THE UNIVERSITY *of* EDINBURGH

Patching, System Administrators, and their Communities

Adam D.G. Jenkins

Doctor of Philosophy
Institute for Language, Cognition and Computation
School of Informatics
University of Edinburgh
2021

Abstract

"Update your devices", is well known security advice in both academia and industry. Yet there exists very little research into the process and the administrators tasked with sourcing, testing, applying, and troubleshooting of updates for computing systems, that serve many end-users. These system administrators (sysadmins) play a critical role in Information security management (ISM), with their decisions impacting the security of potentially millions of end-users. However, these decisions involve complex risk assessments on an update by update basis, as although patches can remove potential software vulnerabilities, they may also introduce new errors to systems that negatively impact their organization.

In this thesis I will present one of the first attempts at studying this user group and their impact on the patching process. To do so, I primarily focuses on sysadmins' Online Communities of Practice, which provides sysadmins with up-to-date patching information, such as known issues or related vulnerabilities. To begin with, I provide an in-depth qualitative artifact analysis of emails from a prominent patching orientated mailing list: PatchManagement.org. The analysis identifies several different types of information that is shared and requested by community members throughout their patching schedules. This information including requests for help troubleshooting patching errors or community generated lists of security patches to prioritise. I complement this work by constructing a descriptive case study, detailing distinct communities' collaborative information gathering and problem solving behaviours following the release of two security critical Microsoft patches. By detailing this online life cycle I find that these communities provide sysadmins with a dynamic, centralised source for their patching information, and that these communities share information often sourced from the work of other communities and their respective members. To conclude, I provide a survey of sysadmins detailing the prominence of patching behaviours at each stage of the patching process, and balance out the previous observational works with self-reported data from sysadmins from these online communities.

This work is one of the first explorations into the types of information system administrators share online with each other during patching, as well as the challenges they face and solutions they are using, such as forming these Communities of Practice. Patching, although on the surface appears very simple, is a more complicated task requiring a number of social-technical decisions to be considered before "just applying" the update. I present the lessons learned from our studies and indicate potential routes for future research within this space.

Acknowledgements

There are a number of people that I wish to thank for their help and guidance through out this PhD thesis. I may not explicitly state their name but everyone's efforts are all appreciated and I feel indebted to you.

To my family, thanks for the support, the jokes, and constant pressure to finish. I love you all.

To my lab mates of Tulips, thank you for the engaging conversations that helped me grow as a researcher and develop my ideas for this thesis. I have made friends for life. Thanks to Mohammad, Florian, Sara, Dilara, and Kholoud. I would also like to thank Helen, Nadin, Lachlan, and Dave. Thank you to Paul Anderson for his yearly feedback and guidance, and thank you to Duncan Reid for your insightful industry experience.

To my friends and collaborators who kept me motivated and jovial in hard times. Thank you to Linsun, Pieris, Manon, Nialls & Archie. Also thanks to Cult Coffee, for caffeine and laughs. Cheers to Rory and Taliah for keeping me sane.

A special word of thanks to my participants, and the moderators from PatchManagement.org. Their continual enthusiasm for my research let me know I was heading in the right direction!

I would like to thank both of my Viva examiners, Rick Wash and James Stewart, who came at my work at an angle I did not expect. I thoroughly enjoyed the experience and I felt that I leveled up as a researcher during the process!

To Maria, whose guidance is unparalleled. Your emotional intelligence and ability to know exactly what and when to say has continued to push me forward. I can't thank you enough for inspiring me to do research in HCI.

And last but certainly not least, to Kami, my supervisor. I cannot begin to express my appreciation for you and your guidance. You kept me motivated and on track through out this project, allowing me to guide it as I saw fit. Thank you for everything.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Adam D.G. Jenkins*)

To the unsung heroes of IT

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

System administrators work to manage the IT infrastructure of their organizations, keeping them functional, fit for purpose, and secure but doing so efficiently comes with many challenges. Administrators play a critical role in the day-to-day activities of many modern organizations and maintain the security for huge numbers of end-users and customers of organization systems. Sysadmins' security related decisions can have far reaching effects. Cybersecurity threats continue to change and evolve over time. One such threat is that of software vulnerabilities, which are flaws in software which allow malicious actors to perform unauthorized actions on system; during 2021 ethical hackers identified over 66,000 individual vulnerabilities, which was an increase of 21% on the previous year [1]. A number of techniques exist for remediation of vulnerabilities, but one solution remains constant: the rapid application of software updates. This advice, which on first inspection appears straightforward, is widely known by security practitioners within both academia [2] and industry [3], yet preventable security incidents continue to happen where a software update had been available for considerable amount of time before being exploited [4].

System administration can be complex, and resource intensive [5] causing business to invest increasing amounts in appropriate resources. In 2021, for example, 39% of business planned to increase their overall spending for IT personnel, and 53% intended to increase their IT staff numbers [6]. Administrators work within and across a number of distinct technology streams, such as databases, networks, websites, cloud services, and cybersecurity, all of which involve the updating of software as part of their professional duties [7]. Additionally, they work with a diverse range of software vendors, from large well established brands, such as Microsoft, to small individual vendors with their own range of products. To give a sense of scale, Microsoft re-

leased a total of 9,554 security updates between between January and June, 2021 [8]. These updates represented 275 unique Knowledge Base (KB) articles (update documentation) and 612 unique vulnerabilities for a range of Microsoft products, including Windows 7, which is no longer officially supported [9]. Considering that each update can require a sysadmin to conduct testing, mitigation, planning for downtime, reboots, etc. it is evident that the landscape that sysadmins must negotiate to remain secure is non-trivial. The above numbers are even more concerning if we consider that they ignore non-security related updates or even the numerous other non-Windows software vendors that also release their own updates.

The application of software updates to computing systems, also known as patching, is essential for maintaining security, however notorious examples exist of the negative impacts of failing to do so. For example, in 2017 the National Health Service (NHS) of the UK was attacked by malicious actors using vulnerability known as EternalBlue to gain access to a number of unpatched Windows systems, ultimately impacting at least 80 out of 236 Health Trusts in England, resulting in a total of 6,912 patient appointments being cancelled or re-arranged. This damage could have been avoided given that the patch that would have addressed the vulnerability had been available for two months prior to the incident [10]. This is not the sole example of missed patches having a massive impact on an organisation and its users: in the same year, Equifax, one of the world's largest consumer credit reporting agencies, suffered one of the largest ever data breaches. Private records of over 140 Million US citizens were compromised [11], even though a patch for the Apache Struts exploit used by attackers had been available for two months prior.

Why do sysadmins appear to be avoiding updates? Given that updates introduce fundamental changes to the inner workings of software, their application introduces unknown risk into complex non-homogeneous system. Hence, updates carry two distinct risks that a sysadmin must balance: the risk of not applying an update in sufficient time leaving systems vulnerable to attack, and applying an update too early before all potential errors have been found and addressed [12, 13]. For example, in early 2018, Microsoft's patches aimed at fixing the infamous vulnerabilities Spectre [14] and Meltdown [15], introduced errors causing PCs to become unbootable [16]. More recently, updates took down Microsoft's cloud based office services, including Teams, Outlook and Office, preventing users from authenticating and gaining access to the services world-wide [17].

As mentioned previously, the users tasked with the process of sourcing, prioritiz-

ing, testing, applying, and troubleshooting errors are known as System Administrators (i.e. Sysadmins). These technical users work in highly complex environments with their technical decisions having direct impact on systems and their end-users [18, 5]. In the case of Equifax above, the former Chief Executive Officer (CEO) stated that "human error" was to blame for the patch not being deployed [19]. This user group must work on a range of technologies [20, 21, 22, 23], yet there exits very little current research into their working practices and the issues they face with technological processes [24], including patching. This thesis, therefore, is one of the first attempts to investigate both the important security practice of patching, and the role, attitudes, and behaviours of sysadmins in relation to this process.

1.1 Research Overview

With Patch Management being an essential aspect of Information Technology Service Management (ITSM), and software vendors releasing patches that can both introduce new features and remove offending vulnerabilities, this research began with a simple question of, "What information is important for a sysadmin when deciding to patch?". Understanding the information available to sysadmins and how it was used, is necessary to better understand sysadmin behaviours around patching. Hence, the work in this thesis begins by looking into one of the online sources designed specifically for sysadmins and their patching needs.

1.1.1 "Anyone Else Seeing this Error?"

The first focus for my work is on the use of Online Communities of Practice (OCoP) by system administrators as an additional source of patching information for admins who participate in these communities. Communities of Practice [25] have previously been linked with the working processes of sysadmins [5], and this work extends upon these observations by detailing the construction and sharing of patching knowledge and practices. Communities of Practice is a social theory of learning, which at its simplest form is a group of people with interests focused around a single subject or practice, who regularly interact to share knowledge. This work was carried out using qualitative artifact analysis and codebook generation based on emails shared in a single, prominent OCoP, PatchManagement.org [26]. Through this detailed qualitative analysis I was able to answer the following research questions:

RQ1 In what ways do contributing members of PatchManagement.org engage with the mailing list?

