Systematic Literature Review on Usability of Firewall Configuration

ARTEM VORONKOV, LEONARDO HORN IWAYA, LEONARDO A. MARTUCCI, and STEFAN LINDSKOG, Karlstad University

Firewalls are network security components that handle incoming and outgoing network traffic based on a set of rules. The process of correctly configuring a firewall is complicated and prone to error, and it worsens as the network complexity grows. A poorly configured firewall may result in major security threats; in the case of a network firewall, an organization's security could be endangered, and in the case of a personal firewall, an individual computer's security is threatened. A major reason for poorly configured firewalls, as pointed out in the literature, is usability issues. Our aim is to identify existing solutions that help professional and non-professional users to create and manage firewall configuration files, and to analyze the proposals in respect of usability. A systematic literature review with a focus on the usability of firewall configuration is presented in the article. Its main goal is to explore what has already been done in this field. In the primary selection procedure, 1,202 articles were retrieved and then screened. The secondary selection led us to 35 articles carefully chosen for further investigation, of which 14 articles were selected and summarized. As main contributions, we propose a taxonomy of existing solutions as well as a synthesis and in-depth discussion about the state of the art in firewall usability. Among the main findings, we perceived that there is a lack (or even an absence) of usability evaluation or user studies to validate the proposed models. Although all articles are related to the topic of usability, none of them clearly defines it, and only a few actually employ usability design principles and/or guidelines.

CCS Concepts: • **General and reference → Surveys and overviews**; • **Security and privacy → Usability in security and privacy**; *Firewalls*; • **Networks →** *Firewalls*; • **Human-centered computing →** *Information visualization*;

Additional Key Words and Phrases: Firewall, usability, visualization, systematic literature review

ACM Reference format:
Artem Voronkov, Leonardo Horn Iwaya, Leonardo A. Martucci, and Stefan Lindskog. 2017. Systematic Literature Review on Usability of Firewall Configuration. *ACM Comput. Surv.* 50, 6, Article 87 (December 2017), 35 pages.
https://doi.org/10.1145/3130876

1 INTRODUCTION

A firewall is a system consisting of software and/or hardware that is designed to prevent unauthorized access to/from a network or device. The term *firewall* was borrowed from the field of fire prevention, where a firewall is a barrier intended to confine a fire within a building [29]. Until

The work is being carried out in the High Quality Networked Services in a Mobile World (HITS) project, partly funded by the Knowledge Foundation of Sweden.

Authors' addresses: A. Voronkov, L. H. Iwaya, L. A. Martucci, and S. Lindskog; emails: {artem.voronkov, leonardo.iwaya, leonardo.martucci, stefan.lindskog}@kau.se.

https://doi.org/10.1145/3130876

the late 1980s, the only means of network security were routers with Access Control Lists (ACLs) that determined whether network access should be allowed or denied to specific IP addresses [29]. However, the enormous increase in the size of the Internet meant that this type of filtering was no longer sufficient to keep out malicious traffic. It is hard to track the history of firewalls and who, actually, invented the technology. One of the first articles about firewalls was published in 1987 by Digital Equipment Corporation (DEC) [43]. (Ranum 1992) discusses application layer firewalls and reports that DEC firewalls had been operating for over six years, that is, since 1986. Another article [13] presents a secure internet gateway, produced by AT&T Bell Laboratories, which also operates at the application layer.

Firewalls are employed to regulate the network traffic between computers or other devices and networks. They accept or drop packets according to a given policy [14]. Firewalls can be divided into two classes [47]:

(1) *Personal firewalls* (PF) aim at protecting a single host from unauthorized access.
(2) *Network firewalls* (NF) aim at protecting the resources of a whole network.

Firewalls can also be classified according to the OSI layer at which they operate: (1) *packet filters* operate mostly at data link, network and transport layers; (2) *circuit gateways* (also known as *stateful inspection firewalls*) operate at data link, network, transport and session layers; and (3) *application gateways* operate in the full range from the data link to the application layer [14]. For further details about different kinds of firewalls, we refer the reader to References [14, 29].

Network administrators normally use a phased approach for firewall planning and implementation. For instance, the NIST guideline on firewalls and firewall policy [57] has five phases that are repeated whenever the security policy changes significantly:

(1) *Plan*, that is, consider the organization's security policy and determine which type of firewall should be implemented.
(2) *Configuration*, where firewalls' software and/or hardware are installed and configured together with setting up rules.
(3) *Implement* and *test* a prototype of the design solution and thus evaluate functionality, performance, scalability and security.
(4) Initiate *deployment* of the firewall into the enterprise.
(5) *Management* of the infrastructure, component maintenance and support for operational issues.

Among the aforementioned phases, we focus on the process of *correctly* configuring a firewall, that is, setting up the rule set that implements the organization's/end user's security policy. We are aware that end users do not usually use all the steps of this approach when work with their PFs, however, the configuration phase is probably always present. According to (Rubin et al. 1997), configuration is "the most important factor of firewall's security." (Wong 2008) compared this task to programming a distributed system in assembly language, due to the fact that *configuration languages* are low-level and vendor-specific. Furthermore, administrators need to *separately* configure multiple firewalls (perhaps from different vendors) in the network. Changes in one firewall might affect others (i.e., rule interactions are complex); and rule sets can be large, containing hundreds or even thousands of rules [20, 66].

A quantitative study of firewall configurations revealed that all investigated rule sets ($n = 12$) had errors [67]. This finding was afterwards corroborated by examination of a larger number of rule sets ($n = 36$) [68]. The latter study showed that "firewalls are still poorly configured, a rule set's complexity is positively correlated with the number of errors" and there is "no significant indication that later software versions have fewer errors." Such errors in rule sets are also referred

Table 1. Usability Aspects as Described in References [30, 45, 58] (Adapted from Reference [60])

ISO 9241-11	**Nielsen**	**Shneiderman & Plaisant**
Efficiency	Efficiency	Speed of performance
	Learnability	Time to learn
Effectiveness	Memorability	Retention over time
	Errors/Safety	Rate of errors by users
Satisfaction	Satisfaction	Subjective satisfaction

to as anomalies (misconfigurations) [2, 3, 28]. Automatic anomaly detection and correction is an entire consolidated field devoted to helping solve configuration problems.

However, nearly all the articles on anomaly detection tools lack user studies to validate applicability of proposed models. It is equally important to find effective ways to convey the outputted information about firewall misconfigurations to the user. In this respect, we noticed that very few works dealt with the *usability* issues regarding firewalls.

Usability, in turn, is defined by ISO 9241-11 [30] as: “The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.” Therefore, according to (Bevan et al. 2015), how successfully goals were achieved indicates effectiveness, how properly time was utilized indicates efficiency, and how willing a user is to utilize a system indicates satisfaction. Still, more aspects of usability can be considered. For instance, in addition to efficiency and satisfaction, (Nielsen 1994) also includes: learnability in early use; memorability after a period of nonuse; and, that errors during use should be corrigible, and should not lead to undesirable consequences. Likewise, (Shneiderman and Plaisant 2005) also introduce five *usability measures*: time to learn; speed of performance; rate of errors by users; retention over time; and subjective satisfaction. Such *usability aspects* defined in References [30, 45, 58] are all closely connected. Table 1 provides a summary of these definitions, as presented in Reference [60].

These usability definitions come from the research area of Human-Computer Interaction (HCI). However, security researchers also expressed concern regarding usability long before the ISO 9241-11 existed. In 1975, (Saltzer and Schroeder 1975) introduced *psychological acceptability* as one of the eight design principles of information protection systems. Usability and security thus started to align to areas currently known as HCI-SEC or usable security. Initial work on usable security focused mostly on end users and the conception of design principles, for example, References [64, 69]. Notwithstanding, usable security for advanced users (e.g., system administrators, security experts and hackers) and its design principles have also been studied, as in References [9, 15, 31]. Visualization techniques became one of the most widely used approaches for the usable security [18, 36, 53], because “given the huge amount of data needed to analyze security problems, visualization seems to be the right approach …” [42]. It is worth mentioning that visualization was particularly used in the management of security policies [52, 61], where the authors of the latter work argue in favor of it: “one of the main problems with today’s policy-authoring user interfaces is that the dominant model they use for displaying policies—the list-of-rules model—is deficient.” Therefore, visualization techniques can be considered inherently relevant to the efficiency aspect of usability.

In summary, usability is one of the most important properties of security mechanisms. As mentioned by (Sasse and Smith 2016): “security mechanisms are often too time consuming for people to bother with, or so complex that even those willing to use them make mistakes.” In other words,

Table 2. The Selected Databases used for the SLR Search Stage

Database	Type	URL
ACM Digital Library	Digital Library	http://dl.acm.org/
IEEE Xplore	Digital Library	http://ieeexplore.ieee.org/Xplore/home.jsp
dblp	Bibliographic Database	http://dblp.uni-trier.de/
Inspec	Bibliographic Database	http://www.theiet.org/resources/inspec/

not being able to perform required tasks for a reasonable amount of time with a reasonable amount of effort will most likely lead to the refusal of use of security mechanisms for end users or to the presence of mistakes for professionals. This is of course unacceptable for maintaining an organization's or individual computer's security.

Given that, to understand the state of the art in usability of firewall configuration, we conducted a Systematic Literature Review (SLR). For this purpose, four different full-text and bibliographic databases, shown in Table 2, were used. Other databases were considered, but were not taken into account, because they return a significant number of irrelevant results for the following reasons: (1) the impossibility of searching for peer-reviewed articles only, for example, Google Scholar[1]; or (2) the impossibility of using logical expressions for searching through titles/keywords and/or abstracts, that is, search through the full-text only, for example, Springer Link.[2]

This SLR has identified 14 articles addressing the usability of firewall configuration. Among the selected articles, nine are dedicated to network firewalls and five to personal firewalls. As a result, this SLR contributes a taxonomy of the selected articles and a synthesis of the work done so far on the usability of firewall configuration.

The remainder of the article is organized as follows. Section 2 presents the SLR protocol, a document that describes the whole review process, working as an agreement and planning tool for researchers. Sections 3 and 4 explain and document the conducting of the primary and secondary selection procedures. Section 5 presents a taxonomy and a summary of the 14 selected articles. The synthesis of the SLR and discussion of the main findings are presented in Section 6. Concluding remarks are given in Section 7.

2 SYSTEMATIC LITERATURE REVIEW METHODOLOGY

The SLR method is briefly described in the following text, but we refer the reader to the original article [35] for greater detail. SLR is a well-planned review for the purpose of answering specific Research Questions (RQs). It helps with identification, selection, and critical evaluation of the results of included studies. The SLR method comprises five main stages:

(1) Survey protocol
(2) Primary selection
(3) Secondary selection
(4) Meta analysis (if needed)
(5) Synthesis

The survey protocol is described in the next subsections. All other aforementioned stages can be found in Sections 3–6.

[1] https://scholar.google.com/.
[2] https://link.springer.com/.

2.1 Research Questions

The most crucial step of the SLR protocol is to formulate right Research Questions (RQs). In this SLR, the following RQs were asked:

> **RQ1:** What are the existing solutions/approaches that address usability aspects in the configuration of firewalls?
> **RQ2:** How are these solutions/approaches evaluated and/or validated?

In what follows, we detailed our SLR strategy that addresses the aforementioned RQs.

2.2 Search Strategy

The first search of an SLR is called the *primary selection* and consists of a retrieval of articles from databases. The databases used in this SLR are listed in Table 2. Each database may return hundreds of articles. In the case of a large amount of data in the secondary selection, a scanning method may be adopted to quickly evaluate the relevance of articles by reading their titles and abstracts. An article's title and abstract should indicate conformity to the RQs for it to be selected for further analysis. At this point, the article's content is reviewed in its entirety, and the core information is extracted following a pre-defined strategy.

It is also important to make an account, whenever feasible, of the number of articles retrieved from the databases, how many were screened (if not all) and how many were chosen in the secondary selection. In this way, it is possible to scrutinize the inclusion/exclusion of articles (during the primary and secondary selection procedures) and to document the narrowing process that SLR entails. The SLR synthesis should then be done on the extracted data of the remaining relevant articles.

As previously mentioned, the search sources for this SLR were narrowed down to the four databases in Table 2. The Association for Computing Machinery (ACM) *Digital Library* contains the full-text collection of all articles published by the ACM, comprising a body of knowledge of more than one million entries. The Institute of Electrical and Electronics Engineers (IEEE) online digital library is called *IEEE Xplore*. The most frequently cited publications in electrical engineering, electronics and computer science can be accessed as part of a vast collection of more than 3.5 million items. The ACM Digital Library and IEEE Xplore together form a fundamental core of today's scientific research on electrical engineering and computing. To broaden the SLR's scope, it was decided to include two additional relevant bibliographical databases: dblp and Inspec. The *dblp computer science bibliography* is an online reference for bibliographic information on major computer science publications that indexes over three million publications, 25,000 journal volumes and more than 24,000 conferences or workshops. Finally, scientific literature related to the fields of physics, computing, control and engineering is provided by an indexing database called *Inspec*. It is worth noting that there is an intersection between information sources; thus, some articles can appear in more than one database.

Inclusion criteria must be decided after completing the selection of information sources. The articles that meet these criteria will compose a set of articles relevant for answering our RQs.

- The publication date is within the period from 2005 to 2016. The reason for this is because we want to focus on recent publications. Besides that, 2005 is when this research area started to become popular. However, in the secondary selection, if a article cites another potential/competing solution that was published earlier, we should include this reference for further investigation (even if we could not find it in the databases). If a study had been published several times, then we kept the latest (the most complete) publication.

