functionality trade-off**: *“Most [G]UIs are either poorly laid out making them difficult to use or are too user-friendly and do not have all settings available.”*

7. Reliability (22 coding references; 5.4%). The respondents mentioned reliability issues (22 times) with GUIs: *“They can crash and become unresponsive, sometimes you cannot tell if it is processing a new config/update or if it is locked up.”*

4.3 Suitability for Different Tasks

In addition to the strengths and limitations of firewall CLIs and GUIs, the system administrators informed us of which interface they deem to be the most suitable for each task.

All use cases associated with entering many similar rules on one computer or bulk changes on several computers at once require the use of a CLI. Furthermore, non-standard tasks in which, for example, rules with a set of advanced options are necessary, are more easily solved using a CLI according to the respondents. One system administrator stated:

> *“GUIs and APIs are terrible at handling special cases and rarely expose the underlying command structure properly (they always create a monopoly on how things are done). With CLIs, it is usually a straightforward process*

> *to use the underlying commands and kernel modules directly (iptables and netfilter on Linux).*"

Firewall GUIs are preferable in more tasks according to the polled system administrators. The most frequently mentioned use cases in which the use of a GUI is beneficial are the creation of individual rules and building entire configurations. One respondent commented:

> "*The more complicated a task is, the more important a GUI becomes. Who wants to set complex web proxy configuration options via CLI? Let us say that I want to proxy students, teachers, and administrators in a school differently. Let us start with just http request options for request headers, allowed auth[entication] methods, DLP scanning for HTTP POST, etc. Imagine the difference between clicking checkboxes and dropdown menus vs trying to type all this out via a CLI with some reference manual . . .* "

Another task that is easier to perform in a GUI is viewing and inspecting firewall rules and policies. GUIs usually have an option to link objects with rules, which allows the bigger picture to be observed.

> "*Examining and working with firewall rules is an instructive example of where the CLI is not a good option. The CLI is not capable of representing all the firewall rule data in a clean and easy-to-read format. This is much easier on a GUI.*"

Monitoring is another use case in which the system administrators decide to use GUIs. GUIs provide the ability to view connection statistics, monitor traffic flows through real-time graphs, and so on, and are therefore preferred. The system administrators also tend to choose GUIs when there is a need to change the order of rules in a rule set.

5 Discussion

The collected quantitative data provides insights into the usage of different firewall interfaces that are considerably different from what has been previously published in the literature. We observe a significant shift towards the use of GUIs, although CLIs have been widely utilized by system administrators in the recent past [18]. There are three possible explanations for this shift.

The first concerns the case of security tools, in particular firewalls, where designers attempt to follow the design principles formulated for system administration tools. Our participants confirmed that firewall GUI implementations are improving. One system administrator opined:

> "*Decades of GUI development: a 2D mouse and keyboard with a keyboard shortcuts interface instead of the serial text in/out of a classic CLI. More available and powerful searching, sorting, and filtering of information; discoverability of available commands; and visual/graphical possibilities of a large and high-res screen.*"

The second possible reason is that the number of system administrators has significantly increased, including those with limited technical expertise, as described by Xu et al. [22]. The statement on less experienced system administrators is not valid for our data sample, as 83% of the respondents have worked for over three years as system administrators (see Table 1).

The third possible reason is that there are many system administrators who are not security experts, but rather general purpose system administrators. As we can also observe from Table 1, 74% of the respondents spend no more than four hours per week managing firewalls. Therefore, they are most likely general purpose system administrators, and this explains their reluctance to work with firewall CLIs, which are less usable, require more learning time, and are prone to errors according to our participants.

Our qualitative data show that GUIs are less preferable for experts compared to system administrators with a lower firewall proficiency. The respondents noted that GUIs are not very useful for experts, as they severely restrict the user with limited functionality, a low operation speed, and low user interaction efficiency owing to the lack of automation capabilities.

Another feature that we noticed when analyzing the data from the survey is that our respondents' preferences for one interface do not always depend on their strengths or limitations. Sometimes system administrators are more comfortable with a CLI or GUI simply because they familiarized themselves with this interface first. One respondent stated:

> "*I am old school and there were no GUIs back in the day, so it [CLI] is more comfortable for me.*"

Another possible reason is that system administrators do not always have experience with other interfaces, and therefore cannot objectively compare their strengths and limitations.