RQ2 What types of information is shared and requested by sysadmins in the community?

Our results, detailed in Chapter 3, highlight the importance of these communities as socially collaborative systems which provide community members with a centralised source for useful patching information. Additionally, the work provides distinct thematic areas of information that admins need to conduct their patching tasks.

1.1.2 To Patch or Not to Patch?

The previous study indicated the scope of information available to admins, highlighting a community-wide effort to source and share relevant information for its members. Our previous analysis highlighted citation behaviour, where members would link to external sources of patching information, often accompanied with thanks for the author of the material. Hence, I decided to determine the scope of this linking of information by providing an in-depth descriptive case study of two security critical Microsoft patches. The case begins with the patches initial release on a "Patch Tuesday", to a final call from prominent community members giving the all clear to install those patches. I focused on the following two research questions:

RQ3 How does the knowledge of risks associated with a patch develop over time?

RQ4 How do the various communities support the collating and synthesizing of available information for sysadmins and other patching communities?

The results illustrate that these communities construct and share knowledge across different communities, with the detective work of one community providing crucial details pertinent to why a sysadmin may choose to avoid or delay applying patches. We highlight that these numerous Communities of Practice represent a wider collective, or Network of Practice [27], with information slowly propagating through to provide an overall picture of the patching landscape. The time that it takes for this information to propagate through the network to reach both admins and vendors can be lengthy, leading to logical justification of waiting before applying a new patch [13]

1.1.3 Deciding to Patch in Context

To compliment the previous chapters of work which are constructed using observational online data, I will provide the results of a survey of system administrators. The work detailed in the previous two chapters presents the information shared and how knowledge of risks surrounding patches can develop upon release to the wider community, yet we do not know if this information is considered by admins when patching. Additionally, we wish to understand and map out the context that they work with, since we know that their decisions will be contextual, driven by their unique system, the organization they represent, and the users of the organization. Hence, we wish to investigate the members of these online communities to understand the prevalence of behaviours identified in this thesis, and within other bodies of work [28, 29, 30]. This Chapter will answer the following research questions:

RQ5 What contexts do the system administrators studied work in? In particular: what types of systems do they administer, what size of organization do they work in, and finally what tools and resources do they have at their disposal?

RQ6 How often do sysadmins engage in the various commonly known patching behaviors that prior work has identified as being associated with the six patching stages?[28, 29]

RQ7 How does the context of a sysadmin (e.g. organization size, systems supported) impact their approaches to patching?

The results highlight the range and number of machines, operating systems, and applications that sysadmins must manage, including the management of End-of-Life (EOL) softwares. Furthermore, we identified differences that exist between sysadmin who manage updates for Small and Medium Enterprises (SMEs) compared to large organizations (250+ employees) and sysadmins' patching processes for servers compared to client machines. Finally, we are able to expand upon the testing practices of admins, with known techniques such as dedicated testing environments and staged deployment [28] appearing to be less prevalent than previously believed. The work indicates that sysadmins do not restrict information gathering from online sources and communities within the initial awareness stages of patching [28, 29], and in fact will continue to monitor sources for patching issues and troubleshooting of errors.

1.2 Research Contributions

This thesis consists of findings made from two distinct sources: the first is the use of observational online data and digital ethnographic methods of online communities, and the second uses self-reported data from sysadmins to validate behaviours identified from earlier observations. Combining these two streams of data I am able to provide one of the first detailed investigations into this user group, which is known to be difficult to research due to their idiosyncratic and complex working environments [31, 32].

The first study presented in this thesis provided the Usable Security community with an in-depth understanding of the role of mailing lists in sharing patching information. Previous work had identified that system administrators would search a number of sources to obtain their patching information [29, 28]. I show that the mailing list, PatchManagement.org, is an example of an Community of Practice [25], where community members have organically created a centralised source to share and request information regarding patches. I show that verification of a working patch is a complex task, with community members relying on the second-hand information of patch impact as indicators as to the safety of certain patches.

The second study provides a deeper understanding of the interactions between distinct communities, with the information used by one community often cited and referenced from the work of another community. The scaffolding provided by these Communities of Practice provides ample space for members to learn and engage with their peers, allowing for the evolution and sharing of policies and strategies which guide administrators daily patching workflows. I highlight that these communities work together to represent a wider Network of Practice [27], with distinct sub-communities with differing norms and standards.

In the final study presented, I provide the largest sample obtained when studying system administrators (N=220) [33, 34, 29, 28, 35], giving greater confidence in the results derived. This study asked questions regarding the importance of certain contextual factors including sector of organization, system being updated, and testing set-ups used by admins. The survey indicates that sysadmins will extend their information searching beyond initial stages of the patching process, and rely on information verification regarding the quality of released patches. End-users remain a useful resource in patch quality as well, with their reports being used to find troublesome patches.

1.3 Thesis Outline

The remaining chapters of this thesis are as follows: Chapter 2 will detail the relevant existing literature, covering System Administrators, Patch Management, and supplemented with end-users and updates. Chapter 3 provides an in-depth artifact analysis of emails found on the popular mailing list, PatchManagement.org, capturing the information requested and shared within a single prominent online patching community. Building on this, Chapter 4 will provide a descriptive case study detailing two security critical Microsoft patches and their online information life-cycle between distinct communities. Chapter 5 compliments the previous chapters by providing the findings from a survey concerning how system administrators self-reported beliefs regarding behaviours when patching for their organisations. Chapter 6 ends the thesis with a discussion of the implications of the findings, and identifies lessons and future directions for studying system administrators.

Chapter 2

Background

This chapter details the related literature that highlights the importance of Patch Management as well as the related work surrounding System Administrators and their working practices. It provides a detailed summary of research into their daily working practices, and the many technologies that they will work with to keep their systems and their users secure. Additionally, I supplement this literature with a review of work involving end-users and their experiences with software updates, highlighting similarities and differences between them and sysadmins.

2.1 Vulnerabilities and Patch Management

Organizations employ a number of distinct and complimentary security practices to maintain their systems' and users' security. One such practice is the proactive measures associated with Vulnerability Management (VM), where they actively detect and reduce or eliminate potential vulnerabilities from being exploited by malicious actors. A large component of this process is Patch Management (PM), involving the sourcing and application of Software Updates (i.e. Patches), which are aimed at addressing known vulnerabilities [36, 37].

Advice provided by both cybersecurity professionals and government guidance states that the ideal state is for a system to have all software updated to the latest versions [38, 2, 3], however studies have indicated that organizations are far from achieving this goal [39, 40, 41, 42]. An integral factor in the success of PM is "time to patch", since once a vulnerability has been publicly disclosed the volume of attacks using that vulnerability increases by five orders of magnitude. This increase happens because cyber criminals monitor for these patch announcements and use the information in them

to design new attacks [43]. Research has shown that the decision to delay patching can result in windows of vulnerability that can exist for many months, and even for years following their disclosure [40, 39].

Additionally, it has been shown that automating the deployment of patches may also not be sufficient due to the *social technical* aspects relevant to the decision to patch [39, 12, 42, 44]. For example, the systematic literature review of patch management literature by Dissanyake et al. [44] identified 14 social technical challenges to software security patch management, including the impact of organizational policies, the need for human expertise, and the difficulties in coordination and collaboration around patch deployments. These identified common challenges delay the application of patches and is detrimental to a system's security as it increases time to patch and leaves attack vectors open to exploit until it has been updated. However, the race to apply patches before they are actively exploited is countered by the opposing risk of applying updates to system that inadvertently introduce new "bugs" and hence impact business essential operations [12, 13, 44]. The work in this thesis is aimed at further understanding the patching process from the perspectives of the user group primarily tasked with PM: System administrators.

2.2 System Administrators and Working Practices

The day-to-day management of IT systems for many organization and customers is delivered by System Administrators (sysadmin), either organised into distinct teams with a specific service focus (i.e. Database Administrator) or by a sole administrator. The earliest attempt at understanding the complexities of system administration was by Hrebec and Striber [33], who developed a survey (n=54) based on initial interviews, to investigate sysadmins' mental models and situational awareness. The results indicated that sysadmins manage systems they do not fully understand, with an average reported understanding of around 77%. This result indicates the non-homogeneous and complex nature of modern IT infrastructure sysadmins are tasked with managing. Additionally, when asked how they fix problems they are unfamiliar with, responses included research and investigation of documentation (24%) and experimentation with the system (37%). Sysadmins also reported conducting research online with newsgroups (44%) and consulting personal networks and contacts to find relevant expertise (25%). The results highlighted the wide range of sources and approaches that sysadmins take when managing large complicated systems.