Table 3. Three Groups of Key Terms used for the First Search String

	Group 1	Group 2	Group 3
Term 1	firewall*	rul*	config*
Term 2		polic*	anomal*
Term 3			*consisten*
Term 4			conflict*
Term 5			error*

Table 4. Two Groups of Key Terms used for the Second Search String

	Group 1	Group 4
Term 1	firewall*	visual*
Term 2		graphic*
Term 3		GUI
Term 4		usab*
Term 5		user* AND stud*

- A article must have gone through a peer review process. It is worth noting that non-scientific publications, for example white articles and any kind of gray literature, are not considered.
- A article must discuss questions related to the usability of the firewall configuration process. It can either be a user study (user evaluation) of interaction with firewalls or a solution that reduces the effort needed to understand, create or manage rule sets applying visualization techniques.

2.3 Search Terms

To answer our RQs, we conducted multiple queries to the different databases with well-specified *search terms*. For this step, we adopted the search strategy introduced in the work of (Lillegraven and Wolden 2010).

We aimed to create two search strings with a focus on: (1) Firewall conflict detection and/or resolution; and (2) Firewall usability (hereinafter, we use the terms "firewall usability", "usability of firewalls" and "usability of firewall configuration" as synonyms). For this, we formed four (the number of groups need to be reduced to four) *groups of one or more key terms*, which are shown in Tables 3 and 4. The asterisk ($*$) matches any character zero or more times. Each group consists of search terms that are either synonyms or have related semantic meanings in our study. Group 1 targets articles that focus on firewalls; Group 2 targets articles about rule sets; Group 3 targets articles on configuration mistakes; Group 4 targets papers that deal with visualization and usability.

The studies we aimed to find through the search were the ones in the intersection of Group 1, Group 2 and Group 3, as well as Group 1 and Group 4. For this purpose, we used the logical OR operator within the groups to find studies that are related to any of the terms in each group. The groups themselves were combined with the logical AND operator. The two following search strings were derived:

(1) ([G1,T1]) AND ([G2,T1] OR [G2,T2]) AND ([G3,T1] OR [G3,T2] OR [G3,T3] OR [G3,T4] OR [G3,T5])

(2) ([G1,T1]) AND ([G4,T1] OR [G4,T2] OR [G4,T3] OR [G4,T4] OR [G4,T5])

As previously mentioned, the *first search string* aims to find all articles on conflict detection and/or resolution in rule sets. In spite of the fact that the focus of such articles is mainly security, some of them address two or more usability aspects and, thus, must be included in our SLR. Group 2 plays an auxiliary role and is here used for limiting the number of results returned by Group 1. The *second search string* retrieves the articles, explicitly dealing with visualization and usability.

2.4 Quality Assessment and Data Extraction

The quality assessment strategy should be pre-defined by means of a checklist (see List 1). According to Reference [35], this instrument can be applied *"to assist data analysis and synthesis [. . .] quality data can be collected at the same time as the main data extraction activity using a joint form."* All the articles were evaluated by at least two researchers independently, using the quality criteria presented in List 1, to assess their compliance with the scope of the survey. After the assessment was accomplished, all disagreements were resolved for each article.

List 1. Quality criteria check-list

- Is there a clear contribution of a article?
- Is the system architecture/design/algorithm or experiment well-defined? Is it feasible from the article to clearly identify its features, scope of implementation, and so on?
- Is it practicable to reproduce the research?

Essential information was then extracted from the articles. This information should be sufficient to address our RQs, facilitating posterior analysis and comparison of works. The data extraction was carried out in parallel with the quality assessment. In this SLR, the information, shown in List 2, was extracted from each article.

List 2. The information extracted from each article

- Bibliographic information (title, year, authors, and affiliations)
- Main contributions and summary of the work

3 PRIMARY SELECTION

During the primary selection of articles, the four scientific databases listed in Table 2 were used. It is important to note that the original *search strings* as described in Section 2.3 could not directly be used due to practical database constraints. Not all the databases were able to support a *search string*, such as

```
firewall* AND (visual* OR graphic* OR GUI OR usab* OR (user* AND
stud*))
```

For this reason, the search strings were slightly adapted (when needed) according to each database query syntax without affecting the search space. Further details about the primary selection process can be found in Appendix A.

Table 5 shows the number of articles returned by using the search strings in each database. Note that some articles appear in more than one source, and the total number of articles is in fact fewer than the sum of findings from the four databases. Please also note that, for convenient representation, the number of articles found in the dblp database by *strings N2 and N3* (see Table 13 in Appendix A) were merged into one (*string N2*, Table 5).

4 SECONDARY SELECTION

During the primary selection, a total of 1,202 articles was retrieved from the databases. In the secondary selection, all the articles were scanned through to check their consistency with the

Table 5. Total Number of Returned Papers from Each Database (with Duplications)

String	ACM Digital Library	IEEE Xplore	dblp	Inspec	Total
N1	224	207	29	313	773
N2	116	124	17	172	429
N1+N2	340	331	46	485	1202

Table 6. Total Number of Selected Papers from Each Database (without Duplications)

String	ACM Digital Library	IEEE Xplore	dblp	Inspec	Total
N1	11	6	4	7	14
N2	20	9	10	13	31
Merged N1+N2	23	11	13	15	35

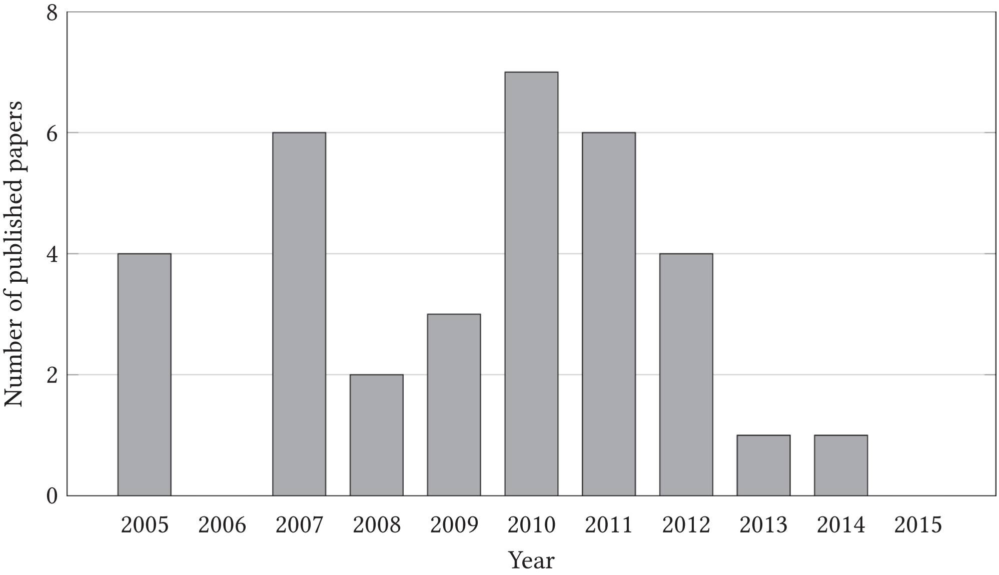


Fig. 1. Number of published articles that fit our search criteria per year.

RQs, quality criteria, and to exclude possible duplications. In this scanning process, the decisions were made by reading only the titles, abstracts and keywords. Our focus is on the *usability* of firewalls. Hence, proposals in other domains have not been considered, for example, generic policy authoring mechanisms or specific ones (RBAC-like, P3P policies and file system access control). We are aware that some mechanisms could potentially be extended for a firewall configuring process (e.g., Expandable Grids and SPARCLE), but this survey is restricted to solutions made specifically for firewalls. As a result, the second article scanning ended up with 35 unique articles; see Table 6. Figure 1 depicts the article distribution per publication year.

4.1 Removing Overlapping Works

As mentioned earlier, only the latest versions of articles were kept to avoid reading about the same research studies at different stages. This reduced the number of articles to 27. The following eight articles were excluded:

- Excluded “A Flexible and Feasible Anomaly Diagnosis System for Internet Firewall Rules” (Chao 2011). Most recent “A Novel Three-tiered Visualization Approach for Firewall” (Chao and Yang 2011).
- Excluded “A Visualized Internet Firewall Rule Validation System” (Chao 2007). Most recent “A Novel Three-tiered Visualization Approach for Firewall” (Chao and Yang 2011).
- Excluded “FAME: Firewall Anomaly Management Environment” (Hu et al. 2010). Most recent “Detecting and Resolving Firewall Policy Anomalies” (Hu et al. 2012).
- Excluded “Investigating an Appropriate Design for Personal Firewalls” (Raja et al. 2010). Most recent “It’s Too Complicated, So I Turned It Off!: Expectations, Perceptions, and Misconceptions of Personal Firewalls” (Raja et al. 2010).
- Excluded “Promoting a Physical Security Mental Model for Personal Firewall Warnings” (Raja et al. 2011). Most recent “A Brick Wall, a Locked Door, and a Bandit: A Physical Security Metaphor For Firewall Warnings” (Raja et al. 2011).
- Excluded “Towards Improving Mental Models of Personal Firewall Users” (Raja et al. 2009). Most recent “Revealing Hidden Context: Improving Mental Models of Personal Firewall Users” (Raja et al. 2009).
- Excluded “Visualizing Firewall Configurations Using Created Voids” (Morrissey and Grinstein 2009). Most recent “Developing Multidimensional Firewall Configuration Visualizations” (Morrissey et al. 2010).
- Excluded “Offline Validation of Firewalls” (Windmuller 2011). Most recent “Simplifying Firewall Setups by Using Offline Validation” (Windmüller 2013).

4.2 Data Extraction

Following the SLR protocol, at this stage, the articles should be carefully read, summarized, and assessed. All 27 selected articles were first evaluated using a check-list for quality assessment, presented in List 1. In parallel, we extracted all the relevant information described in List 2 from each article. During the *quality assessment* and *data extraction* stages, we thoroughly read each of 27 remaining articles and selected 14 of them, which propose solutions for firewall usability. A more detailed explanation of the 13 excluded articles is given in Section 5.3

To ascertain that we have not missed relevant articles, an additional step was taken. We tracked all the latter works of the authors (40 in total) of the 14 selected articles using Google Scholar to find out whether the authors have had further contributions in this field thereafter. The usage of Google Scholar is justified in this step, since it is the most complete indexing database and the number of irrelevant results is not of great concern when searching for specific authors. Thus, two articles (References [11, 16]), which were not found in the selected databases or did not match the searching terms, were found.

- “A Feasible Visualized System for Anomaly Diagnosis of Internet Firewall Rules” [11]. This article is a more recent but less complete version of the article “A Novel Three-tiered Visualization Approach for Firewall” [12]. Therefore, its original version [12] was kept in the list of the selected articles.
- “Hybrid Tree-rule Firewall for High Speed Data Transmission” [16]. This work is based on the previous research, presented in Reference [25]. However, the article focuses on the functional speed of a firewall, rather than on its usability, and thus was not included in our selection.

5 SUMMARY OF SELECTED PAPERS

All the selected articles can be classified into two main categories, depending on the type of firewall to which they are devoted: (1) PFs; (2) NFs. These categories are oriented towards different classes of users and, as a result, the topics that have been discussed in the articles are also different. PFs are the means typically used by home users, which are usually non-experts in security. Thus, articles about PFs are generally oriented toward expectations and perceptions that users have of them. NFs are mainly utilized by system/network administrators, who have an in-depth knowledge of security principles and notion of how firewalls work. Consequently, the articles in this category study ways of simplifying the configuration process, as this process has been discovered to be complex and error-prone.

The articles on PFs discuss the following topics:

- Mental Models, that is, how users behave in certain circumstances
- Requirements Elicitation, that is, what users expect from firewalls
- Usability Evaluation, that is, the authors investigate firewall solutions against known usability principles

For NFs articles, the subject matter is:

- Policy Representation
- Policy Inspection
- Misconfiguration-free Rule Sets, that is, the ways of eliminating rule sets anomalies and errors

Figure 2 depicts a taxonomy of the selected articles according to their solution space.

In this section, some figures were reproduced from the reviewed publications. They are used to give an idea about the proposals and solutions. For complete information, we refer the reader to the original publications. Unfortunately, it was not possible to use figures from all the studies due to copyright restrictions. However, we made an effort to describe those articles in more detail in text form.

5.1 Personal Firewalls

As previously mentioned, the target audience of PFs is users without any specific knowledge of IT security. Thus, they are often inclined to make wrong decisions when firewalls ask them to take actions when a security issue occurs. The articles on this topic investigate the difficulties that users have when working with PFs to improve the design of a security system, which in turn will help them to take informed decisions.

5.1.1 Mental Model. The concept of mental models is one of the most important in HCI [46]. A mental model is a model of how users reason about the system and behave in certain circumstances. It is a main goal for designers to create a user interface that helps users to form correct mental models about the particular system. It is even plausible that different mental models are built from the same interface, because each user forms a unique mental model. The difference between the mental models of designers and users creates a usability problem. A design and features that a user feels are overly complex are probably easy for a knowledgeable designer to understand. Thereby, many research articles focus on studying users' mental models from different perspectives to improve the usability of systems.