5.1 Limitations

One of the limitations of our study is that most of the respondents were recruited through online forums for system administrators. Because the survey participants were volunteers, there is a self-selection bias that leads to the sample not being fully representative.

Furthermore, the study was conducted online and we could not observe the participants answering the questions. Moreover, some of the answers were ambiguous, and so we had to interpret them, which could lead to a distortion of the meaning that the respondent had originally intended. For example, the comment *"slow"* can refer both to the speed of operation of the software and the speed of interaction between the user and interface. There is also a possibility of questions being misunderstood or misinterpreted by the participants. Additionally, self-report surveys have several common limitations [7], such as social desirability biases and acquiescent responses.

We mitigated the limitations by carefully considering the design of the survey, pretesting it with several participants, making it anonymous so that people could answer honestly, and shortening it to minimize respondent fatigue.

5.2 Design Recommendations

Our survey identified some problems for both CLIs and GUIs that should be taken into account. In this section, we present some design recommendations for CLIs and GUIs based on the results of our survey, as well as discussing the benefits of combining these two interfaces into one.

As one respondent noted:

> "*CLI interfaces are not usually as forgiving as other interfaces. If you are not paying attention, then the slightest typo could cause large issues.*"

Our recommendation is to employ a syntactical verification of commands when a user types in an instruction to prevent errors in firewall configuration processes.

Furthermore, because CLIs have a reasonably long learning curve, assistance in writing rules is necessary for less experienced system administrators. Respondents noted the following:

> "*It may be difficult to compose rules [in CLIs] without an example.*"

Providing a knowledge base of examples of rules could be a useful approach.

We make three recommendations regarding GUIs. First, the system administrators complained about the speed of operation of GUIs. Our recommendation is to not make GUIs bloated, so that they do not consume a lot of system resources and can be run on mediocre hardware.

The second recommendation relates to the GUI installation process. As one of the system administrators commented:

> "*They [GUIs] are not really for beginners because of the initial setup required to configure them.*"

Because the highest percentage of GUI use is among system administrators with the least firewall expertise, installing a firewall should not be a complicated task.

In addition, to increase the speed of user interaction with a firewall GUI it is necessary to allow system administrators to create their own combinations of hotkeys for the most popular actions. This will help to make GUIs more attractive for firewall experts.

While we have provided recommendations for how to improve each interface, there remain problems that are difficult to solve within one interface. For example, textual interfaces are inherently inadequate for presenting a large amount of information:

> "*When a config file has over 2000+ lines it is easy to lose track of what is what [in CLI].*"

Another limitation originates from the concept of a CLI: it is impossible to create and edit rules using the mouse cursor and check boxes, which in some cases can significantly increase the productivity of a firewall operator. For GUIs, the problem is the lack of automation tools, as was noted by a large number of respondents.

A more effective solution would be to combine two interfaces into one, with the ability to seamlessly switch from one to the other, so that interacting with one interface affects the other. Such an approach can leverage the strengths of each interface while mitigating their limitations [11]. A GUI can provide an overview of configurations and display additional graphs and statistics, as well as being used to create rules, while a CLI can offer on-demand access to the powerful automation capabilities. Such a combined interface could be suitable for users with different firewall expertise. Less experienced system administrators could be trained to use the CLI by viewing the underlying text-based commands while working in the GUI. Expert users could continue using commands to create rules, while using the GUI for a better policy overview. We strongly believe that such a firewall interface would be widely accepted in the system administration community.

6 Conclusion

In this work, we present an online study concerning system administrators, in which we examine how they interact with different firewall interfaces. The survey results show that 70% of the polled system administrators are primarily GUI users, and 60% prefer this interface for interacting with a firewall. This finding differs from previously published findings in the literature, in which CLIs were claimed to be the first choice of system administrators.

We classify the strengths and limitations of firewall CLIs and GUIs. Our participants report that CLIs are flexible, efficient, transparent, reliable, and achieve ultimate functionality and a good performance. However, they are inconvenient for representing data, do not help users by preventing errors, and have a long learning curve. On the other hand, GUIs help users to perceive firewall configuration information more effectively and have a shorter learning curve compared to CLIs. They are also easy to use, easy to create and modify rules with, and good for occasional use. Regarding the limitations of GUIs, they restrict users with limited functionality, a low operation speed, and a low user interaction efficiency. They are neither transparent nor reliable. In addition, we report the preferred interface for each task according to the system administrators.