The majority of what is known regarding the working practices of sysadmins is detailed in a series of ethnographic studies [45, 46, 47, 48, 49, 18, 50, 51, 52, 53] conducted in the early 2000s, and later complied into a book, "Taming of Information: Lessons from Studies of System Administrators" [5]. The authors detail numerous lessons learned from their observations, including a reliance on developing and adapting best practices or tools to suit sysadmins idiosyncratic working contexts. Sysadmins would continually develop practices with varying lifespans (i.e. one-time use to an organizational-wide standard) and where they were unable to find a suitable tool or example, they would construct or adapt something suitable. The authors state that system administration is fundamentally a collaborative endeavour as modern IT system are constructed from numerous distinct systems managed between numerous teams of admins with different but complementary expertise and focus. For example, consider the case of "George", a web admin tasked with debugging issues with a new Web server [5][p19-49]. Throughout the example used, George must communicate across several technologies and their respective teams of sysadmins to collaborate and troubleshoot his server issues. He interacts with the network team to allocate an IP address for his server instance; with the firewall team to open ports to allow the new instance to communicate with a middleware server; and he must communicate with the software vendors who know the un-documented intricacies of their software. George collaborates with all these distinct groups and facilitates communication between these groups to form an overall understanding of the system and his issue.

Therefore, *grounding* [54, 55, 56] becomes a vital component of system administration as they aim to create the shared understanding that is necessary for efficient collaboration. Grounding is the process of establishing mutual knowledge, beliefs and assumptions between two people. The necessity of grounding is partially due to the distributed nature of IT systems and responsibilities within an organization, with IT Service Management being a *distributed cognitive* process, where numerous sysadmins, tools, and artefacts work in-conjunction to complete a given task [45, 57, 58]. For example, the work of Althobaiti et al. [59] identified the phishing incident response practices of a UK university to be a distributed cognitive process involving several teams, and tools working in conjunction to handle phishing reports and ongoing campaigns. The authors highlight the importance of the role of tools used, in this case a ticketing system, which facilitates the hand-offs between teams and indicates the necessary actions necessary from the respective teams. The impact of the ticketing system on the workflows of the respective teams resulted in unintentional losses of

pertinent data, indicating the influential role tools can play in the practice of system administration.

Velasquez et al. [60] complemented previous work by examining the features of tools that sysadmins found important. Given the complex and high-risk environment of sysadmins work, their results indicate a strong focus on the accuracy, verification, reliability, and credibility of tools and information they produce. Fundamentally, this results in sysadmin tools requiring an alternative design approach to the tools and software used by end-users [7], for example the authors highlight the need for tool "flexibility" and "scalability" to suit the working practices of sysadmins who will work with their own unique systems which continually change and represent a diverse range of components. Additionally, Velasquez and Durckiova [61] extended on their observations, by identifying the positive relationship between task complexity and their need for verification information. Essentially, sysadmins are more likely to engage in information seeking behaviours when confronted with a complex technical issue, taking in information and translating it to their contexts, and validating that actions performed have had the intended result.

This common practice of information gathering and adaption to their contexts has led the role of sysadmins to be described as "*Broker Technicians*" [62], where they act as a translator between the users of the infrastructure in their organization, and the developers of the technology in the wider technical community. Highlighting the importance of these information sources to sysadmins and their day-to-day work, as users raise requirements, which must be translated into a technical query to the wider community, and then any response and action must again be translated to suit their organization and stakeholders.

2.2.1 Sysadmins and Security

Given the importance of sysadmins and their role for their organization and its end-users, a number of studies have been conducted to evaluate their impact on security. For example, sysadmins can influence their system and its users security through configuration errors [20, 21, 63], and with the implementation of SSL [22], HTTPS [23, 64], and firewalls [65].

Security orientated sysadmins and their work practices differ from those in a non-security context. This is often related to the nature of cybersecurity, requiring a steeper learning curve and a more reactive approach to events that are high risk and complex,

resulting in a stronger focus on collaboration [18, 53].

Kraemer and Carayon [66], interviewed 16 network and security administrators to explore their perspectives on human error within security and found that organizational factors were the most frequently cited cause. Furthermore, they found that network admins were more likely to interpret end-user errors as intentional acts, highlighting the convoluted relationship between users and their admins.

2.2.1.1 Sysadmins and Updates

There exists a few scattered examples of sysadmins and patching in the previously discussed literature [5, 7], however patching was not a direct focus. Shostack detailed the balancing act that is at the heart of patching decisions made by sysadmins, as they weigh out and compare the potential risks associated with installing a patch to a business essential systems against the security risks of not installing and leaving a vulnerability with the system [12]. This work was extended upon by Beattie et al. [13], who developed a model to identify the optimal time for an update to be applied. Their model found that the optimal times for application of a patch fell on 10 and 30 days following the initial release of a patch. This timing is due to the fact that upon release patches may contain errors, and it is therefore prudent behaviour to wait while these errors are discovered and remedied through hotfixes before applying to systems. Additionally, by waiting sysadmins may be able to tap into their personal networks and collect further data on the impact the patch is having on other systems. In related work, Cameri et al. surveyed 50 sysadmins as part of their work developing Mirage, a distributed framework for deploying patches [35]. Their results detailed that patching is a regular occurrence with 90% of respondents handling patches at least once a month. Furthermore, 70% reported delaying the installation of an update and the average failure rate of patches was reported to be 8.6%.

More recently we have seen research that has focused solely on sysadmins and patch management, with Li et al. [28] providing a survey (n=102) and semi-structured interviews to identify the typical actions performed by sysadmins during the patching process. From their results they were able to identify 5 stages of patching, and their impact on patching effectiveness. For example, they observed that due to a lack of a single centralised hub of patching information, sysadmins were forced to search a number of different sources to gather information on patches. The survey responses included; official vendor notifications (71%), security advisories (78%), professional mailing lists (54%), online forums (53%), news (39%), and blogs (38%). When asked

how they would handle patches which cause errors, just under half of respondents (47%) reported they would uninstall the update, which may leave their systems open to compromise. This work was extended upon by Tiefenau et al. [29], who provided very similar results with a smaller survey (n=67) of a predominantly European sample of sysadmins. They identified a number of additional obstacles including the scheduling of necessary system downtime for patch installation (88%). An interesting observation from their results was that 55% of respondents reported that post-deployment errors from updates were a minor concern, with only 8 participants strongly disagreeing with this statement.

Martius and Tiefenau [30] complimented the previous work by focusing on the information sysadmins gathered to inform their patching decisions. To do so they compiled information that was contained within patch release notes and constructed two surveys (n=41 & n=16) of sysadmins to identify what information was regarded as important. The results showed that 68% of sysadmins found the lack of patch information made the task of patching more difficult, and that *Known Issues* regarding a patch were highly valued. Additionally, the authors noted the lack of standards regarding patch release note contents.

2.3 End-Users and Updates

The need to patch systems quickly is a well known and frequently shared piece of advice among security experts [2], however non-technical users fail to realise the security implications of failing to apply an update [38, 67], with many studies highlighting that end-users regularly delay or entirely avoid updates [68, 69, 70].

With the security benefits of patches apparently elusive to users, further investigations have been conducted to identify reasons for avoiding updates. Numerous pain points of updating have been identified, such as the updating taking time and work-flow interruptions through forced restarts of systems [71, 69, 72, 73]. Additional factors include changes the update introduces such as UI changes [74, 75, 69, 72, 76] or privacy concerns [77]. The negative user experiences caused by updates create distrust with updates, influencing future updating decisions [73, 71, 74].

Attempts to increase patching rates have been proposed such as the use of automatic and silent updates, which have been shown to improve security [78, 70, 41, 79]. However, automatic solutions do have drawbacks in that they negatively impact the mental models of end-users and their systems leading to unintentional security conse-

quences as shown by Wash et al [72]. Using a mixed-methods approach by collecting computer log data, survey responses and interviews with 37 Windows 7 users, the authors were able to identify that through automating updates and removing users from that decision process resulted in unintended side-effect of users not fully understanding what their computers were doing. In some instances this resulted in some users unwittingly extending their window of vulnerability and conflicted with their beliefs regarding the systems patching status. Automation can play a role in the patching process, but it has been argued that automation can not fully replace the role of the sysadmin, with many IT systems requiring expert human interventions [5].