One class of articles that has been reviewed consists of making some interface improvements to a PF and observation of changes in user mental models afterwards. To make the results more accurate, it is necessary to have a significant user study. The article **(Raja et al. 2009)** falls into this category. Its goal is to present a study of participants' mental models of the Microsoft Windows Vista Firewall (MS-VF). The authors also investigate changes to those mental models after using the MS-VF basic interface and their proposed prototype, which, according to Raja et al., contains

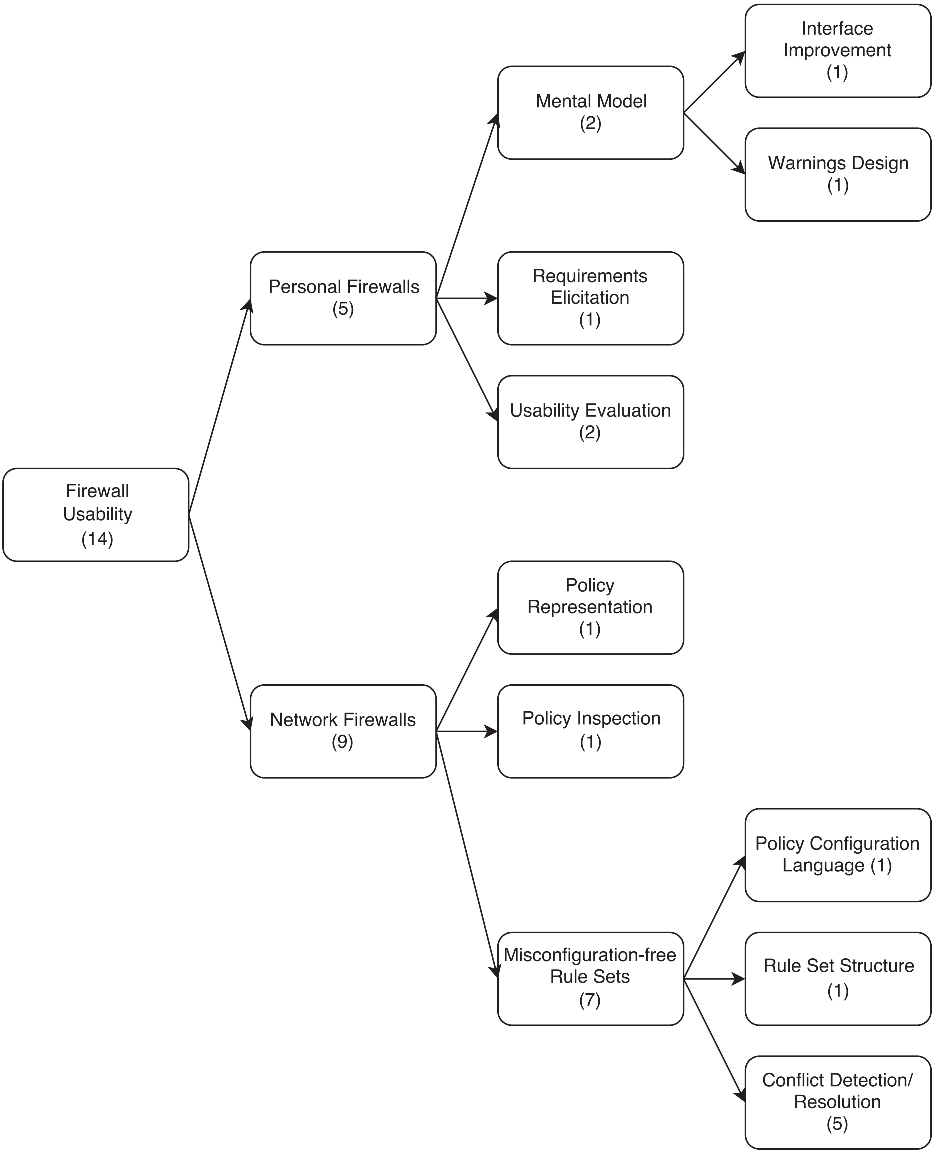

Firewall Usability (14)
Personal Firewalls (5)
Network Firewalls (9)
Mental Model (2)
Interface Improvement (1)
Warnings Design (1)
Requirements Elicitation (1)
Usability Evaluation (2)
Policy Representation (1)
Policy Inspection (1)
Misconfiguration-free Rule Sets (7)
Policy Configuration Language (1)
Rule Set Structure (1)
Conflict Detection/ Resolution (5)

Fig. 2. Categorization scheme for the selected articles. Numbers in brackets indicate quantity of articles in the particular category.

"a more explicit representation of the network context and its impact on the firewall's security state."

Two within-subjects studies with 30 participants were conducted by the authors: (1) participants used MS-VF basic interface before their prototype; (2) participants used their prototype before MS-VF basic interface. Participants' mental models were examined twice, before and after performing two common firewall tasks with two interfaces. The key results of the study were divided into four groups:

(1) Observation of changes in mental models after performing the tasks
(2) Different configuration paths of two interfaces
(3) Understanding firewall configuration
(4) Qualitative feedback on interfaces

The main drawback of the article is that the research is too specific and limited only to MS-VF.

Another class of articles on mental models proposes changes not to PFs' interface, but to the security warnings, because they are frequently misunderstood by users. It also requires a user study as a validation method. **(Raja et al. 2011)** present an "iterative design of a firewall warning using a Physical Security Metaphor" (PSM, for example, brick wall, locked door and bandit). In the study, they have mainly compared PSM-based warnings (P-warnings) and a second group, which takes firewall warnings from Comodo PFs as a basis (C-warnings).

In the main user study, 60 participants were involved. The authors created a scenario, where the participants must urgently use a chat application; before they needed to download and install it. However, the security software gave a warning when they started using the application. Then, using other interfaces, the same procedure was repeated. The participants were presented with all the interfaces at the conclusion of the experimental phase and were questioned regarding their preferences and choice of warning design. Due to the ease and faster comprehension of P-warnings, they were preferred by the majority of participants. However, 20 participants preferred the C-warnings, because they look more professional (11), can be taken more seriously (4), have more information (7), and are descriptive (2).

The work has some limitations: (1) The authors ask users to focus on firewall warnings, thus forcing participants to think more carefully than usual; (2) The user study does not capture the interruptive characteristic of firewall warnings (i.e., not an adequate representation of a real world environment).

5.1.2 *Requirements Elicitation.*

Another possible way to increase the usability of a system is to elicit requirements from users. The requirement elicitation process includes interviews, questionnaires, user observation, workshops, brainstorming, and so on. It is a component part of requirements engineering, along with analysis and documentation of the requirements.

(Raja et al. 2010) focused on requirements elicitation for PFs. To ascertain participant requirements, perceptions and misconceptions of PFs, and to improve usability, the authors selected 30 participants for semi-structured interviews. The authors used a metaphor of a black box. This black box was a security application, which was placed between the user's computer and the network. The participants were asked questions such as: "What do you want this application to do?", "What is important for you to be protected against by this application?", "Do you want the software to always have the same behavior?", and so on.

Raja et al. classified their findings into four categories:

(1) Perceptions of the black box
(2) Knowledge about PFs
(3) Context
(4) Interaction ([50])

The authors also proposed eight design recommendations for improving PFs. These recommendations are summarized here:

(1) An all-in-one solution, that is, PF integrated with other security software
(2) Improve users' awareness about warning and decisions
(3) Give recommendations to users when possible

(4) Help users to identify relevant contextual factors
(5) Provide a simple way of changing the level of security when the context changes
(6) Automate possible actions
(7) Keep users informed about automated decisions
(8) Adapt to user's knowledge and expertise

The article also provides a substantial review of literature, not only on firewalls, but also on general usable security. One shortcoming of the article, as the authors mentioned, is that the study is based on self-reported data. Another drawback is that the participants' demographics are not discussed in detail, and the age diversity could be broader (especially for groups with medium- and high-level knowledge).

5.1.3 Usability Evaluation. Usability evaluation shows how efficiently a software can be learned and used by users, and their satisfaction with it. Different methods can be applied to gather users' feedback on an existing solution. One of them is a cognitive walkthrough method. In cognitive walkthrough, to understand the system's usability, series of tasks are completed and a set of questions are asked from the perspective of a new or infrequent user.

(Alfayyadh et al. 2010) analyzed four popular PFs (ZoneAlarm, ESET Nod32 Smart Security, Norton 360, and Trend Micro Internet Security) on a Microsoft Windows XP platform with respect to eight usability principles, proposed by **(Jøsang et al. 2007)**, using a cognitive walkthrough evaluation method.

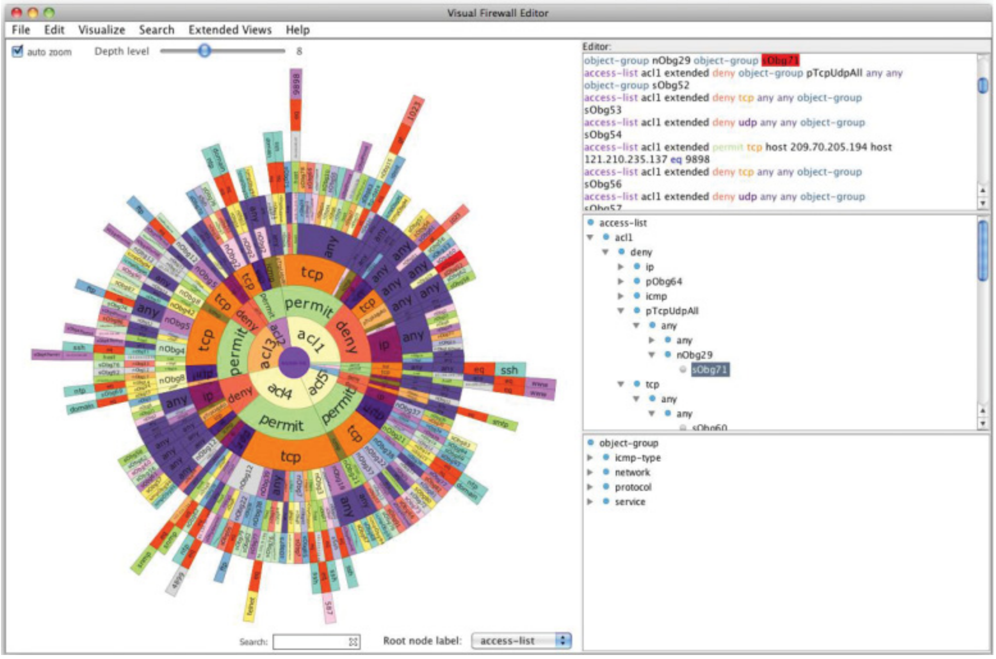
Usability problems refer to the violation of predefined usability principles during interaction with the interface of a firewall. According to the authors, some steps in the installation and configuration process could not be completed, because, on average, regular users lack the necessary knowledge of security and were not helped by information provided.

Usability issues fit into two categories: (1) Users have a vague information or a lack of it when they need to make security decisions; (2) Inconspicuous alerts and status changes. The authors recommend involving usability experts in the design process of firewalls and conducting usability tests with normal users to provide higher quality information. However, the work could have had a more detailed description of its procedures and results, for example, firewall features that have been evaluated, intermediary experimentation procedures, qualitative/quantitative assessment criteria and comparisons.

Another article **[26]** in this category assesses the effective security of 13 PFs (BlackICE PC Protection 3.6, Comodo Personal Firewall 2.0, F-secure Internet Security 2006 6.13-90, LavaSoft Personal Firewall 1.0, McAfee Personal Firewall Plus 7.0, Microsoft Windows Firewall (SP2), NetVeta Safety.Net 3.61.0002, Norman Personal Firewall 1.42, Norton Personal Firewall 2006, Sunbelt Kerio Personal Firewall 4.3.268.0, Tiny Desktop Firewall 2005 6.5.126 and the free and professional versions of ZoneAlarm 6.1.744.001) on the Microsoft Windows XP platform. All these firewalls were submitted to two use cases that typically require user decisions: (1) Connecting an application to the Internet; (2) Establishing parameters on a local host that restrict incoming connections to only one host. Certain firewall behavior, specifically cases of misuse involving port scanning and switching authentic, network-allowed applications with others, were also evaluated by the authors.

The article results in the author's recommendations to manufacturers of PFs:

- Firewalls must be more visible
- Encourage learning
- Provide the created rule, clearly establish severity as well as what needs to be done
- Principle of least privilege
- Provide a later opportunity to correct a hasty decision



Editor:

```
object-group nObg29 object-group sObg71
access-list acl1 extended deny object-group pTcpUdpAll any any
object-group sObg52
access-list acl1 extended deny tcp any any object-group
sObg53
access-list acl1 extended deny udp any any object-group
sObg54
access-list acl1 extended permit tcp host 209.70.205.194 host
121.210.235.137 eq 9898
access-list acl1 extended deny tcp any any object-group
sObg56
access-list acl1 extended deny udp any any object-group
sObg57
```

```
● access-list
▼ ● acl1
    ▼ ● deny
        ▶ ● ip
        ▶ ● pObg64
        ▶ ● icmp
        ▼ ● pTcpUdpAll
            ▼ ● any
                ▶ ● any
                ▼ ● nObg29
                    ○ sObg71
        ▼ ● tcp
            ▼ ● any
                ▼ ● any
                    ○ sObg60
```

```
● object-group
▶ ● icmp-type
▶ ● network
▶ ● protocol
▶ ● service
```

Fig. 3. Visual Firewall Editor showing the Sunburst visualization (left), the editor (top right), and two interactive tree views for access lists (middle right) and object groups (bottom right), respectively [41]. © 2012 ACM.

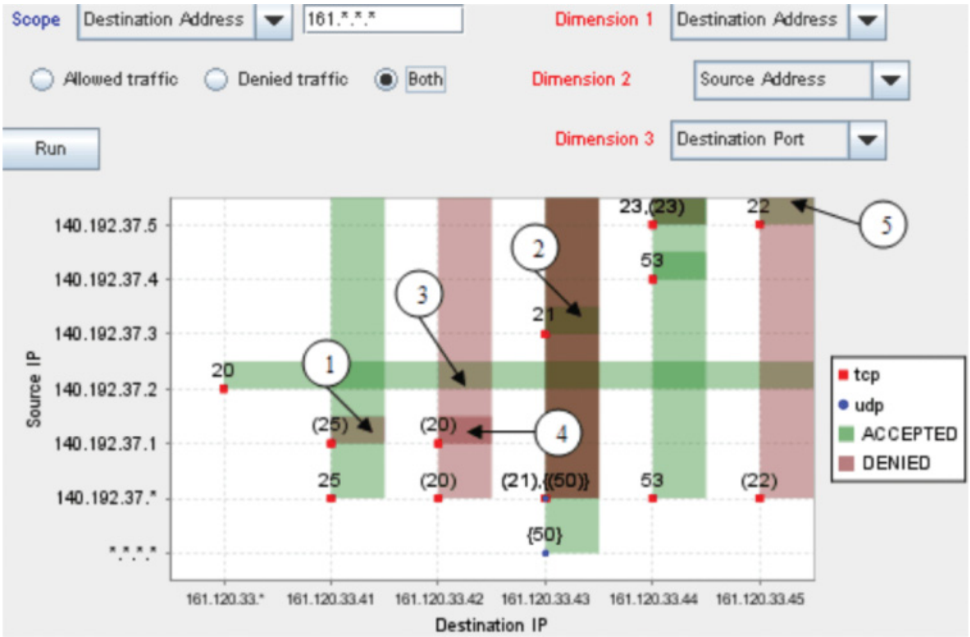
However, the work has a limited related work section and the number of use cases could have been increased.