Our findings present opportunities for future research. A well-designed firewall interface should predict and interpret its user's needs and assist them in becoming proficient with the firewall. In this case, the system administrator is satisfied with the firewall and can efficiently perform the required work. On the other hand, a poorly designed firewall interface might hinder the successful execution of tasks and lead to the future disuse of that solution. We provide some design recommendations that should be taken into account by designers aiming to develop better CLIs and GUIs.

Acknowledgments

We are grateful to all the system administrators that participated in our study. We would also like to thank the moderators of the *Sysadmin* subreddit (`r/sysadmin`) for allowing us to reach out to their community.

This work was supported by the Knowledge Foundation of Sweden HITS project and by the Swedish Foundation for Strategic Research SURPRISE project.

Availability

The final codes and other details are available at https://github.com/soups2019-126/supplementary_material.

References

[1] Catherine Ashcraft, Brad McLain, and Elizabeth Eger. *Women in tech: The facts*. National Center for Women & Technology (NCWIT), 2016.

[2] David Botta, Rodrigo Werlinger, André Gagné, Konstantin Beznosov, Lee Iverson, Sidney Fels, and Brian D. Fisher. Towards understanding IT security professionals and their tools. In *Symp. on Usable Privacy and Security (SOUPS)*. ACM, 2007.

[3] Blake Burkhart. Subreddit gender ratios. http://bburky.com/subredditgenderratios/, 2017.

[4] Joseph L Fleiss, Bruce Levin, and Myunghee Cho Paik. *Statistical methods for rates and proportions*. John Wiley & Sons, 2003.

[5] International Organization for Standardization. *ISO-IEC 25010: 2011 Systems and Software Engineering-Systems and Software Quality Requirements and Evaluation (SQuaRE)-System and Software Quality Models*. ISO, 2011.

[6] André Gagné, Kasia Muldner, and Konstantin Beznosov. Identifying differences between security and other it professionals: a qualitative analysis. *HAISA*, 8:69–80, 2008.

[7] Robert M Gonyea. Self-reported data in institutional research: Review and recommendations. *New directions for institutional research*, 2005(127):73–89, 2005.

[8] Carol Grbich. *Qualitative data analysis: An introduction*. Sage, 2012.

[9] Eben M Haber and John Bailey. Design guidelines for system administration tools developed through ethnographic field studies. In *Proceedings of the 2007 symposium on Computer human interaction for the management of information technology*, page 1. ACM, 2007.

[10] Jeevitha Mahendiran, Kirstie A Hawkey, and Nur Zincir-Heywood. Exploring the need for visualizations in system administration tools. In *CHI'14 Extended Abstracts on Human Factors in Computing Systems*, pages 1429–1434. ACM, 2014.

[11] Sandra R Murillo and J Alfredo Sánchez. Empowering interfaces for system administrators: Keeping the command line in mind when designing GUIs. In *Proceedings of the XV International Conference on Human Computer Interaction*, page 47. ACM, 2014.

[12] Jakob Nielsen and Robert L Mack, editors. *Usability Inspection Methods*. John Wiley & Sons, Inc., New York, NY, USA, 1994.

[13] Johnny Saldaña. *The Coding Manual for Qualitative Researchers*. SAGE Publications, 2012.

[14] William Sattelberg. The demographics of reddit: Who uses the site? https://www.techjunkie.com/demographics-reddit/, 2018.

[15] Hadi Shiravi, Ali Shiravi, and Ali A Ghorbani. A survey of visualization systems for network security. *IEEE Transactions on visualization and computer graphics*, 18(8):1313–1329, 2012.

[16] Svensk Författningssamling (SFS). *Lag (2003:460) om etikprövning av forskning som avser människor [The Act concerning the Ethical Review of Research Involving Humans]*. Utbildningsdepartementet, Stockholm, Sweden, 2003.

[17] Swedish Research Council (VR). Conducting ethical research. https://www.vr.se/utlysningar-och-beslut/villkor-for-bidrag/att-forska-etiskt.html, 2018. Accessed: 2019-02-26.

[18] Leila Takayama and Eser Kandogan. Trust as an underlying factor of system administrator interface choice. In *CHI'06 extended abstracts on Human factors in computing systems*, pages 1391–1396. ACM, 2006.

[19] Ramona Su Thompson, Esa M Rantanen, William Yurcik, and Brian P Bailey. Command line or pretty lines?: comparing textual and visual interfaces for intrusion detection. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, page 1205. ACM, 2007.