Further attempts at user-centred solutions include the design of update notifications [80, 81]. Fagan et al. [80] provided two surveys of users (N1 = 78, N2 = 172) to identify beliefs and attitudes regarding software update notifications, finding that many notifications failed to alert users to the privacy and security related consequences of specific updates. Fagan et al. [81] extended upon this by surveying 155 users to further identify attitudes and beliefs regarding 13 real world software update messages. The authors identified features from multiple designs that were highly correlated to messages being considered confusing, annoying, noticeable, and important. Marthur et al. [82] attempted to design and evaluate a prototype updating interface which provided users with information such as the update's size, or time required to install. They trialed their prototype with 22 users and produced four recommendations for future interface designs, such as allowing users to personalize their interfaces and to centralise update management to a single update manager as opposed to a range of distinct services. Attempts to create novel update notification interfaces include Sankarpandian et al. [83] and their TALC system. TALC would alert users to the existence of new updates by adding graffiti to the desktop backgrounds of users and encouraged users to clean their desktops by applying the necessary update. Their small user study (N=10) showed that the TALC system was significantly effective in raising users' awareness and motivation to install patches. Tian et al. [84] provided users with user generated reviews regarding an application updates and their permission requests to allow for more privacy conscious decisions from users, finding that their inclusion were more effective in raising awareness. Additionally, Frik et al. [85] investigated whether the use of 'commitment devices', a technique from behavioural economics [86], to allow users to schedule their updates or use reminder options. Using two online experiments, with over 1,000 participants, they found that the use of commitment notifications increased users intentions to install updates, suggesting that current update interfaces are

too restrictive when offering user suitable times to update.

2.4 Summary and Gaps

The literature on sysadmins and both their work practices and task related to cyber-security continues to grow and develop. Sysadmins are clearly very important in the security posture of both the organization they work for, and the end-users they support. However, with regards to software updates and patch management, little is still understood. This is especially the case around their use of "social" information sources, such as forums and mailing lists.

Previous work [5, 61], has explicitly highlighted their reliance on external peers and communities, further justifying the need to develop the Usable Security community's investigations within these online spaces.Developing our understanding of such online sources and peer groups will help understand how security advice and information within and beyond patching is generated and consumed by sysadmins. Additionally, considering that end-users themselves have reported a need for seeking advice from others, one can argue the need to investigate this relationship further within the sysadmins' context and to draw out further investigations of the essential nature of their "broker technician " [62] role.

Chapter 3

"Anyone Else Seeing this Error?": Community, System Administrators, and Patch Information[1]

3.1 Introduction

System administrators are known to engage in collaboration on complex tasks [5, 45] and information gathering [61] around system maintenance or unfamiliar problems, but little is known regarding how this tendency translates to online spaces and the high-risk activities such as patching.

Sysadmins work in inherently complex environments, both technical and social [5], with modern IT management within organizations often split across a number of distinct services, with their own teams with specific and specialised knowledge [59, 88, 45]. Sysadmins must therefore collaborate across systems when troubleshooting errors which incorporate a number of services [5][p.19-49], and work closely with the respective team members to find a solution that works across the board. Additionally, these collaborations may extend beyond one-off incidents, with sysadmins' communities providing admins guidance on the critical tools and knowledge necessary for their daily working practices, with the focus on collaboration intensified when discussing security related tasks [18].

[1]This chapter was published in IEEE European Security and Privacy in September 2020 [87], and it was a collaboration with Pieris Kalligeros and my supervisors, Kami Vaniea and Maria Wolters. I was the lead author of the paper, with the second author being a second coder for the codebook creation, to ensure an objective and reduce bias when analysing and reporting of findings. I wrote the first draft and collaboratively edited it with the co-authors.

Patch management is a difficult task, requiring sysadmins to have a strong understanding of their system components, obtain good information about potential patches, and apply them to the systems without incident. The task is also difficult because of the unknowns around how a system will react to a patch.

The Meltdown [15] and Spectre [14] patches mentioned previously, are a good example of the security importance related to updates, as well as the uncertainty as to the outcome of patching. These vulnerabilities affected all modern processors, allowing malicious programs to access and read information stored in memory, including memory used by other programs running on the same processor. Large vendors, such as Google, Apple, Microsoft, and IBM quickly released patches to protect users. The following week, Microsoft was forced to pull its patches due to users complaining about unbootable PC's [16]. Hoax websites also started appearing offering malware laden Meltdown/Spectre patches [89].

Patching is crucial to security, the majority of compromises involve an exploited vulnerability for which a patch exists, but has not yet been installed [90, 91]. From a purely security viewpoint, installing patches quickly is vitally important. Official patch advice, such as the UK Government's "Cyber Essentials" scheme recommends that patches be applied within two weeks of release [3]. Yet, in the wild we still do not see critical patches being as widely installed as they should be. For example, the WannaCry malicious software took several UK National Health Service trusts offline blocking patients from accessing health care. The associated patch had been available for a full two months before the attack, but had not been installed [92]. Similarly, HeartBleed – a serious vulnerability that provides anyone with a dump of current server memory – was still an issue three years after the patch release as a third of systems had still not applied the patch [93]. Recent work shows that this situation has not improved [94].

Despite the security benefits of patch installation, sysadmins may still be hesitant when installing them. Patches have a long history of coming not only with security improvements, but also with problems [13, 12]. Code Red, a computer worm from the early 2000's, famously featured a faulty patch solution, followed by a second less faulty patch which most people installed, and finally a third good patch that few installed because the second one mostly worked [42]. One of the roles of a sysadmins is to balance the security needs of patching quickly against the risks of installing a problematic patch, which can be challenging [43].

Our work aims to expand on the research community's understanding of sysad-

mins' information seeking behaviors around patch management by conducting a qualitative analysis of the prominent mailing list: PatchManagement.org. Recent work has indicated that similar information sources are being used by sysadmins during the initial stages of assessing a patch [28, 29], and we present the first analysis of such a data source. This mailing list is dominated by Microsoft related content, as seen in Figure 3.1, so we provide a contextual focus to the information shared on the list by narrowing our focus to that of Windows Update.

In this work, we qualitatively code 356 list emails sent between March and July, 2018, to understand how the list community interacts. Based on our results, we argue that the mailing list is an example of an Online Community of Practice, where practitioners engage in communal learning and support.

Our primary questions when analyzing the list were:

RQ1 In what ways do contributing members of PatchManagement.org engage with the list?

RQ2 What types of information is shared or requested by sysadmins in the community?

We find that the PatchManagement.org community supports members in multiple phases of the patching process from providing guidance on pre-patch mitigation, information to support patch prioritization, suggesting workarounds for error-prone patches, and sourcing information on potential errors. We detail the community effort in contributing advice on tool selection and facilitating discussions of best practice approaches to patch management.

3.2 PatchManagement.org

PatchManagement.org [26], has been in existence since December 2003, and is the first mailing list dedicated to the topic of patch management. It was designed to be used by network and sysadmins along with security professionals to discuss the latest events and information related to patching, with no restriction on vendor or operating system. It encourages discussion of a range of topics including experiences with released patches, and "how-to" questions regarding deployment or assessment of patches. The list's charter does have some restrictions on post content. Most notably for this research, is the restriction of announcements regarding vulnerabilities, unless they are accompanied with a mitigation.

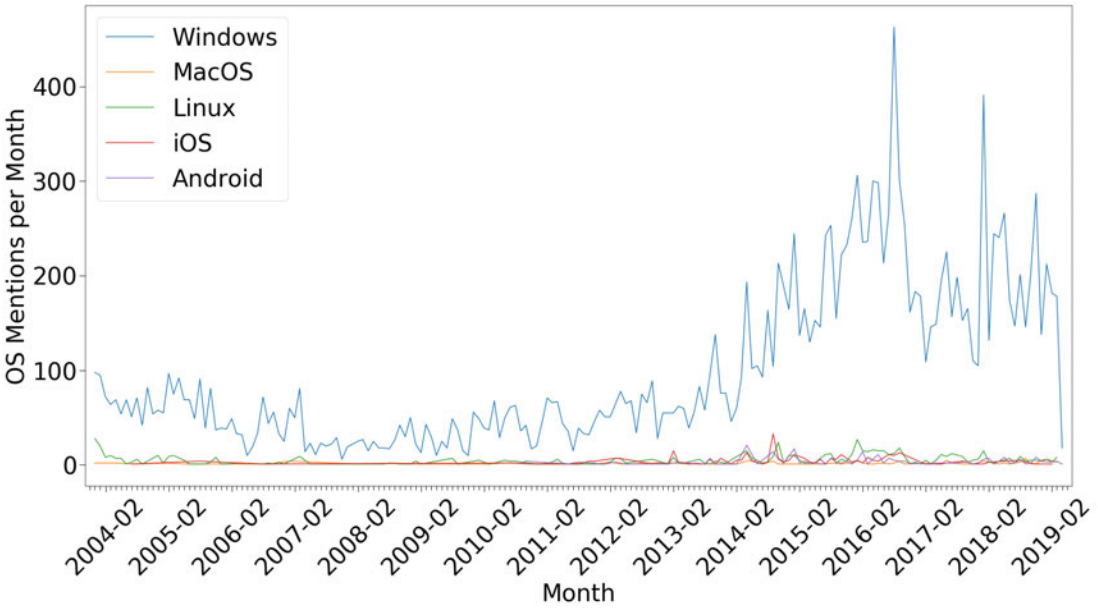

Figure 3.1: Number of times each OS was mentioned in list emails per month.