5.2 Network Firewalls

NFs are generally tools for system/network administrators to provide network security. NFs must be properly configured to receive any benefit from using them. The process of configuring an NF is far more complex than configuring a PF, and that defines the main tendency of scientific works in this category. Hence, the articles in this category focus on investigating of tools and techniques that can assist administrators in the configuration process.

5.2.1 Policy Representation. Rule sets are usually presented in text form. It is not the best representation method, especially when a firewall configuration file contains thousands of rules. Here visualization techniques can be more useful. Data visualization refers to the techniques used to communicate information by encoding it as visual objects contained in graphics. The goal is to communicate information clearly and efficiently to users. That can be measured by various user studies. In their article, **(Mansmann et al. 2012)** present the Visual Firewall Editor (VFE), which is essentially a Sunburst diagram showing the hierarchical grouping of rules based on the characteristics they have in common. The VFE also features components for a tree view of object groups and rules as well as a configuration editor. See Figure 3 for an illustration of the Sunburst visualization.

The administrator can zoom in and out in the Sunburst view, expand nodes, select elements and visualize in the other interface components (text editor and tree view), and search for specific keywords. VFE also allows comparison of traffic matchings with the ACL entries, by adding a hit count value to show how often a rule was matched.



Scope: Destination Address; 161.*.*.*

Dimension 1: Destination Address

○ Allowed traffic ○ Denied traffic ● Both

Dimension 2: Source Address

Dimension 3: Destination Port

Fig. 4. The PolicyVis interface. This panel depicts the firewall rules and five (numbered) overlapping areas that PolicyVis suspects that might contain anomalies [59]. © 2007 T. Tran, E. Al-Shaer, and R. Boutaba.

The authors received *encouraging* feedback from five firewall experts, who *liked* the possibility of having an overview of ACLs: the two-linkage between visualization and editing; and the hit count visual analysis. The user study is limited, however, since the authors did not provide any formal methodology, and the interviewees were from their own IT department. In addition, they presented many *security visualization techniques* in various fields (mainly networking), but did not make any comparative analysis. Ultimately, the title claims "visual analysis of complex firewall configurations," but the term *complexity* was not defined anywhere in the article, and their case study was on a configuration file with only 160 rules.

5.2.2 Policy Inspection. This subcategory is similar to the one discussed earlier, except that the main assistance to administrators here is to help with inspecting a policy for a particular condition. Policy inspection also uses visualization techniques to represent the results of inspection. **(Tran et al. 2007)** propose a tool called PolicyVis; see Figure 4. PolicyVis supports visualization of NFs' rules as well as policies to efficiently enhance the understanding and inspecting of firewall policies. Tran et al. claim that "Unlike previous works that attempt to validate or inspect firewall rules based on specific queries or errors, our approach is to visualize firewall policies to enable the user to place a general inquiry such as 'Does my policy do what I intend to do?' unrestrictedly."

In their article, the authors describe design principles, implementations and application examples that deal with discovering the properties of firewall policy and rule anomalies pertaining to single or distributed firewalls. It has a detailed related work section.

The user study with 11 participants proves that PolicyVis may simplify the management of firewall policies and thus improve network security. However, some improvements can be brought by: (1) Supporting more viewing levels; and (2) A larger user study, where users cannot only evaluate the solution but also give their feedback on the tool's interface.

5.2.3 Misconfiguration-free Rule Sets. A firewall that allows more open ports than necessary, or unauthorized host connections, exposes the network to risk. Misconfigurations in rule sets are very common, as demonstrated in References [62, 68]. It is not surprising that this category is given the most attention. (Al-Shaer and Hamed 2004) classified intra- and inter-firewall anomalies in rule sets (see Lists 3 and 4).

It is worth noting that the anomaly detection/resolution in rule sets normally improves the security of systems and does not focus on usability. Thus, only articles that deal with visualization, that is efficient ways of data representation, and/or user studies, where users' satisfaction or other usability related aspects are discussed, were considered as related to the usability.

This subcategory has been divided into three classes:

(1) Policy configuration language
(2) Rule set structure
(3) Conflict detection/resolution

List 3. Classification of intra-firewall anomalies [3]

- Shadowing (S): a rule is shadowed when a previous rule matches all the packets that match this rule, such that the shadowed rule will never be activated
- Correlation (C): two rules are correlated if they have different filtering actions, and the first rule matches some packets that match the second rule and the second rule matches some packets that match the first rule
- Generalization (G): A rule is a generalization of a preceding rule if they have different actions, and if the first rule can match all the packets that match the second rule
- Redundancy (R): a redundant rule performs the same action on the same packets as another rule such that if the redundant rule is removed, the security policy will not be affected
- Irrelevance (I): a filtering rule in a firewall is irrelevant if this rule cannot match any traffic that might flow through this firewall ([3])

List 4. Classification of inter-firewall anomalies [3]

- Shadowing (Sh): a shadowing anomaly occurs if an upstream firewall blocks the network traffic accepted by a downstream firewall
- Spuriousness (Sp): a spuriousness anomaly occurs if an upstream firewall permits the network traffic denied by a downstream firewal
- Redundancy (Re): a redundancy anomaly occurs if a downstream firewall denies the network traffic already blocked by an upstream firewal
- Correlation (Co): a correlation anomaly occurs as a result of having two correlated rules in the upstream and downstream firewalls ([3])

The first class of articles in this category focuses on high-level configuration languages. Such languages serve as a substitute to standard low-level languages. An administrator is assumed to be able to get familiar with a high-level language reasonably fast and use it to configure firewalls. The configuration process becomes less complicated, since such a language is more user-friendly. The configuration created is then translated to a standard low-level form. **(Zhang et al. 2007)** present FLIP, "a high-level firewall configuration policy language for traffic access control." Rules generated by FLIP are guaranteed to be conflict-free, and the algorithm for translation has been proven to be complete and functional. Thus, FLIP has the following features:

(1) Service-oriented
(2) Modular and reusable

(3) Rule order-independent
(4) Conflict-free ([71])

The authors created an exercise on firewall policy that was completed by 12 network administrators with different levels of security knowledge to determine the practicality of FLIP. The results show that, using FLIP, administrators completed the task approximately two times faster without misconfigurations, compared to the group of people who did not use it. However, the authors do not mention how much time the participants spent on learning the syntax of the language, although this is very important in the context of their user study. This can be seen as a drawback of the article. A shortcoming of FLIP is that it does not support Network Address Translation (NAT)

The second class is dedicated to alternative structures of firewall rule sets. The traditional firewalls have their rules order-dependent and organized in lists. That creates some difficulties, for example, to find a place for a new rule, a system administrator has to look through all the preceding rules. **(He et al. 2014)** introduce a solution, called the Tree-Rule Firewall, that has a different firewall rule set structure. They also mathematically tested the Listed-Rule Firewall (a firewall that uses the list-of-rules model) to prove that this type of firewall potentially causes conflicting rules. The design is depicted in Figure 5, and implementation details of the proposed Tree-Rule Firewall are also given in the article. According to the authors, the main advantages of their solution are:

(1) No shadowed rules
(2) No need of rule swapping, because all rules will be sorted automatically
(3) No redundant rules
(4) Ease of rule design (with independent rule paths)
(5) High-speed for packet access decision ([25])

The work has a few drawbacks (some of them were mentioned by the authors): (1) It does not have all the fields that can be needed when creating a rule; (2) It does not support NAT; (3) It lacks a user study to show the enhancement of usability.

The third class of articles is also about visualization techniques. Visualization is here used to just detect or to detect and resolve rule conflicts. **(Morrissey et al. 2010)** developed two visualization approaches, Parallel Coordinates (PC) (Figure 6(a)) and Flow Picture (FP) (Figure 6(b)), and evaluated these approaches by applying them to constructed rule sets. A five-dimensional (source address, source port, destination address, destination port and protocol) visualization of the convex solid decomposition of the set of acceptable packets together with a visual representation of the created voids (using the modified Guttman algorithm [23]) uses PC. In PC, the axes are parallel vertical lines. A point with coordinates in multiple dimensions is drawn by connecting the values for each of the vertical axes with a set of line segments. Therefore, the visual result is penteracts (i.e., different rules) that are plotted, enabling the visualization of rule interactions and (most importantly) anomalies, that is, shadowing, redundancy, correlation, and generalization. The axes in FP representation are plotted in a three-dimensional space so pairs of coordinates are in orthogonal projections on separate planes, that is, source address and source port form one plane, destination address and destination port another plane, and the protocol is a one-dimensional plane (line). By connecting the coordinates from one plane to another, it is possible to create a penteract, which is in the three-dimensional space. It also has acceptance volume and created voids and visualizes interaction between rules.

According to the authors, FP provides a clearer way to show overlaps of rules. The proposed approach is, however, only a limited proof-of-concept for a new firewall visualization technique; and, thus, it only handles simple rule constructions and lacks usability studies.

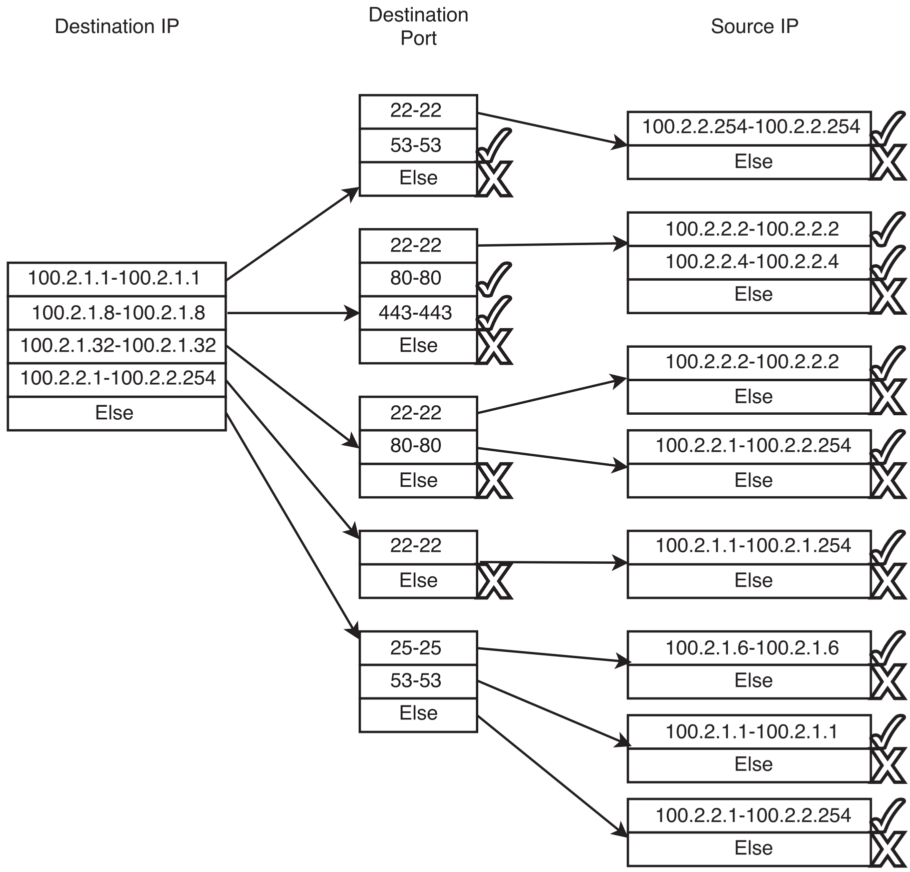
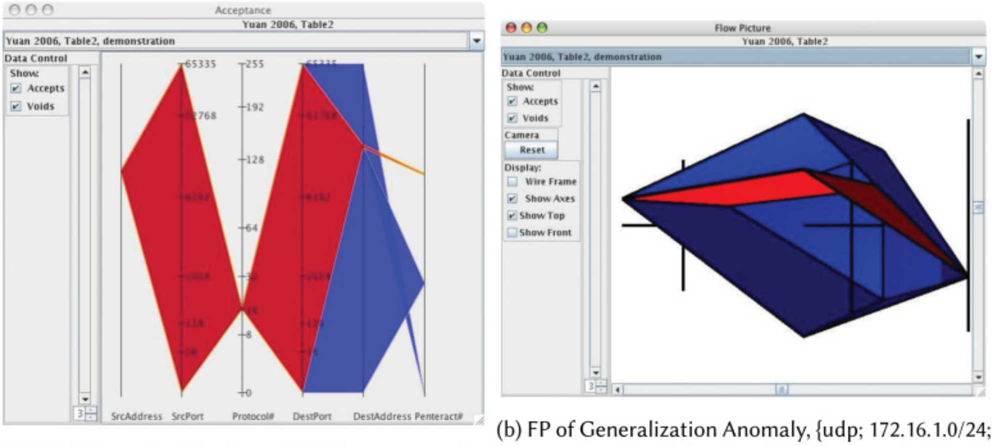


Fig. 5. Improved design of a Tree-Rule Firewall using IP address ranges and port ranges. Based on Figure 5 in Reference [25]. All IP addresses are given in ranges, that is, 100.2.1.1–100.2.1.1, means one IP address, whereas 100.2.2.1–100.2.2.254 refers to 254 IP addresses (the same reasoning is valid for ports). For example, the traffic to the destination address 100.2.1.1 should be accepted only when one of the two following conditions is true: (1) the destination port is 53; or (2) the destination port is 22 and the source address is 100.2.2.254.