[20] Nicole F Velasquez, Suzanne P Weisband, and Alexandra Durcikova. Designing tools for system administrators: An empirical test of the integrated user satisfaction model. In *LISA*, pages 1–8, 2008.

[21] Artem Voronkov, Leonardo Horn Iwaya, Leonardo A Martucci, and Stefan Lindskog. Systematic literature review on usability of firewall configuration. *ACM Computing Surveys (CSUR)*, 50(6):87, 2018.

[22] Tianyin Xu, Vineet Pandey, and Scott Klemmer. An HCI view of configuration problems. *arXiv preprint arXiv:1601.01747*, 2016.

Appendix

A Survey Questions

Page 1

1. How much time per week (on average) do you spend directly interacting with (managing) firewalls?
    - ○ Less than 1 hour/week
    - ○ 1–4 hours/week
    - ○ 5–8 hours/week
    - ○ 9–12 hours/week
    - ○ More than 12 hours/week
    - ○ I do not directly manage firewalls
2. Can you enumerate all the firewall-related tasks that you have dealt with?
    - ○ Adding/removing firewall rules
    - ○ Examining firewall policies to understand their functionalities
    - ○ Inspecting firewall rules/policies to find errors or inconsistencies
    - ○ Other (please specify)
3. What is the PRIMARY firewall interface that you use at work?[5]
    - ○ Command Line Interface (CLI)
    - ○ Graphical User Interface (GUI)
    - ○ Application Programming Interface (API)
    - ○ Other (please specify)
4. What is your PREFERRED firewall interface?[6]
    - ○ Command Line Interface (CLI)
    - ○ Graphical User Interface (GUI)
    - ○ Application Programming Interface (API)
    - ○ Other (please specify)

```
if answer(Q3) = answer(Q4) then
    go to Page 2
else if answer(Q3) = 2 then
    go to Page 3
else if answer(Q4) = 2 then
    go to Page 4
else
    go to Page 5
```

Page 2

5. Are there certain tasks that the *%preferred%* allows you to do, which are more difficult to do using other firewall interfaces?
6. What are the strengths of the *%preferred%*, if any?
7. Can you think of any problems associated with the *%preferred%*?

```
if answer(Q3) = 2 then
    go to Page 10
else
    go to Page 5
```

Page 3

8. Why do you prefer the *%preferred%* to the *%primary%* when managing firewalls?
9. What are the strengths of the *%preferred%*, if any?
10. Do you see any strengths in the *%primary%*?
11. What problems do you see with the *%primary%*?

```
go to Page 10
```

---

[5] *%primary%* returns the selected option in Question 3

[6] *%preferred%* returns the selected option in Question 4

Page 4

12. Why do you prefer the *%preferred%* to the *%primary%* when managing firewalls?
13. What are the strengths of the *%preferred%*, if any?
14. Can you think of any problems associated with the *%preferred%*?
15. Do you see any strengths in the *%primary%*?

```
go to Page 10
```

Page 5

16. Have you ever used a graphical user interface (GUI) to manage a firewall?
    - ○ Yes
    - ○ No

```
if answer(Q16) = 2 then
    go to Page 6
else
    go to Page 7
```

Page 6

17. Can you name the reasons for not trying a firewall graphical user interface (GUI)?

```
go to Page 10
```

Page 7

18. Are you currently using a GUI to manage your firewall?
    - ○ Yes
    - ○ No

```
if answer(Q18) = 2 then
    go to Page 8
else
    go to Page 9
```

Page 8

19. Can you name the reasons for not using a firewall with a GUI and whether you see problems with GUIs?

```
go to Page 10
```

Page 9

20. For which tasks do you use the firewall graphical user interface (GUI)?

Page 10

21. How long have you been working as a system/network administrator?
    - ○ Less than a year
    - ○ 1–3 years
    - ○ 4–6 years
    - ○ 7–9 years
    - ○ 10 years and more
22. How would you describe your proficiency with firewalls?
    - ○ Basic knowledge
    - ○ Intermediate
    - ○ Advanced
    - ○ Expert
23. How old are you?
    - ○ 18–24 years old
    - ○ 25–34 years old
    - ○ 35–44 years old
    - ○ 45–54 years old
    - ○ 55–64 years old
    - ○ 65 years or older
    - ○ Prefer not to answer
24. What is your gender?
    - ○ Female
    - ○ Male
    - ○ Other
    - ○ Prefer not to answer