Moderation is provided by several key members of the patching community, from whom we have received consent to conduct community-respectful research using the mailing archive. Furthermore, research we have produced is being shared with them to allow for feedback and validation. When we did provide this paper for feedback to the community we saw a number of positive confirmations regarding the themes that we discovered, but also in the importance of such social communities for their working practice and task success in not only patching but related system administration duties. The list is hosted by Ivanti, and the emails posted through the mailing list are stored on MARCive online [95]. In April 2019, after data collected for this study was completed, the list re-located to a Google Group.

3.3 Windows Update and Patch Tuesday

A large percentage of the PatchManagement.org emails focus on Microsoft-related patches (Figure 3.1) which are released in monthly cycles. This focus on Microsoft makes sense as it is still the most popular Operating System (OS), running on around 88.80% of all desktop platforms [96]. We give an overview of Microsoft’s patching process to help the reader understand the context the email list exists within.

Microsoft’s patching process has been through many variations. Prior to Windows 98, patches were only posted on a website where users could download them,

but no Microsoft-produced automated process existed for downloading and installing them. With the release of Windows 98, the Critical Update Notification Utility [97] was added as a background process that would automatically check the website for new "Critical" patches and notify the user if they were found. In 2000, Windows ME shipped with "Automatic Updates", a tool that automatically checked for patches, but only once the user had opted-in for the feature. Windows XP SP2 changed the default settings to automatically download and install patches with users having the right to opt-out [72]. As a result of the defaults change, Microsoft saw Windows XP patching rates jump from 5% to 90% [78].

On the administration side, before Windows XP and 2000, there was no easy way to centrally deploy or manage patching. Then, around 2003, Microsoft released the Software Update Service (SUS) for Windows XP and 2000 [98].

SUS provided control and centralisation by allowing administrators to run their own local "Windows Update" servers which allowed them to select which patches should and should not be deployed in their organisation. The Automatic Update software on the organization's computers would then point at the local update server and automatically install the sysadmin's selected patches. SUS was a powerful tool, but it still lacked many crucial features, such as varying patching platform (i.e. servers or desktop), which would not arrive until the release of SUS's successor Windows Server Update Service (WSUS) in late March 2005.

Additionally, WSUS allowed sysadmins to implement staged deployment of patches for testing purposes before a full deployment.

Starting October 2003, all Microsoft product updates are released on the second Tuesday of every month, unofficially dubbed "Patch Tuesday". On Patch Tuesday all available patches are released at once between the hours of 17:00 and 18:00 UTC, with related information such as Knowledge Base (KB) articles following soon after. The KB articles are important because they provide a unique number for the patch as well as documentation such as the update's purpose, related system or platform, known issues, and workarounds.

Historically, the second Tuesday was chosen as it was theorized that it would allow sysadmins the full Monday to fix any issues found in their system over the weekend, before having to manage the updating process. The scale and dominance of Patch Tuesday has resulted in some companies piggybacking off of the model, with Adobe also releasing their patches on the same day. Attackers have also normalized to the cycle, with the next day now unofficially called "Exploit Wednesday." They also download

and analyze the patches, find changes the patch makes (vulnerability locations), and then develop exploits to target unpatched platforms.

Prior to Windows 10, patches were released separately for each issue and sysadmins could select which patches to install or not. With the release of Windows 10 Microsoft switched to a cumulative updating model for all Windows versions where each version of Windows theoretically receives only two patches each month: the cumulative patch and the security-only patch. Installing the cumulative patch installs all outstanding updates for the system for the current month and prior months. Installing the security-only update will only patch security issues for the current month. Cumulative updates are potentially problematic for security because they force sysadmins into an all-or-nothing situation. Previously they could selectively block patches that were having issues while installing the others, but with the cumulative model, if any of the changes cause problems, the whole patch must be blocked and they have no option of installing non-problematic elements.

3.3.1 List Data Collection

We scraped the PatchManagement.org mailing list archive in April of 2019 using a custom Python script to automatically download the emails from MARCive and stored them and accompanying metadata in a secure PostGreSQL database. We used the Beautiful Soup library [99], to strip HTML and extract the plain text. The study design was certified by the School of Informatics’ ethics panel (#84622). Sender emails were also processed to ensure that multiple presentations of the same sender were linked together.

Host information was then extracted from email addresses and used to determine the likely country of origin and the sector using FortiGuard’s Web Filter [100]. Out of respect for the list community, we avoid any discussion of specific organizations, and instead focus on sector and region of the world to describe members.

3.3.2 List Demographics

We collected a total of 63,536 emails with send dates ranging from December 2003 to April 2019. Figure 3.2 shows the total numbers of emails sent per month for the whole dataset, and Figure 3.3 shows emails sent only in 2018. Both figures show the wide variation of month-to-month email counts. Figure 3.2 also illustrates how the list has shrunk and then grown over time, with the recent up-tick in emails roughly

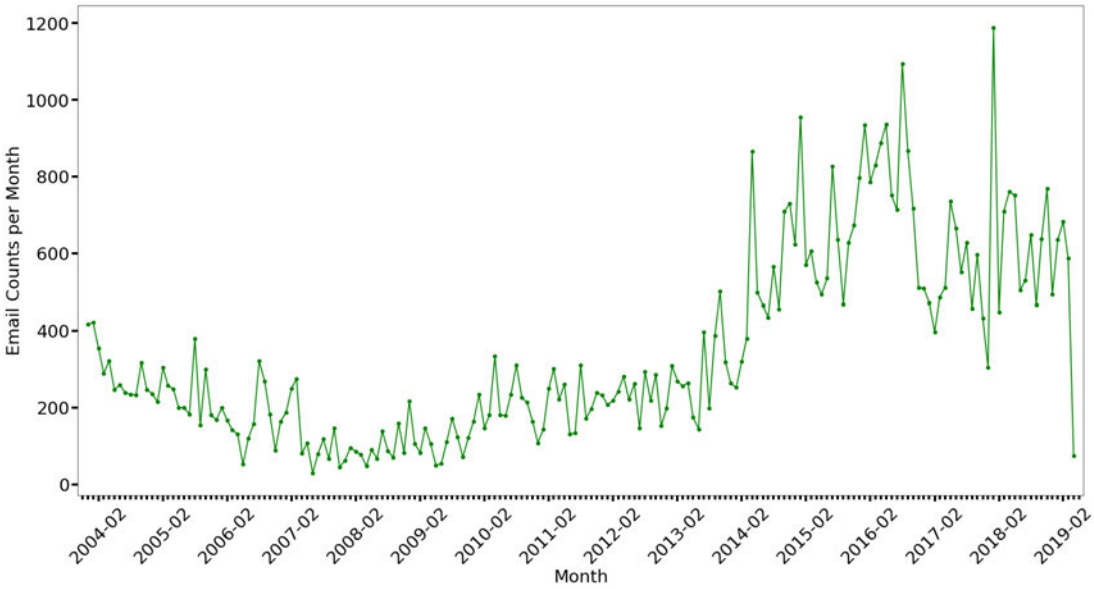

Figure 3.2: Total number of emails sent per month for the whole history of the list.

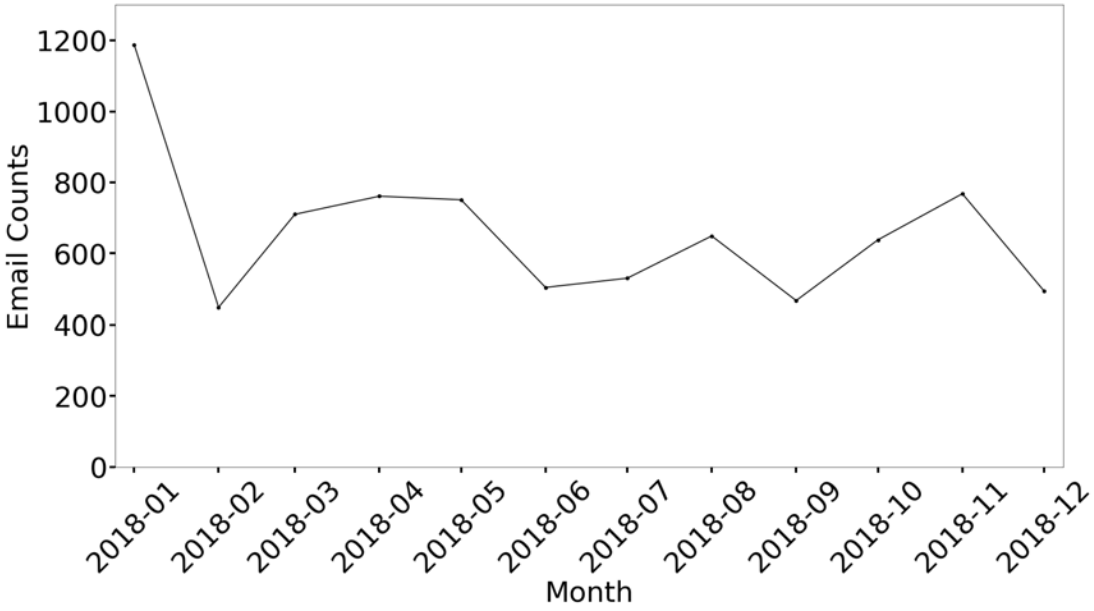

Figure 3.3: Total number of emails sent per month in 2018.