(Kim et al. 2012) introduced the Firewall Policy Checker (FPC), which inspects a firewall policy for four types of anomalies (i.e., shadowing, redundancy, correlation and generalization). The work is strongly based on the Firewall Policy Advisor (FPA),[3] described in Reference [2]. Using an N-ary tree module to build and manage the policy tree, the tool processes and supports a significant quantity of rules at high speed. With this approach, the authors achieved a faster anomaly detection mechanism than using the FPA.

[3] The Firewall Policy Advisor (FPA) was proposed by Reference [2] as a user-friendly set of tools for firewall policy management. It has two main components: (1) **Policy Anomaly Detector** for identifying conflicting, shadowing, correlated and redundant rules. When a rule anomaly is detected, users are prompted with proper corrective actions; (2) **Policy Editor** for facilitating rules insertion, modification and deletion. The policy editor automatically determines the proper order for any inserted or modified rule.



(a) PC of Generalization Anomaly, {udp; 172.16.1.0/24; *; 192.168.1.0/24; * deny} and {udp; 172.16.1.0/24; *; *.*.*.*; * accept}, where the first rule denies a subset within the accept range of the latter rule.

(b) FP of Generalization Anomaly, {udp; 172.16.1.0/24; *; 192.168.1.0/24; * deny} and {udp; 172.16.1.0/24; *;*.*.*.*; * accept}, clearly showing the created void is a subset contained within packets allowed by the latter rule.

Fig. 6. Comparison between the two views. Retrieved from http://www.slideserve.com/truman/applying-visualization-to-the-management-of-firewall-rulesets/?utm_source=slideserve&utm_medium=website&utm_campaign=auto+related+load. © 2009 S. P. Morrissey.

The main interface of FPC is composed of three panes:

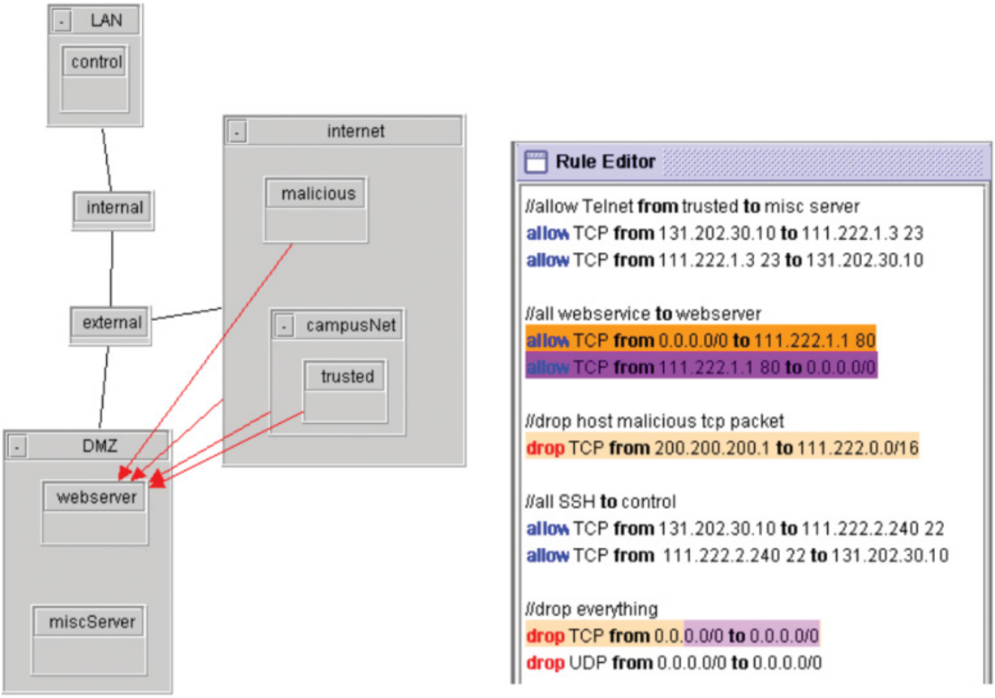
- *Pane A*, a list of detected anomalies
- *Pane B*, a list of rules related to a given anomaly (on-click event from *Pane A*)
- *Pane C*, a linked graph that shows relations between selected items in *Panes A* and *B*

Besides that, the authors also developed two 3D views for the FPC: (1) An overall view of anomalies in the firewall policy (one sphere for each anomaly type) with summary reports; (2) Risk and illegal service discovery (three spheres—$Src.IP$, $Dst.Port$, $Dst.IP$). The administrator can navigate, rotate, move, and zoom in/out of these 3D views. The authors did not motivate, explain in detail or discuss/compare the design of proposed 2D and 3D visualization schemes. There is no user study regarding the new interfaces. The main contribution, in comparison to FPA, would be the faster anomaly detection algorithm that uses N-ary trees.

(Geng et al. 2005) described a solution to aid system administrators in comprehending and updating firewall configurations by using a combination of interaction, visualization, and simulation. The authors represented the network topology as an indirected graph, as can be seen in Figure 7(a), supplemented with an entity description: Host, Network, Gateway, and Service. In this visualization scheme, an arrow in the simulation corresponds to the delivery of packets from source to destination. Clicking on the arrow will highlight all the related rules in the Rule Editor; see Figure 7(b).

The article, however, has the following drawbacks: (1) No related work section is presented; (2) The solution does not find all the misconfiguration types, for example, redundant or unused rules; and (3) The authors did not carry out any user study to prove the usability benefits brought by the solution.

(Chao and Yang 2011) proposed "a novel three-tiered visualization approach" to validate firewall rules/policy consistency. It is possible to reveal inconsistencies in firewall rule sets by noting either: (1) Anomalies between rules; or (2) Mismatching between the real and desired network



```
//allow Telnet from trusted to misc server
allow TCP from 131.202.30.10 to 111.222.1.3 23
allow TCP from 111.222.1.3 23 to 131.202.30.10

//all webservice to webserver
allow TCP from 0.0.0.0/0 to 111.222.1.1 80
allow TCP from 111.222.1.1 80 to 0.0.0.0/0

//drop host malicious tcp packet
drop TCP from 200.200.200.1 to 111.222.0.0/16

//all SSH to control
allow TCP from 131.202.30.10 to 111.222.2.240 22
allow TCP from  111.222.2.240 22 to 131.202.30.10

//drop everything
drop TCP from 0.0.0.0/0 to 0.0.0.0/0
drop UDP from 0.0.0.0/0 to 0.0.0.0/0
```

(a) Picture of overlap mistake

(b) Highlighted rules of overlap mistake

Fig. 7. Overlap mistake in a rule set [20]. The problem is caused by the wrong order of rules 3 and 5, that is, the fifth rule should appear first to drop packets from a malicious host. © 2005 National Research Council of Canada.

behavior (e.g., erroneous port number configuration). For rule anomalies, the authors adopted Al-Shaer's classification (see Lists 3 and 4).

A high-level language of specification was created using a syntax similar to that of ACL rules to describe various global security policy demands. As mentioned, the system visualization provides a three-tiered *hierarchy*:

(1) Physical Network Topology is at the bottom tier.
(2) Logical Firewall Topology (logical connectivity between firewalls and ACL-configured routers) is at the middle tier.
(3) The *Anomaly View* and the *Misbehavior View* are provided at the top tier.

The Anomaly and Misbehavior Views are the most important contribution of the work. The authors fully described their design and proof-of-concept implementation. The Anomaly View consists of three visual windows:

(1) Rule Anomaly Topology
(2) Intra-ACL Anomaly Subview
(3) Inter-ACL Anomaly Subview ([12])

The Misbehavior View consists of two windows: (1) Global Policies and (2) Misbehavior Subview. All information required from the network is automatically collected by the Network Monitor

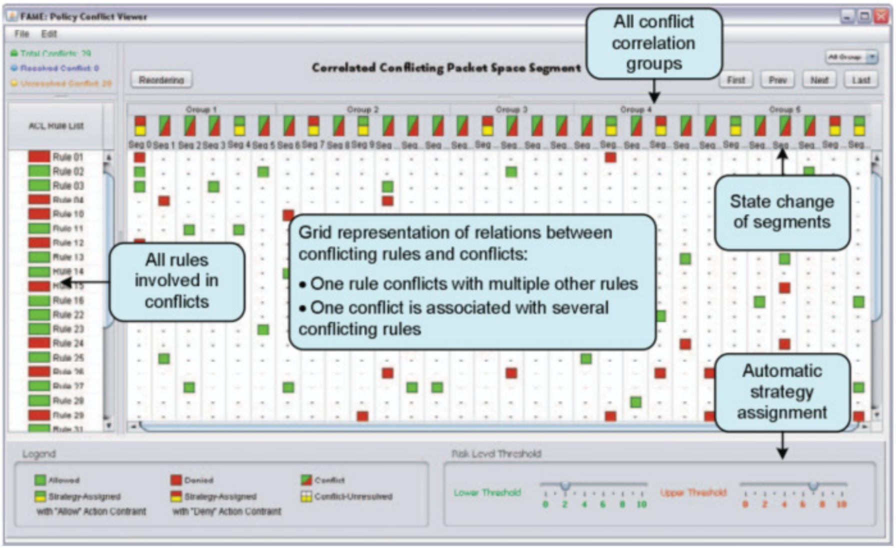


Fig. 8. Interface of FAME conflict viewer [28]. © 2012 IEEE.

subsystem. Using the information, the network monitor can visualize "the physical network topology as well as the logical firewall topology" [12]. Errors in behavior mismatching are checked for by the Behavior Mismatching subsystem, while any anomalies in inter- and intra-ACL rules are scanned for by the Anomaly Analysis subsystem. The work also has a system demonstration part. A shortcoming of the work is that it only considers packet filtering firewalls and does not include gateways/firewalls at the application level. However, as part of the authors' future work, they mention the need of more efficient methods to compute behavior mismatching, automatic correction of it, and the need to extend the system's functionality to support stateful firewalls.

In the last article **[28]**, the authors proposed the use of "a rule-based *segmentation mechanism*" combined with "a *grid-based technique*" to create an anomaly management framework that can detect and resolve firewall policy anomalies. Hu et al. have also implemented "a proof-of-concept prototype called Firewall Anomaly Management Environment (FAME)." The article describes a "rule-based segmentation technique, which adopts a Binary Decision Diagram (BDD)-based data structure to represent rules and perform various set of operations, to convert a list of rules into a set of disjoint network spaces." Policy anomalies are displayed in a matrix-based form with space segments on the horizontal axis and rules on the vertical axis.

The FAME architecture is composed of two distinct levels. The upper layer visualizes to system administrators the results of policy anomaly analysis. It has two interfaces: (1) Policy conflict viewer, as depicted in Figure 8; and (2) Policy redundancy viewer.

The authors thoroughly defined in the article what they mean by policy *conflicts* and *redundancy*. Resources containing relevant information about rules, network assets and vulnerabilities are provided along with underlying functions on the lower layer. Results showed that FAME can handle rule sets with up to 926 rules (107 conflicting segments), detect and resolve 92 percent of conflicts, thus, outperforming previous methods, such as References [3, 70]. The work presents a substantial literature review on firewall anomaly detection and resolution (e.g., References [3, 5, 19, 70]) and on other assisting methods (e.g., References [10, 52]). An important limitation of the article is the absence of evaluation studies that demonstrate the efficiency and effectiveness of the solution.

5.3 Excluded Papers

The following 13 articles were excluded after careful reading. We consider these articles to be out of the scope of our SLR. The reasons for exclusions are summarized in the following text.

5.3.1 Conflict Classification and Analysis of Distributed Firewall Policies. The authors of Reference [1] aimed to classify all intra- and inter-firewall anomalies in rule sets. This work focuses on the security rather than the firewall usability.

5.3.2 Security Analysis of Firewall Rule Sets in Computer Networks. In Reference [33], the authors proposed a comparison of two methodologies for firewall rule set analysis: Policy Tree and Relational Algebra. The performance of each method is discussed as regards their ability to find and solve anomalies in the given rule set. Again, the article is security-oriented.

5.3.3 Does Context Influence Responses to Firewall Warnings? (Mahmoud et al. 2012) investigated whether a response to firewall warnings is influenced by what the user was doing at that moment. This article thus deals with decision making.

5.3.4 Fast, Cheap, and in Control: Towards Pain-Free Security! A set of firewall configuration complexity metrics was proposed in Reference [8]. However, no user study or formal proof is provided to substantiate the usability improvement.

5.3.5 Locking the Door but Leaving the Computer Vulnerable: Factors Inhibiting Home Users' Adoption of Software Firewalls. In Reference [37], the authors studied users' perception and attitude to using firewalls. Hence, it is out of the scope of our SLR.

5.3.6 Network Infrastructure Visualization Using High-Dimensional Node-Attribute Data. In Reference [22], a version of targeted projection pursuit is presented. The article contains an analysis of Intrusion Detection System's (IDS) logs, and, thus, it is considered a non-related work.

5.3.7 Network Firewall Visualization in the Classroom. Reference [63] describes an educational tool for teaching firewall configuration and network security. This is out of the scope of our SLR. The visualization mechanism helps teaching but will not improve firewall usability, since the mechanism is not designed for real-life systems.

5.3.8 Simplifying Firewall Setups by Using Offline Validation. (Windmüller 2013) describes the concept of testing firewall configurations by validating a rule set against virtual packets in a simulation environment. Thus, the administrator can verify whether the packets are accepted or denied as expected. This work is more related to the compliance between a rule set and the corresponding security policy.

5.3.9 User Help Techniques for Usable Security. (Herzog and Shahmehri 2007b) described different methods for helping users with security applications and theoretically analyzed their usefulness. The work is more a classification of helping techniques rather than a usability study.

5.3.10 VAFLE: Visual Analytics of Firewall Log Events. The tool presented in Reference [21] is used to monitor a network but does not aim to help the administrator during firewall configuration.