Table 3.1: Comparison between 2018 and other years (04-18) in terms of the number of *emails* sent across the list, the number of unique *senders*, and the number of emails sent that had *no reply*.

# of	**2018**	**All Years**			
		Avg.	Std. Dev.	Max	Min
Emails	7908	4075.93	2718.25	9538	1217
Senders	870	670.60	201.51	1200	476
No Reply	538	342.67	153.82	620	178

corresponding to the increased focus on patching as a preventative security practice as well as the industry-wide shift towards making patching automated [38, 75]. Taking a closer look at the most recent complete year, 2018, Table 3.1 and Figure 3.2 show that the number of overall emails are increasing, with a slightly higher number of total emails, senders, and email threads with no replies. There is a drop-off at the very end, but this is believed to be due to the fact that we scrapped the data-set midway through the final month of April.

We searched the emails for Operating System names (Windows, MacOS, Linux, iOS, and Android) to understand their relative representation. Windows is by far the most discussed operating system (Figure 3.1).

To understand the variety of organizations represented, we looked at the host names of all 870 unique email senders in 2018 along with their sector according to FortiGuard. We identified a total of 670 unique email domains, 94% of which were associated with only one email address. In other words, most organizations represented have only a single sysadmin engaged with this community. We found a total of 24 unique country top level domains from the UK, US, Canada, Australia, New Zealand, Europe, Asia, and South America. The top three sectors by sender were: Business (n=191), Education (n=119), and IT (n=110). Other notable sectors included: Government and Legal Organisations (n=59), Health and Well-being (n=62), Financial (n=60), and Private Domains (n=165) which include email addresses such as @gmail.com and @hotmail.com. In short, the list represents sysadmins located world wide in a range of sectors.

3.4 Methodology

3.4.1 Selection of PatchManagement.org

My initial work with the PatchManagement.org [26] mailing list began at the early stages of my PhD. Due to the earlier stated 'seed' question over the apparent lack of enthusiasm for patching within the sysadmin community, my supervisor had initially done some research into valuable resources to investigate this phenomena. Due to the difficulty in gaining sysadmin participants and security workers in general [31, 32], we chose to look closely at the online sources that appeared relevant to our needs. We took inspiration from early work on developers and Open Source Software (OSS) mailing lists [101] and the ongoing discussions with in the HCI community regarding the continual use of such mailing lists despite new alternatives (Forums, Social Media, etc) now existing [102]. We learned of the Mailing list ARChive (MARCive) [95] through discussions with other researchers and used that to find a relevant patching mailing list. Upon finding the Patch Management mailing list, we contacted the moderation team directly to gain consent to use the publicly accessible mailing list to begin our exploratory analysis.

3.4.2 Codebook Design

Before creating the codebook, the first and second coders who conducted the qualitative analysis reviewed the full content of 13 randomly selected threads from 2018, comprising 106 emails in total. Based on this review, we decided that similar to other work [103], the analysis would focus only on the initial email of a thread. The content of the full thread was interesting, but provided minimal additional information about the reasons people engaged with the list.

To create the codebook, the two coders open coded [104, p.100] the same 50 randomly selected initial thread emails from throughout 2018 and individually created potential codebook structures. They then met, discussed the observed themes and created a combined codebook. One interesting observation was the difference between emails where the primary goal was information seeking versus sharing. So the codebook explicitly had separate codes based on the perceived intention of the email sender. Both coders then individually applied the codebook to a new random set of 50 emails, this time writing memos as they went. Both the memos and results were reviewed and discussed resulting in a refinement of the codebook. Coding conflicts were resolved

within these sessions through continual discussion with issues reviewed until agreement was reached. This process was repeated six more times (350 emails in total) before the coders felt they had sufficient agreement and were no longer seeing new themes emerge (saturation). Inter-rater agreement as measured by Cohen's Kappa was 0.88 (2 s.f.) for the last set of emails, which is considered to be a very high level of agreement [105]. Table 3.2 details the codes present in the codebook along with their descriptions.

3.4.3 Qualitative Coding

The codebook was applied to all emails sent from the beginning of March to the end of July 2018. The dates were chosen for three reasons. First, they are from the most recent full year, 2018. Second, Meltdown and Spectre, headline worthy vulnerabilities, were announced in January 2018, so we selected a start date of two months later to allow the list to return to a normal state. Third, the chosen dates avoid major holidays that might impact patch behavior. A total of 380 initial thread emails were sent within the chosen five months, representing 0.05% of all emails in 2018. The emails were then divided up with 45% given to each coder. The remaining 10% were given to both coders and placed at the end of their coding list. This 10% (n=38) overlap was used to measure the drift as the two coders coded separately. The resulting κ=0.79 (2 s.f.) shows a continued high-moderate agreement [105]. Of the 380 initial thread emails 24 were removed after coding as they were deemed to not be initial thread emails.

3.5 Results

What follows is a discussion of the 356 initial thread emails and the codes applied from our developed codebook. Overall, there was a total of 188 information requests, 150 examples of information sharing, and 18 off-topic emails.

We present the codes below by collecting them into similar thematic areas, such as updates and errors, and the associated tasks, such as troubleshooting. We illustrate each code with examples edited for readability, as well as the total number of emails coded from the final set, and percentage. Table 3.2 has further statistics, including total numbers of URLs shared, and KBs mentioned.

Table 3.2: Codebook for initial emails with counts, percentage, thread length statistics, and counts of KBs and URLs mentioned in the emails.

Theme	Code	Count	%	Thread Length				Total counts	
				Mean	Std. Dev.	Max	Min	KB#	URLs
Patch Prioritization	Update Info	51	14.3	6.43	7.54	34	2	287	355
	Update Query	25	7.0	9.48	9.73	34	2	17	5
Errors and Troubleshooting	Update Error Info	46	12.9	7.93	8.71	42	2	45	46
	Update Error Query	13	3.7	7.85	6.63	22	2	10	11
	Patching Problems	86	24.2	9.29	8.25	36	2	69	22
Threats on the Radar	Vulnerability and Attack Query	4	1.1	7.00	6.88	17	2	0	1
	Attack and Vulnerability Info.	12	3.4	5.50	2.91	10	2	1	16
How-to and Tools	Scenario/Situation Request	15	4.2	8.73	7.19	30	3	5	3
	Tool Info.	19	5.3	6.16	4.57	16	2	0	21
	Tool Query	29	8.1	7.69	7.56	37	2	3	4
Mechanics and Documentation	Update Mechanism Query	14	3.9	5.21	3.07	12	2	123	122
	Update Mechanism Info.	12	3.4	8.00	8.82	33	2	8	16
Vendor Behaviour	Windows MS Information	10	2.8	14.40	19.63	55	2	0	14
	Windows MS Query	2	0.6	5.50	4.95	9	2	0	0
Off Topic and Community	Community	7	2.0	11.14	10.90	32	2	0	5
	List Info	3	0.8	5.67	3.21	8	2	0	1
	List Query	3	0.8	9.00	5.29	15	5	0	0
	Off Topic Info	5	1.4	4.60	3.21	9	2	0	52

3.5.1 Patch Prioritization (n=76, 21.3%)

Two crucial activities after a patch is released are learning about its release and, if there are multiple patches, deciding which ones to prioritize [28, 29]. This prioritization includes understanding the changes the patch is likely to make, risks of not patching (vulnerabilities), and likely impacts on sysadmins' systems. Similar to the related work, we found that patch prioritization is a common activity for list members.

3.5.1.1 Update Info (n=51, 14.3%)

This code was predominantly related to the announcement of a patch's release, usually with URLs to the KB articles. Table 3.2 highlights URL usage with only 51 emails containing 287 KBs and 355 URLs. While the majority of content was focused on Microsoft, announcements were made for a range of products including Apple and Adobe. Announcements included, re-releases or hotfixes, and even online murmurs of a patch's upcoming release. The sender would provide URLs, and often quote important information directly from the source along with any other observations, as seen in the example below:

> "[URL to MS Support Article KB4090913]

> This update includes quality improvements. No new operating system features are being introduced in this update. Significant changes include the following:
>
> -Addresses issue where some USB devices and onboard devices, such as a built-in laptop camera, keyboard, or mouse, stop working. This may occur when the Windows Update servicing stack incorrectly skips installing the newer version of some critical drivers in the cumulative update and uninstalls the currently active drivers during maintenance.
>
> -Addresses issue where some devices may fail to boot with ”INACCESSIBLE_BOOT_DEVICE”.
>
> Release notes updated to reflect that this DOES address the inaccessible boot device issue.”

On Patch Tuesday, external groups, such as GHacks, system administrator subreddits, and AskWoody, create lists of Microsoft patches with prioritization-related information such as: platforms impacted, severity of vulnerability, URLs to KB articles, and even an executive summary with advice regarding the patches to prioritize for that month. For example:

> “...
>
> Executive Summary
>
> Security updates are available for all supported versions of Windows (client and server). Other Microsoft products with security updates are: Internet Explorer, Microsoft Edge, Microsoft Exchange Server, PowerShell Core, Adobe Flash, Microsoft Office. No critical vulnerabilities for Windows versions but for Microsoft Edge and Internet Explorer. Microsoft lifted the antivirus compatibility check on Windows 10 version 1607, 1703 and 1709.
>
> Operating System Distribution
>
> - Windows 7: 21 vulnerabilities of which 21 are rated important
> - Windows 8.1: 20 vulnerabilities of which 20 are rated important
> - Windows 10 version 1607: 29 vulnerabilities of which 29 are rated important
> - Windows 10 version 1703: 28 vulnerabilities of which 28 are rated important
> - Windows 10 version 1709: 24 vulnerabilities of which 24 are rated important
>
> ...”

3.5.1.2 Update Query (n=25, 7.0%)

In this code, the sender is looking for information about a particular patch with the goal of gathering information as opposed to fixing a problem. These emails are generally asking the community for patch information not found in documentation or asking for the community's opinion about a patch.

For example, the below request asks if a particular patch still exists. Patches can go in and out of circulation as Microsoft removes and replaces error inducing patches and the sender is uncertain if that is what they are seeing.

> "Did Microsoft withdraw the 1709 July 10 CU? On endpoints with the July 10 SSU (KB4339420) installed, WU is not offering the July CU (KB4338825)."

In the next example, the sender points out the number of side-effects of installing the patch, asks if installing it is really worth it, and then subtly asks if anyone is aware of an upcoming replacement patch with less side effects.

> "I know this may be old news for some of you but I generally wait a day or so before applying patches. When I read about this patch and see the known issues and workarounds -
>
> [URL to MS Support Article KB4088878]
>
> - is this patch really necessary? Looks like side effects after taking a drug that will fix one thing but make you suicidal. And when is Microsoft going to provide this update...in a future release? And we're supposed to limp along?"

3.5.2 Errors and Troubleshooting (n=145, 40.8%)

The largest theme found in our dataset was focused on the errors and troubleshooting caused by newly released patches. Moreover, the mailing list attempts to keep track of and announce any indications of updates causing problems that may have been observed elsewhere on the web. Due to the cyclical nature of Patch Tuesday, each month we would see the latest error trends. For example, March and April had reoccurring issues with patches impacting the Network Interface Controller (NIC). The community was continually on the look out for information regarding errors, and this could be sourced through citations to blogs, or forums, or through the problems brought to the list by community members. This citation behaviour is shown through the count of KB Numbers, and URLs found within this code, as seen in Table 3.2. These initial pieces of data would accumulate and eventually be recognised by Microsoft, through

official statements, and proposed workarounds. These would then be fed back into the community as a double confirmation of the issues.

3.5.2.1 Patching Problems (n=86, 24.2%)

This code was assigned to emails where the sender found themselves in the debugging stage following the application of a patch. The sender would be focused entirely on receiving information that would explain and alleviate the symptoms of the offending patches. The majority of these emails followed a very similar structure, in which the sender would describe the observation they had made, and the context in which it had appeared, such as in testing environments or in production systems. The sender would elaborate in great detail on the context of the error, such as the system version, the deployment method, the patches applied (usually referenced using KB number), and even the work they had performed to work out what was going on. This could take the form of scripts or listing commands they had run. The sender would also describe any online research they may have conducted, often indicating that reaching out on the list was their last hope. For example:

> “I patched my Dev/Test environment last night and I have 6 servers that failed. I deployed through SCCM, most of them showed an error description of Software update execution timeout. So, I changed the max run time on the updates to 90 minutes. This morning, I ran manual update scans on these servers, they all found and downloaded the updates, but 6 of 8 failed again, this time with an error code of 0x800705b4. Looking this up, it refers to Windows Defender. I found one article, saying to verify Windows Defender service is running, which it is on all of these servers
>
> Is anyone else seeing this issue?”

The final call for any similar observation or workarounds was another prominent feature of this code, and was complimented by other senders who would refer to older emails made in that month or previous, which had similar error features to their own issues. For example,

> “After a patch install on Wednesday night, we saw about 6 virtual W2012R2 servers stuck at the loading windows stage. The servers had obviously tried to auto reboot, but didn’t come back up properly. The VM’s had to be reset and then came up normally - no NIC issues.
>
> The patches installed were: KB4088876, KB4089187, KB4088785 and KB4088879. Nothing in the eventlog, known issues section and I don’t remember seeing anything on this list.

> Anybody else seen boot issues on W2012R2? I'm guessing it might be either KB4088876 or KB4088879 (or both) but I have no evidence for this.
>
> ..."

3.5.2.2 Update Error Query (n=13, 3.7%)

We distinguished Update Error query from Patching Problems by checking the sender did not explicitly state that they were currently troubleshooting. Instead, the sender was requesting more information regarding rumors of errors discussed elsewhere or by clients/ other colleagues. As seen here where the sender lists errors they have observed:

> "Please note this is not happening to all systems. I have updated two home pcs to 1803 with no issues
>
> But am seeing forums and consultants starting to report some issues. Wondering if anyone else is seeing anything similarly and if so can you email me directly so we can grab some log files?
>
> 1. Symptom: Desktop is unavailable
>
> [URL to Windows10Forum Thread on 'Desktop Unavailable' Error]
>
> [URLs to TomsHardWare Thread 'Desktop Unavailable' on Error] ...
>
> It looks like this: [URL to Twitter, Picture of Error]
>
> 2. Roll back loop [URL to Reddit Thread on Windows 10 Roll Back Loop Error]
>
> ..."

Senders would ask if community members had any further information pertinent to the described bug. The focus was on actionable intelligence, such as workarounds or contextual triggers for bugs, like the offending patch and the applications it interfered with. Asking the community for this information allowed the sender to perform a risk analysis as to whether the known issue was anything they would have to worry about in their patching strategies.

> "[URL to MS Support Article KB4103718]
>
> For those of you that did suffer a loss of networking after the May updates, what exact nic card did you have?
>
> What brand of computer?"

3.5.2.3 Update Error Info (n=46, 12.9%)

Here the sender shares information regarding the existence of potential errors found in that month’s batch of patches. This information could take the form of forum thread discussion of problems, to news or blog articles detailing problematic patches. Often, these URLs could also be sourced from other online communities (Technet, MS Answers, Reddit, Twitter, etc.), or the patch documentation. For example:

> “I just wanted to pass this along and wondered If anyone else has come across this issue yet..? We haven’t but apparently many others are now:
>
> [URL to Reddit Thread on Error following July Updates] ”

Data could also come directly from the sender’s system, however they would not explicitly ask for help troubleshooting. Their intention was to make the community aware of a problematic patch, as they may have already raised the issue with Microsoft, or had simply uninstalled the patch. For example:

> “We are running Office Professional 2016 64bit (msi install) on Windows 10 Enterprise v.1703 and encountered the Word has stopped error when double clicking a Word file in File Explorer to open it. You can start Word and open files within Word without the issue.
>
> ...
>
> Uninstalling the patch resolved the issue”

If available, they would hint at a possible workaround or suggested mitigation strategies for the errors. If this was not available, the sender would often ask for thoughts, or if anyone else had first hand experience with this issue.

3.5.3 How-To and Tools (n=63, 17.7%)

We observed that senders were seeking the advice and suggestions from the community for a range of patching related tasks. These codes appeared to be eliciting the experiences of the community to inform their current process, or to aid with proposed future changes. For example, we saw situations in which the sender was seeking the recommendations of tools for a given task, such as Automatic Deployment Rules (ADR), or asking for for PowerShell scripts that were shared amongst the community. Asking the community for these recommendations, or experiences, allows community members to better understand, and develop their systems and patching process.