5.3.11 Visual Discovery in Computer Network Defense. The Visual Assistant for Information Assurance Analysis (VIAssist) was designed and described in Reference [17]. However, this tool is not related to firewalls.

5.3.12 Visual Firewall: Real-time Network Security Monitor. (Lee et al. 2005) introduced a tool called VisualFirewall that has four implemented views: (1) Real-Time Traffic; (2) Visual Signature;

(3) Statistics; and (4) IDS Alarm. It seems to be a very useful tool, but more focused on monitoring firewall operation than helping in the configuration process of firewalls.

5.3.13 Perceptions of End Users on the Requirements in Personal Firewall Software: An Exploratory Study. (Hazari 2005) conducted a study to find out general user requirements for PFs. The respondents classified properties, such as *cost*, *ease of use*, *performance*, *support,* and so on, by importance: from *Most Important (+2)* to *Least Important (−2)*. This work can be useful for organizations that need to select appropriate security software, but it cannot be considered to be usability related.

6 SLR SYNTHESIS AND DISCUSSION

This SLR allowed us to identify and classify existing solutions for usability for PFs and NFs. In particular, the summary of selected articles and the proposed taxonomy, provided in Section 5, are now used to support the SLR's synthesis. Here, we offer a digest of the information previously presented in the article. To do so, we break down the SLR synthesis and discussion into three parts: (1) An overall discussion about the SLR process and methodology; (2) A general discussion on the usability of firewalls; and (3) Specific discussions of PFs and NFs.

6.1 SLR Overall Discussion

In the course of the SLR's primary study, the search and selection of articles were done during April and May, 2016. Four databases were searched, using a pre-defined set of keywords and looking for articles published from 2005 to 2016. The primary searches returned a total of 1,202 articles, where their titles and abstracts were examined. During the secondary study, we made a consistency analysis to double-check the selected articles to remove possible duplications, non-peer-reviewed works (e.g., book chapters, technical reports), and to verify adherence to our RQs and inclusion criteria.

As a result, the secondary study continued with a reduced body of 35 articles. Before the stages of data extraction and quality assessment, we identified articles from the same authors that described more or less the same research, but in different stages. Only the most recent/complete publication of each author/group was selected, thus reducing the number of articles to 27.

Even though we had many filters along the process, from the body of 27 articles, some were still out of the scope of this SLR. Although they dealt with usability and/or visualization, their primary goal was different, for instance:

- The research targeted IDS.
- The research focused on the log analysis or network monitoring instead of the firewall configuration.
- The research aimed at teaching/education processes regarding firewall training.

In the end, this SLR has identified 14 articles that deal with the usability of a firewall configuring process, for both PFs and NFs.

A limitation of this SLR refers to its choice of search terms and databases. The SLR is an exhaustive search process that forces a strict scope delimitation to make it feasible. It is therefore possible that some articles were not found, because they use other terms to refer to usability or because they were published in other scientific databases. To reduce the probability of missing relevant articles, we tracked the further publications of the authors of the 14 selected articles in the field to see whether the authors have contributed more thereafter. SLRs are extensible by definition so researchers could still refine or improve them in the future.

Table 7. Summary of PF Papers

Paper	Focus	Solution	Evaluation method[1]	Addressed usability aspects[2]	Paper summary
(Raja et al. 2009)	Mental Model	Interface Improvement	User Study	L, E/S, S	Study of users' mental models of Microsoft Windows Vista Firewall with basic and improved interfaces and a comparison of them. Two user studies with 30 participants
(Raja et al. 2011)	Mental Model	Warnings Design	User Study	E/S, S	Design of firewall warnings using a physical security metaphor and examination of its efficacy. User study with 60 participants.
(Alfayyadh et al. 2010)	Usability Evaluation	Cognitive Walkthrough	N/A	E, E/S	Usability analysis and comparison of four PFs. Cognitive walkthrough evaluation against eight usability principles (Jøsang et al. 2007)
(Herzog and Shahmehri 2007a)	Usability Evaluation	Cognitive Walkthrough	N/A	E, E/S	Usability analysis and comparison of 13 PFs. Submitting firewalls to different use cases and following cognitive walkthrough evaluation.
(Raja et al. 2010)	Requirements Elicitation	Semi-structured Interviews	N/A	E/S, S	Users' knowledge, requirements, perceptions and misconceptions of PF warnings. Summary of design recommendations. Semi-structured interviews with 30 participants.

[1] N/A - Not applicable.
[2] E - Efficiency, L - Learnability, E/S - Errors/Safety, S - Satisfaction.

6.2 On Usability of Firewall Configuration

To synthesize the key information extracted from studies reviewed in Section 5, Tables 7 and 8 present a short summary of the 14 reviewed articles. Both tables contain the main goal of the work, the contribution and additional points on the user study, and usability tests. The tables are not intended to give a complete overview of the articles but rather to offer a quick reference for the reader.

Now, to start the discussion it is important to distinguish the fundamental differences between the usability of PFs and NFs. In brief, usability boils down to three elements: the user group, the

Table 8. Summary of NF Papers

Paper	Focus	Solution	Evaluation method	Addressed usability aspects[1]	Paper summary
(Mansmann et al. 2012)	Policy Representation	Visualization	User Feedback	E, E/S, S	Visual representation of complex firewall configurations. Visualization: Sunburst. User feedback from five firewall administrators.
(Tran et al. 2007)	Policy Inspection	Visualization	User Study	E, L, E/S	Policy inspection and its visualization. Visualization: Flexible 2D graph with multiple fields. User study with 11 participants.
(Zhang et al. 2007)	Misconfiguration-free Rule Sets	Policy Configuration Language	User Study	E, E/S	High-level firewall policy language. Service-oriented, modular and reusable, rule order-independent, conflict-free. User study with 12 network administrators.
(He et al. 2014)	Misconfiguration-free Rule Sets	Rule Set Structure		E, E/S	Tree-rule firewall. No redundant or shadowed rules, no need to swap rules, ease of rule design and high speed decisions.
(Chao and Yang 2011)	Misconfiguration-free Rule Sets	Conflict Detection/ Resolution	System Demonstration	E, E/S	Three-tiered visualization approach for firewall rule validation. Policy/rule set consistency: Anomaly detection (GCSRI + ShSpReCo). Visualization: three-tiered approach.
(Geng et al. 2005)	Misconfiguration-free Rule Sets	Conflict Detection/ Resolution		E, E/S	A tool that combines simulation and visualization of firewall configuration. Typos and partial anomaly detection (only CS). Visualization: indirected graph with entity description.
(Hu et al. 2012)	Misconfiguration-free Rule Sets	Conflict Detection/ Resolution	System Evaluation	E, E/S	Firewall Anomaly Management Environment (FAME). Anomaly detection (GCSR) and resolution. Visualization: grid-based technique.
(Kim et al. 2012)	Misconfiguration-free Rule Sets	Conflict Detection/ Resolution		E, E/S	Policy inspection using anomaly detection. Anomaly detection (GCSR), comparison with FPA (PolicyVis). Visualization: 3D approach.
(Morrissey et al. 2010)	Misconfiguration-free Rule Sets	Conflict Detection/ Resolution		E, E/S	Multidimensional firewall configuration visualizations. Anomaly visualization (GCSR). Comparison of visualization schemes: - Parallel Coordinates (2D); - Flow Picture (3D).

[1] E - Efficiency, L - Learnability, E/S - Errors/Safety, S - Satisfaction.

tasks, and the context of use. By changing one of the elements you basically have a new usability problem to solve. In the cases of PFs and NFs, the user group and the context of use drastically change. For PFs, we typically have ordinary end users that interact with the firewall while performing other tasks. That is, configuring the firewall is not their main task. Warnings might pop up in a very interruptive manner, and the firewall as a security solution might usually get in the way of the users' main tasks. For NFs, we usually have computer experts (e.g., system/network administrators, security experts) interacting with the firewall, and its configuration is considered an important (if not the main) task.

Hence, articles on PFs and NFs address different usability aspects. These are noted in the column "Addressed usability aspects" of Tables 7 and 8. While the articles on PFs stress errors/safety and user satisfaction, they less often address efficiency and learnability, and never memorability. All the articles on NFs tend to focus on efficiency and errors/safety but not on memorability or user satisfaction (only (Mansmann et al. 2012) addressed satisfaction). Also, firewall administrators have domain and system-specific knowledge, leaving learnability often to be overlooked. As described in Reference [9], "systems administrators are often regarded as a special breed of user that possesses limitless technical knowledge and skill. ... able to overcome any software shortcomings that would otherwise frustrate or prevent the average user from completing required tasks." Such an assumption is obviously erroneous and ignoring some of the usability aspects does not help to solve the firewall misconfiguration problems, exemplified in Reference [68]. So far, articles on NFs have presented different promising ideas, mostly on visualization mechanisms, that could improve the usability of a firewall configuration. Information visualization techniques help administrators to efficiently comprehend and navigate over large amounts of data, such as firewall rule sets. Or, as described by the authors of Reference [59], "Firewall policy visualization helps users **understand** their policies **easily** and **grasp** complicated rule patterns and behaviors **efficiently**."

While the surveyed articles display an encouraging step forward in usability for firewalls, it is also clear that more serious usability evaluations and user studies must be conducted. There are recommendations and guidelines for PFs and NFs (e.g., see Tables 9 and 10), but they must be put into practice; solutions should be further evaluated, and recommendations should be updated accordingly. To the best of our knowledge, there is no established criteria for evaluating the usability of firewall configuration (either for PFs or NFs). In fact, none of the selected articles defined usability in their work. That should not be seen as a complete void, however. Usability is often used as a rather loose term to describe ease of use, or more specific aspects, such as effectiveness and efficiency. That is, authors tend to describe usability in their own terms, without observing well-established definitions in the area.

However, firewall developers may rely on design principles from the usable security area. Indeed, all articles concerning PFs refer to principles described in References [64, 69] or related literature. On the other hand, the articles on NFs largely miss references to the topics of usability and usable security. Although most of the research on usable security focuses on regular end users, there is relevant research that focuses on computer experts and security tools (e.g., References [9, 15, 31]).

In summary, we pinpoint the open challenges that exist in the field. First, to say how usable different solutions are, it is necessary to be able to measure the usability. For this purpose, a set of mathematically formalized usability metrics is required. The set of metrics will help to optimize the usability of one solution and to make it possible to compare different solutions. While this type of research is missing in the field of firewalls, the issue was addressed in the field of access control [6]. The second open challenge stems from the first one: how to address each usability aspect, shown in Table 1, and how each of them influences the overall usability. The third challenge is that not all the aspects of usability are yet addressed in the literature, that is, PFs' efficiency, learnability, and memorability and NFs' learnability, memorability, and satisfaction are sparsely studied. These issues need to be addressed in the future to consolidate the field and make firewall solutions more usable.

6.2.1 Personal Firewalls. In the SLR, we reviewed five different articles about PFs, each one with different goals regarding usability. This reveals a potential lack of research in the field. Apart from that, comparisons between approaches are significantly impaired. Notwithstanding, authors commonly agreed about the common poor usability of PFs and drew similar conclusions about the importance of effective communication with the user (e.g., exploiting text, alerts and color schemes) to educate users and support informed decisions. As further emphasized in Reference [48], designers should not "hide inner complexity for the sake of interface simplicity," since the lack of contextual information leads to "an ineffective mental model" and "dangerous errors."

Interestingly, although all articles clearly addressed the area of usability, none of them clearly defined it. In fact, only Reference [4] mention "eight security usability principles," previously introduced in Reference [32]. In brief, the principles refer to users' security *actions* and *conclusions* about the security state of the system: (1) users must understand security functions; (2) users must have sufficient knowledge/information to make decisions; and (3) the mental and physical load must be tolerable. However, such principles do not follow well-known definitions, for example, Reference [30] or [45], yet they could be linked to usability aspects such as efficiency and error/safety.

The authors also examined slightly distinct problems in the usability of PFs. Greater concern is given to user's *mental models* regarding PFs in References [48, 50], and specifically for PFs' warnings in Reference [49]. In References [4, 26], the authors are more concerned about the general usability analysis of PFs. Three of five articles also explicitly provide a list of usability issues and/or recommendations as part of their contribution, as presented in Table 9. Finally, as mentioned in Reference [49], this kind of research can also potentially benefit the broader community of usable security, since the importance of security mental models, metaphors and visual cues to convey risk is not limited to PFs.

6.2.2 Network Firewalls. The articles reviewed present eight different visualization schemes; see Table 8. It is a common trend for all the articles to exploit visualization mechanisms to facilitate the task of firewall configuration. Or, as pointed out by the authors of Reference [41], there is "pressing need to visually support network administrators in their complex task." Specifically, Reference [41] also stressed that "structural information implicitly contained in the rule set should be made explicit in the visualization techniques." This structural information can be understood as ACLs, rulesets, network addresses, ports, protocols, and rule interactions; in fact, that is exactly what all visualization schemes attempted to do: make implicit information explicit.

However, despite the number of schemes, there is a lack of studies that conducted usability tests—as aforementioned, only the feedback from users in Reference [41] and a user study in Reference [59] were discussed. Therefore, it is still not clear which technique would achieve higher usability when configuring NFs. None of the articles actually defined usability, even if their research clearly approaches the topic. However, the authors often used the words *efficiency* and *effectiveness* to describe usability goals (e.g., References [12, 28, 59, 71]) as well as terms such as *ease of use* and *ease to understand* (e.g., References [12, 25, 34, 44]). Interestingly, only Reference [41] addresses the term *accessibility*, which is a concept usually associated with usability (i.e., "accessibility as usability for people with the widest range of capabilities" [7]).

Very few authors clearly described design principles or goals for their visualization schemes. The exceptions are References [20, 59]. A summary of their design principles for visualization is presented in Table 10.

Table 9. Summary of Usability Issues and Recommendations for PFs

Paper	Usability issue and/or recommendation
(Alfayyadh et al. 2010)	Users were faced with **insufficient information** although they were required to make a choice that affected their security. There was often **poor visibility** of alerts or status changes. Surveying average users on usability would enable those responsible for engineering firewalls to offer clearer information. Have experts on usability participate from the infancy of the design process.
(Herzog and Shahmehri 2007a)	**Firewalls must make themselves more visible.** Use clear logos so the user can identify the firewall; the user should be informed about certain actions running in the background. **Encourage learning.** Make an effort to teach users, for example about ports and IPs, so they can make informed decisions. **Give the user a chance to revise a hasty decision later.** Since it is not a primary task, some reminders should be used. **Enforce least privilege** wherever possible. **Indicate severity, indicate what to do and show the created rule.** Provide warnings about dangerous actions and inform users.
(Raja et al. 2010)	**All-in-one solution.** PF should be integrated with other security software; with consistent configuration and terminology. **There should be an awareness in warnings and notices.** They can help users to create a correct functional mental model of PFs. **Recommendations should be given in warnings.** Explicit recommendation or threat level and instructions should be given. **Decision based on context.** Users should be assisted in making connections between the necessary security levels and contextual factors. **Allow an easy change of the security level.** If a relevant contextual factor changes, then the user should be offered a clear path to adjust security accordingly. **Automate possible actions.** **There should be an awareness of automated decisions.** Keep users informed about the automated decisions. **Adapt to users' knowledge and expertise.** There should be basic and advanced interfaces, configuration panels, types of warnings and recommendations.

Most authors also did not cite each other's work, an issue possibly caused by the short timespan between the publications (i.e., 2010–2012). Notwithstanding, the more integrative approaches introduced by References [12, 28] were able to offer the most appropriate answers to the problem.

7 CONCLUDING REMARKS

Firewall configuring is a process that is complicated and prone to error that administrators and/or end users must tackle in their daily work. The misconfiguration of firewalls leads to a broad number of vulnerabilities in the network; environments with distributed and multiple firewalls just make matters worse. We, therefore, understand that the firewall configuring process can greatly benefit

Table 10. Summary of Design Principles and Goals for NF Visualization

Paper	Design principle and/or goal
(Tran et al. 2007)	**Simplicity**. Easy for users to comprehend and manage. **Expressiveness**. Utilize every suitable field, rule order and action to illustrate potential behaviors of those rules with a comprehensive visualization. **Flexible visualization scope**. To comprehend the policy from different perspectives and to add convenience, fields for graph coordinates should be selected by users. **Ability to compress, focus and zoom**. Focusing enables investigation of any policy rule anomaly. The policy can be viewed in its entirety or only partially depending upon which ranges were selected in a particular field. **Ability to use policy segmentation**. **Ability to use symbols, colors, notations**. The internal policy interactions and level of performance can be best understood by users when only notation, coloring and symbolism crucial for understanding are utilized.
(Geng et al. 2005)	The picture must require a very limited amount of mental processing and closely match the firewall behavior to the mental image the user has of the policy, but provide sufficient information for an administrator to grasp network connectivity and global topology. The pictures should be recognizable. Tool usability relies on the clarity and organization of the imagery.

from usability studies. This SLR presents an effort toward the synthesis of the state of the art on firewall configuration from the usability perspective. The whole SLR methodology has been described in detail; in the end, we fully reviewed and evaluated 14 relevant articles. As a result, this SLR provides a sound overview and discusses some gaps in the area of firewall usability.

The most significant issue refers to the lack of usability testing and/or user studies regarding the proposed visualization schemes. This leads us to a multitude of unwarranted visualization schemes that cannot prove themselves usable. To close this gap, more comparative analysis and user studies should be done to corroborate the methods of various works and give evidence on the most adequate visualization schemes. Further, it is clear that work on PFs that is reviewed could adopt detailed methodologies on usability and user tests (compared to the articles on NFs). That is, in short, both areas could mutually benefit from the exchange of ideas with respect to the usable security.

APPENDIX

A PRIMARY SELECTION SEARCH DETAILS

The *Results* column in Tables 11–14 represents the number of articles retrieved in a particular search. The date at which the search was performed can be found in the *Date* column. In the following, we present practical comments, adapted search terms and results for each of the databases.

A.1 ACM Digital Library

Search queries were performed using the “Advanced Search” feature. Papers from “The ACM Guide to Computing Literature” were selected. Three filters were then applied to the search results:

Table 11. Search Strings and Number of Retrieved Papers from the ACM Digital Library

N	String	Results	Date
1	firewall AND (rule OR policy) AND (configure OR anomaly OR (consistent OR inconsistent) OR conflict OR error)	224	5 April 2016
2	firewall AND (visualization OR visualise OR graphical OR GUI OR usability OR (user AND study))	116	5 April 2016

(1) Search Title, Abstract and Keywords (with a help of the query commands *title*, *recordAbstract* and *keywords.author.keyword*).
(2) Published since 2005.
(3) All publications → Proceedings and Periodical.

For some reason, the asterisk ("*") did not work properly. By using the same filters, only four articles were retrieved when searched with the following string:

```
firewall* AND (visual* OR graphic* OR GUI OR usab* OR (user* AND
stud*))
```

Instead, the following two logically equivalent strings were therefore used:

```
(1) (firewall OR firewalls) AND ((rule OR rules) OR (policy
OR policies)) AND ((configure OR configuration OR configuring
OR configurations) OR (anomaly OR anomalies) OR (consistent OR
inconsistent OR consistency OR inconsistency) OR (conflict OR
conflicts OR conflicting) OR (error OR errors))
```

```
(2) (firewall OR firewalls) AND (((visual OR visualization OR
visualisation OR visualizing OR visualising OR visualize OR
visualise) OR graphical OR GUI) OR (usable OR usability) OR ((user
OR users) AND (study OR studies)))
```

However, it was decided to shorten the search strings by eliminating word forms that do not affect the total number of results. The final search strings can be found in Table 11.

A.2 IEEE Xplore

Searches were again conducted using the "Advanced Search" feature with the following parameters:

(1) Metadata Only.
(2) Content types: Conference Publications, Journals & Magazines.
(3) Publication year: 2005 to present.

One restriction was found during the search process: a user of the database is not allowed to use more than five wildcards in a query. Hence, we applied a similar technique to the one used for the ACM Digital Library. For instance, the following search string was taken:

```
firewall* AND (rul* OR polic*) AND (config* OR anomal*)
```

Table 12. Search Strings and Number of Retrieved Papers from the IEEE Xplore

N	String	Results	Date
1	(firewall AND (rule OR policy) AND ((((config* OR anomal*) OR *consisten*) OR conflict*) OR error))	207	6 April 2016
2	(firewall*) AND ((((visual*) OR graphic*) OR usab*) OR (user AND stud*))	124	6 April 2016

It was then checked whether the replacement of wildcards by certain word forms affects the number of results. If not, then the wildcard was removed from the search string. In the end, the following string was derived:

```
firewall AND (rule OR policy) AND (config* OR anomal*)
```

The procedure was repeated for all remaining search terms of Group 3 and Group 4 … in Table 3 and 4. It is worth noting that the tuple (G4, T3) was removed, since it did not produce any new results. The two strings in Table 12 were used when searching the database instead of the initial ones.

A.3 dblp

The dblp does not have a feature "Advanced Search." Instead, it has a field "search for publications" in the top right corner. A publication type can be seen in *Refine list → refine by type*. A year range could not be specified, but results returned were ordered by year, and that was therefore convenient to use. Another option when searching the database is to see a number of articles per year through the *Refine list → refine by year* feature. The dblp uses the following syntax: (1) *AND*: separate words by space; (2) *OR*: connect words by pipe symbol (|); (3) prefix search, that is a string *rul* matches rule, rules, and so on. Due to the constraints of the syntax, three search strings were created, see Table 13. Note that tuple (G3, T5) was deleted without affecting the number of retrieved articles.

Table 13. Search Strings and Number of Retrieved Papers from the dblp Database

N	String	Results	Date
1	firewall rul|polic config|anomal|consisten|conflict	29	6 April 2016
2	firewall visual|gui|graphic|usab	16	6 April 2016
3	firewall user stud	1	6 April 2016

A.4 Inspec

The Inspec website does not provide the users with a search interface. The searches were therefore made via the EBSCOhost[4] engine using the "Advanced Search" feature. By using this interface, it was possible to use the asterisk ("∗") and boolean operators together with the keywords. It was also possible to specify the part of article in which to look, for example, *All Text, Abstract, Title*. However, there is no "Metadata" field. It was therefore decided to search *Abstracts*.

[4] https://www.ebscohost.com.

Table 14. Search Strings and Number of Retrieved Papers from the Inspec Database

N	String	Results	Date
1	(firewall*) AND (rul* OR polic*) AND (config* OR anomal* OR *consisten* OR conflict* OR error*)	313	6 April 2016
2	(firewall*) AND (usab* OR visual* OR graphic* OR GUI OR (user* AND stud*))	172	6 April 2016

The following filters were applied:

(1) Publication date: 2005–2016.
(2) Source types: Conference Papers and Academic Journals.
(3) Language: English.

The search strings are presented in Table 14.

REFERENCES

[1] Ehab S. Al-Shaer, Hazem Hamed, Raouf Boutaba, and Masum Hasan. 2005. Conflict classification and analysis of distributed firewall policies. *IEEE J. Select. Areas Commun.* 23, 10 (2005), 2069–2084.

[2] Ehab S. Al-Shaer and Hazem H. Hamed. 2003. Firewall policy advisor for anomaly discovery and rule editing. In *Proceedings of the IFIP/IEEE 8th International Symposium on Integrated Network Management*. IEEE, 17–30.

[3] Ehab S. Al-Shaer and Hazem H. Hamed. 2004. Discovery of policy anomalies in distributed firewalls. In *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'04)*, Vol. 4. IEEE, 2605–2616.

[4] Bander Alfayyadh, James Ponting, Mohammed Alzomai, and Audun Jøsang. 2010. Vulnerabilities in personal firewalls caused by poor security usability. In *Proceedings of the 2010 IEEE International Conference on Information Theory and Information Security (ICITIS'10)*. IEEE, 682–688.

[5] Florin Baboescu and George Varghese. 2003. Fast and scalable conflict detection for packet classifiers. *Comput. Netw.* 42, 6 (2003), 717–735.

[6] Matthias Beckerle and Leonardo A. Martucci. 2013. Formal definitions for usable access control rule sets from goals to metrics. In *Proceedings of the 9th Symposium on Usable Privacy and Security*. ACM, 2.

[7] Nigel Bevan, James Carter, and Susan Harker. 2015. *ISO 9241-11 Revised: What Have We Learnt About Usability Since 1998?* Springer International Publishing, Cham, 143–151. DOI: http://dx.doi.org/10.1007/978-3-319-20901-2_13

[8] Sandeep N. Bhatt, Cat Okita, and Prasad Rao. 2008. Fast, cheap, and in control: Towards pain-free security! In *Proceedings of the Conference on Large Installation System Administration (LISA'08)*. USENIX Association, 75–90.

[9] Michael Bingham, Adam Skillen, and Anil Somayaji. 2014. Even hackers deserve usability: An expert evaluation of penetration testing tools. In *Proceedings of the 9th Annual Symposium on Information Assurance (ASIA'14)*. 23–31. Retrieved from http://www.albany.edu/iasymposium/proceedings/2014/ASIA14Proceedings.pdf.

[10] Carolyn Brodie, Clare-Marie Karat, and John Karat. 2006. An empirical study of natural language parsing of privacy policy rules using the SPARCLE policy workbench. In *Proceedings of the Symposium on Usable Privacy and Security (SOUPS'06) (ACM International Conference Proceeding Series)*, Vol. 149. ACM, 8–19.

[11] Chi-Shih Chao. 2012. A feasible visualized system for anomaly diagnosis of internet firewall rules. *J. Commun. Comput.* 9 (2012), 679–691.

[12] Chi-Shih Chao and Stephen Jen-Hwa Yang. 2011. A novel three-tiered visualization approach for firewall rule validation. *J. Vis. Lang. Comput.* 22, 6 (2011), 401–414.

[13] Bill Cheswick. 1990. The design of a secure internet gateway. In *Proceedings of the USENIX Summer Conference*. Citeseer.

[14] William R. Cheswick, Steven M. Bellovin, and Aviel D. Rubin. 2003. *Firewalls and Internet Security: Repelling the Wily Hacker*. Addison-Wesley Longman Publishing Co., Inc.

[15] Sonia Chiasson, Robert Biddle, and Anil Somayaji. 2007. Even Experts Deserve Usable Security: Design guidelines for security management systems. Retrieved from http://cups.cs.cmu.edu/soups/2007/workshop/Design_Gu idelines.pdf.

[16] Thawatchai Chomsiri, Xiangjian He, Priyadarsi Nanda, and Zhiyuan Tan. 2016. Hybrid tree-rule firewall for high speed data transmission. *IEEE Trans. Cloud Comput.* (2016).

[17] Anita D. D'Amico, John R. Goodall, Daniel R. Tesone, and Jason K. Kopylec. 2007. Visual discovery in computer network defense. *IEEE Comput. Graph. Appl.* 27, 5 (2007), 20–27.

[18] Rogerio De Paula, Xianghua Ding, Paul Dourish, Kari Nies, Ben Pillet, David F. Redmiles, Jie Ren, Jennifer A. Rode, and Roberto Silva Filho. 2005. In the eye of the beholder: A visualization-based approach to information system security. *Int. J. Hum.-Comput. Studies* 63, 1 (2005), 5–24.

[19] Joaquín García-Alfaro, Nora Boulahia-Cuppens, and Frédéric Cuppens. 2008. Complete analysis of configuration rules to guarantee reliable network security policies. *Int. J. Inf. Sec.* 7, 2 (2008), 103–122.

[20] Weiwei Geng, Scott Flinn, and John M. DeDourek. 2005. Usable firewall configuration. In *Proceedings of the 3rd Annual Conference on Privacy, Security and Trust.* Retrieved from http://www.lib.unb.ca/Texts/PST/2005/pdf/geng.pdf.

[21] Mohammad Ghoniem, Georgiy Shurkhovetskyy, Ahmed Bahey, and Benoît Otjacques. 2013. VAFLE: Visual analytics of firewall log events. In *Proceedings of the IS&T/SPIE Conference on Electronic Imaging.* International Society for Optics and Photonics, 901704–901704.

[22] Helen Gibson and Paul Vickers. 2012. Network infrastructure visualisation using high-dimensional node-attribute data. In *Proceedings of the IEEE Conference on Visual Analytics Science and Technology (VAST'12).* IEEE Computer Society, 293–294.

[23] Joshua D. Guttman and Amy L. Herzog. 2005. Rigorous automated network security management. *Int. J. Inf. Sec.* 4, 1–2 (2005), 29–48.

[24] Sunil Hazari. 2005. Perceptions of end-users on the requirements in personal firewall software: An exploratory study. *J. Organ. End User Comput.* 17, 3 (2005), 47–65.

[25] Xiangjian He, Thawatchai Chomsiri, Priyadarsi Nanda, and Zhiyuan Tan. 2014. Improving cloud network security using the tree-rule firewall. *Future Gen. Comput. Syst.* 30 (2014), 116–126.

[26] Almut Herzog and Nahid Shahmehri. 2007. Usability and security of personal firewalls. In *Proceedings of the Symposium on Edge Computing (SEC'07) (IFIP)*, Vol. 232. Springer, 37–48.

[27] Almut Herzog and Nahid Shahmehri. 2007. User help techniques for usable security. In *Proceedings of the Symposium on Computer Human Interaction for Management of Information Technology (CHIMIT'07).* ACM, 11.

[28] Hongxin Hu, Gail-Joon Ahn, and Ketan Kulkarni. 2012. Detecting and resolving firewall policy anomalies. *IEEE Trans. Depend. Secure Comput.* 9, 3 (2012), 318–331.

[29] Kenneth Ingham and Stephanie Forrest. 2002. *A History and Survey of Network Firewalls.* Technical Report, University of New Mexico.

[30] International Organization for Standardization. 1998. *ISO 9241-11: Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs): Part 11: Guidance on Usability.*

[31] Pooya Jaferian, David Botta, Fahimeh Raja, Kirstie Hawkey, and Konstantin Beznosov. 2008. Guidelines for designing IT security management tools. In *Proceedings of the 2nd ACM Symposium on Computer Human Interaction for Management of Information Technology (CHIMIT'08).* ACM, New York, NY, Article 7, 10 pages. DOI: http://dx.doi.org/10.1145/1477973.1477983

[32] Audun Jøsang, Bander AlFayyadh, Tyrone Grandison, Mohammed AlZomai, and Judith McNamara. 2007. Security usability principles for vulnerability analysis and risk assessment. In *Proceedings of the 23rd Annual Computer Security Applications Conference (ACSAC'07).* 269–278. DOI: http://dx.doi.org/10.1109/ACSAC.2007.14

[33] Bilal Khan, Muhammad Khurram Khan, Maqsood Mahmud, and Khaled S. Alghathbar. 2010. Security analysis of firewall rule sets in computer networks. In *Proceedings of the 2010 4th International Conference on Emerging Security Information Systems and Technologies (SECURWARE'10).* IEEE, 51–56.

[34] Ui-Hyong Kim, Jung-Min Kang, Jae-Sung Lee, and Hyong-Shik Kim. 2012. Practical firewall policy inspection using anomaly detection and its visualization. In *Proceedings of the International Conference on IT Convergence and Security 2011.* Springer, 629–639.

[35] Barbara Kitchenham and Pearl Brereton. 2013. A systematic review of systematic review process research in software engineering. *Info. Softw. Technol.* 55, 12 (2013), 2049–2075. DOI: http://dx.doi.org/10.1016/j.infsof.2013.07.010

[36] Anita Komlodi, Penny Rheingans, Utkarsha Ayachit, John R. Goodall, and Amit Joshi. 2005. A user-centered look at glyph-based security visualization. In *Proceedings of the IEEE Workshop on Visualization for Computer Security (VizSEC'05).* IEEE, 21–28.

[37] Nanda Kumar, Kannan Mohan, and Richard D. Holowczak. 2008. Locking the door but leaving the computer vulnerable: Factors inhibiting home users' adoption of software firewalls. *Decis. Supp. Syst.* 46, 1 (2008), 254–264.

[38] Christopher P. Lee, Jason Trost, Nicholas Gibbs, Raheem A. Beyah, and John A. Copeland. 2005. Visual firewall: Real-time network security monitor. In *Proceedings of IEEE Workshop on Visualization for Computer (VizSEC'05).* IEEE Computer Society, 16.

[39] Terje Nesbakken Lillegraven and Arnt Christian Wolden. 2010. *Design of a Bayesian Recommender System for Tourists Presenting a Solution to the Cold-Start User Problem.* Master's thesis. Institutt for datateknikk og informasjonsvitenskap (IDI-NTNU).

[40] Muhammad Mahmoud, Sonia Chiasson, and Ashraf Matrawy. 2012. Does context influence responses to firewall warnings? In *Proceedings of the eCrime Researchers Summit.* IEEE, 1–10.

[41] Florian Mansmann, Timo Göbel, and William Cheswick. 2012. Visual analysis of complex firewall configurations. In *Proceedings of the 9th International Symposium on Visualization for Cyber Security.* ACM, 1–8.

[42] Raffael Marty. 2009. *Applied Security Visualization.* Addison-Wesley Upper Saddle River.

[43] J. Mogul, R. Rashid, and M. Accetta. 1987. The packet filter: An efficient mechanism for user-level network code. In *Proceedings of the 11th ACM Symposium on Operating Systems Principles (SOSP'87).* ACM, 39–51.

[44] Shaun P. Morrissey, Georges Grinstein, et al. 2010. Developing multidimensional firewall configuration visualizations. In *Proceedings of the 2010 International Conference on Information Security and Privacy (ISP'10).* ISRST, 62–69.

[45] J. Nielsen. 1994. *Usability Engineering.* Morgan Kaufmann.

[46] Jakob Nielsen. 2010. Mental models. Nielsen Norman Group.

[47] Wes Noonan and Ido Dubrawsky. 2006. *Firewall Fundamentals.* Pearson Education.

[48] Fahimeh Raja, Kirstie Hawkey, and Konstantin Beznosov. 2009. Revealing hidden context: Improving mental models of personal firewall users. In *Proceedings of the Symposium on Usable Privacy and Security (SOUPS'09) (ACM International Conference Proceeding Series).* ACM.

[49] Fahimeh Raja, Kirstie Hawkey, Steven Hsu, Kai-Le Clement Wang, and Konstantin Beznosov. 2011. A brick wall, a locked door, and a bandit: A physical security metaphor for firewall warnings. In *Proceedings of the 7th Symposium on Usable Privacy and Security.* ACM, 1.

[50] Fahimeh Raja, Kirstie Hawkey, Pooya Jaferian, Konstantin Beznosov, and Kellogg S. Booth. 2010. It's too complicated, so i turned it off!: Expectations, perceptions, and misconceptions of personal firewalls. In *Proceedings of the 3rd ACM Workshop on Assurable and Usable Security Configuration.* ACM, 53–62.

[51] Marcus J. Ranum. 1992. A network firewall. In *Proceedings of the World Conference on System Administration and Security.*

[52] Robert W. Reeder, Lujo Bauer, Lorrie Faith Cranor, Michael K. Reiter, Kelli Bacon, Keisha How, and Heather Strong. 2008. Expandable grids for visualizing and authoring computer security policies. In *Proceedings of the Conference on Computer-Human Interaction (CHI'08).* ACM, 1473–1482.

[53] Jennifer Rode, Carolina Johansson, Paul DiGioia, Kari Nies, David H. Nguyen, Jie Ren, Paul Dourish, David Redmiles, et al. 2006. Seeing further: Extending visualization as a basis for usable security. In *Proceedings of the 2nd Symposium on Usable Privacy and Security.* ACM, 145–155.

[54] Aviel D. Rubin, Daniel Geer, and Marcus J. Ranum. 1997. *Web Security Sourcebook.* John Wiley & Sons, Inc.

[55] J. H. Saltzer and M. D. Schroeder. 1975. The protection of information in computer systems. *Proc. IEEE* 63, 9 (Sept 1975), 1278–1308. DOI: http://dx.doi.org/10.1109/PROC.1975.9939

[56] M. Angela Sasse and Matthew Smith. 2016. The security-usability tradeoff myth [guest editors' introduction]. *IEEE Secur. Priv.* 14, 5 (2016), 11–13.

[57] Karen Scarfone and Paul Hoffman. 2009. *Guidelines on Firewalls and Firewall Policy.* Technical Report. National Institute of Standards and Technology (NIST). Retrieved from http://csrc.nist.gov/publications/nistpubs/800-41-Rev1/sp800-41-rev1.pdf.

[58] B. Shneiderman and C. Plaisant. 2005. *Designing the User Interface: Strategies for Effective Human-computer Interaction.* Pearson/Addison Wesley.

[59] Tung Tran, Ehab S. Al-Shaer, and Raouf Boutaba. 2007. PolicyVis: Firewall security policy visualization and inspection. In *Proceedings of the Conference on Large Installation System Administration (LISA'07)*, Vol. 7. 1–16.

[60] Martijn Van Welie, Gerrit C. Van Der Veer, and Anton Eliëns. 1999. Breaking down usability. In *Proceedings of Interact'99.* 613–620.

[61] Kami Vaniea, Qun Ni, Lorrie Cranor, and Elisa Bertino. 2008. Access control policy analysis and visualization tools for security professionals. In *Proceedings of the Symposium on Usable Privacy and Security (SOUPS'08) Workshop (USM).*

[62] Artem Voronkov, Stefan Lindskog, and Leonardo A. Martucci. 2015. Challenges in managing firewalls. In *Proceedings of the Nordic Conference on Secure IT (NordSec'15) (Lecture Notes in Computer Science)*, Vol. 9417. Springer, 191–196.

[63] Justin Warner, David Musielewicz, G. Parks Masters, Taylor Verett, Robert Winchester, and Steven Fulton. 2010. Network firewall visualization in the classroom. *J. Comput. Sci. Colleges* 26, 2 (2010), 88–96.

[64] Alma Whitten and J. D. Tygar. 1998. *Usability of Security: A Case Study.* Technical Report. DTIC Document.

[65] Stephan Windmüller. 2013. Simplifying firewall setups by using offline validation. *J. Integr. Design Process Sci.* 17, 3 (2013), 59–69.

[66] Tina Wong. 2008. On the usability of firewall configuration. In *Proceedings of the Symposium on Usable Privacy and Security (SOUPS'08) (Workshop on Usable IT Security Management (USM'08))*. http://cups.cs.cmu.edu/soups/2008/USM/wong.pdf.

[67] A. Wool. 2004. A quantitative study of firewall configuration errors. *Computer* 37, 6 (June 2004), 62–67. DOI: http://dx.doi.org/10.1109/MC.2004.2

[68] Avishai Wool. 2010. Trends in firewall configuration errors: Measuring the holes in swiss cheese. *IEEE Internet Comput.* 14, 4 (2010), 58–65.

[69] Ka-Ping Yee. 2002. User interaction design for secure systems. In *Proceedings of the International Conference on Information and Communications Security*. Springer, 278–290.

[70] Lihua Yuan, Hao Chen, Jianning Mai, Chen-Nee Chuah, Zhendong Su, and Prasant Mohapatra. 2006. Fireman: A toolkit for firewall modeling and analysis. In *Proceedings of the 2006 IEEE Symposium on Security and Privacy*. IEEE, 15 pages.

[71] Bin Zhang, Ehab Al-Shaer, Radha Jagadeesan, James Riely, and Corin Pitcher. 2007. Specifications of a high-level conflict-free firewall policy language for multi-domain networks. In *Proceedings of the 12th ACM Symposium on Access Control Models and Technologies (SACMAT'07)*. ACM, New York, NY, 185–194.

[72] Chao and Chi-Shih. 2011. A flexible and feasible anomaly diagnosis system for internet firewall rules. *13th Asia-Pacific Network Operations and Management Symposium (APNOMS'11)*. IEEE, 1–8.

[73] Chao and Chi-Shih. 2007. A visualized internet firewall rule validation system. *Asia-Pacific Network Operations and Management Symposium*. Springer, 364–374.

[74] Hongxin Hu, Gail-Joon Ahn, and Ketan Kulkarni. 2010. FAME: A firewall anomaly management environment. In *Proceedings of the 3rd ACM Workshop on Assurable and Usable Security Configuration*. ACM, 17–26.

[75] Fahimeh Raja, Kirstie Hawkey, Konstantin Beznosov, and Kellogg S. Booth. 2010. Investigating an appropriate design for personal firewalls. *CHI Extended Abstracts on Human Factors in Computing Systems*. ACM, 4123–4128.

[76] Fahimeh Raja, Kirstie Hawkey, Steven Hsu, Kai-Le Wang, and Konstantin Beznosov. 2011. Promoting a physical security mental model for personal firewall warnings. *CHI Extended Abstracts on Human Factors in Computing Systems*. ACM, 1585–1590.

[77] Fahimeh Raja, Kirstie Hawkey, and Konstantin Beznosov. 2009. Towards improving mental models of personal firewall users. *CHI Extended Abstracts on Human Factors in Computing Systems*. ACM. 4633–4638.

[78] Shaun P. Morrissey and Georges Grinstein. 2009. Visualizing firewall configurations using created voids. *6th International Workshop on Visualization for Cyber Security, VizSec*. IEEE, 75–79.

[79] Stephan Windmuller. 2011. Offline Validation of Firewalls. In *Proceedings of the 2011 IEEE 34th Software Engineering Workshop (SEW'11)*. IEEE Computer Society, 36–41.

Received August 2016; revised July 2017; accepted August 2017