3.5.3.1 Scenario/Situation Request (n=15, 4.2%)

In this code, the sender would set out a patching scenario, or a situation they needed guidance on. Senders would give future planned changes to their systems, or their current set-up, and look for suggested plans, or the thoughts of the community. For example:

> "We are setting up a WSUS server for Windows 10 systems. The current plan is to limit access to only new or upgraded devices so the old system can sunset with older systems. This should also help with the different management styles of the newer OS, and the storage requirements.
>
> How would you manage this transition?"

Sender's may also enquire about how community members tackle certain aspects of patch management, such as system upgrades, updating portable machines, or getting to grips with monitoring patching levels. Senders were looking to elicit the experiences of list members, and to gain from the knowledge of others. For example:

> "While we have the majority of our systems in-house, there are a handful of portable machines that leave our LAN frequently. Currently utilizing WSUS for internal machines with a GPO pointing towards it for both reporting and update feed. On the portable machines however I set a GPO for them to download from Microsoft directly since they are usually not here. Problem is, there are a few patches that we either do not want installed, or Microsoft can't figure out how to get them to work properly which I would like to filter out. Just wanting to get some pros and cons of which way or another you all have experienced."

Given the scale of Microsoft, gaining from the knowledge of others is highly valuable, and allows sysadmins the opportunity to avoid pitfalls and follow in the steps of those with more experience.

3.5.3.2 Tool Info (n=19, 5.3%)

Here the sender would share the announcement of a tool, service, or script, which they deemed potentially useful in the patching process. We did not include direct discussion of patches for tools in this code, and instead only considered examples where new features or announcements of intended products pushed a sender to share with the community. For example,

> "WAC or Windows Admin Center creates a new method for managing PC's for help desk functions. Update management is included in this.

> Have any of you tested this? It looks like it is only compatible with Chrome and Edge and 2012 and newer OS. Oh, and it doesn't replace RSAT or MMC completely.
>
> [URL to Microsoft Cloudblogs WAC Announcement]"

It could also take the form of settings and tricks that a sender had found, and was willing to share with the community.

> "I'm in the middle of creating a reference image for 1709 and have found a way of disabling the silently installed apps and removing the associated tiles from start. These changes need to be applied to the ntuser.dat in C:/users/default. It may help someone:
>
> Windows Registry Editor Version 5.00
>
> ```
> [HKEY_LOCAL_MACHINE\defuser \Software\Microsoft\Windows
> \CurrentVersion\Conte$\DeliveryManager]
>  "FeatureManagementEnabled"=dword:00000000
> "OemPreInstalledAppsEnabled"=dword:00000000 ... "
> ```

3.5.3.3 Tool Query (n=29, 8.1%)

Here, a sender would pose a question regarding a tool, software (SCCM, ConfigMgr), or script related to an element of patching. This could take the form of feedback regarding a particular tool, or looking for suggestions of an alternative. It should be noted that this code was not limited to only Microsoft products, as a number of third party software and vendors were routinely discussed. The sender would often detail the intricacies of their systems and indicate exactly what they wished the tool to do. For example:

> "Can anyone recommend an alternative to Ivanti/Shavlik for a small business (just 12 licences). We've been very happy with it since the days of hfnetchk, but it seems neither the reseller or Ivanti themselves want our business any more - must be too small - so we're looking for an alternative. Ideally agentless and cost effective without too much admin input or scripting. Can't use an MS offering as we have moved away from Windows servers, so it needs to run on a workstation and not need AD."

Alternatively, the code contained examples where senders were working out errors, or best practices with the tool in question. For example:

> "I currently use Automatic Deployment Rules in SCCM to deploy and manage Windows updates/patches. In the event that an emergency patch is released to the WU que, how do I push this update through. My current ADR is set to run each patch Tuesday with a deadline of 7 days after."

3.5.4 Threats on the Radar (n=16, 4.5%)

SysAdmins must be constantly vigilant for new, and potentially devastating vulnerabilities that could impact in their systems. Being aware of these issues will allow them to plan their patching accordingly, as well as mitigate the risks of yet unpatched vulnerabilities. Cyber Security is constantly adapting as new vulnerabilities are found every day, and hence the community attempts to remain current through their information collaboration. As with previous themes, there were popular topics discussed, such as fallout from the Meltdown and Spectre patches released earlier in the year (but not released within our chosen window). Clarification would be sought on whether vulnerabilities had been plugged following the released patches, or if there was potentially more needing to be done to prevent successful attacks on their systems.

3.5.4.1 Attack and Vulnerability Info (n=12, 3.4%)

This code captured information shared regarding the existence of a newly discovered vulnerability or attack in the wild. This could include the announcement of a new vulnerability, for example:

> "Another Spectre/Meltdown type vuln.
> [URL to US-Cert Alert (TA18-141A)]
> [URL to Verge Article on CPU Vulnerability]"

It could also include discussion of the emergence of attacks, and proof of concept code, such as:

> "In the past 12 hours, "XPN" has released and updated exploit code for Total Meltdown.
> [URL to XpnSec Blog on Meltdown]
> Kevin Beaumont has confirmed that it works.
> [URL to Twitter Post]
> Win7 and Server 2008R2 systems are left with few options.
> [URL to Ask Woody Meltdown Post]"

3.5.4.2 Vulnerability and Attack Query (n=4, 1.1%)

Senders would ask the community for information regarding known vulnerabilities or new attacks in the wild. These requests were linked to some observations, such as news articles or blogs, that spur them to enquire further. This could also take the form of an

observation made about their system, as they seek a verification from the list. Senders would often ask for clarification regarding the technicalities of the vulnerability, or the best mitigation strategies. For example:

> "Just when it seemed like 2018 couldn't get any worse for IT, does anyone have information re: this threat?
>
> Googling vpnfilter will source several reliable resources that this is a legit threat.
>
> Based on my reviews of available information I'm not sure at this point just how the stage1 succeeds. If a device does not have remote management enabled that should address this problem - unless there is a flaw that is being exploited on the device itself.
>
> Can anyone comment on how the exploit succeeds and what if any are the mitigating factors?
>
> Particular to this new headache it would seem worth asking what devices out of the box allow for remote access? None that I know of or have worked with."

3.5.5 Mechanics and Documentation (n=26, 7.3%)

Patching is a complicated business with multiple sources of information, with documentation, and definitions of vendor terminology. However, due to the scale, and complexity of updates, community members would find themselves confused around terminology and their real-world impact, i.e. how patches actually worked. Inconsistencies in documentation would result in senders looking to the community for reassurance or clarification regarding terms, and the mechanics of patching mechanisms like WSUS, or how different branches of releases were delivered. These codes would contain URLs to the offending documentation, with a total of 138 found within this theme. Understanding the documentation, and therefore how patches were intended to work, is vital to a successful patching schedule. Given the scope of Microsoft, we can see that mistakes in clarity are possible.

3.5.5.1 Update Mechanism Info (n=12, 3.4%)

Windows Update Service adapts with time, with changes being made to the inner workings of both patches and patching services. Information sourced in this code could include blog posts made by specific teams within Microsoft, as senders keep the list aware of intended changes to the patching landscape:

> "Office 365 is changing its build numbers to a five-digit format.
>
> [URL to MS Support Article on Office 365 Changing Build number Format]"

Community members would also monitor discussions found on the list, and be compelled to correct or direct the list to sources which would explain patch mechanics:

> "I've seen several list members recently noting that computers with settings to the contrary are being forced to 1709. Please see KB4023814, particularly this paragraph:
>
> 'Windows 10 version 1607 and version 1703 are not yet at 'end of service.' However, they must be updated to the latest versions of Windows 10 to ensure protection from the latest security threats.'
>
> Woody wrote a piece yesterday on Computerworld saying the telemetry level setting is also a factor on who gets the forced upgrade.
>
> [URL to ComputerWorld Article on Telemetry Settings]
>
> We're a SMB, Microsoft does not care what I think. But to those of you that have lots of seats and TAMs, is this acceptable to you? It seems unconscionable to me that you follow all Microsoft's latest rules for reg settings and group policy to defer updates, and despite running a version of Windows that had not yet reached EOL, they update you anyway."

3.5.5.2 Update Mechanism Query (n=14, 3.9%)

We found that changes or subtle inconsistencies in the wording surrounding the description of technical aspects of documentation and patches would cause senders to look for clarification. Patching is extremely detail orientated, therefore anything which did not fit into the sender's mental map of how patches are constructed, applied, or delivered, would cause them to reach out to the community to verify. For example:

> "If I go here: [URL to Microsoft Windows 10 Release Information]
>
> I see Semi-Annual Channel (Targeted) is 1709 and Semi-Annual Channel is 1803. This seems backwards to my understanding of those terms. Am I missing something?"

Update mechanisms also captured misunderstandings regarding how patch delivery mechanism worked and what terminology meant in patching scenarios. As seen here:

> "We are looking at upgrading our Windows 10 fleet and yes we do have a mix bag from 1511 to 1803. We use Configuration Manager 1710 with WSUS.

> Windows 10 1511 is now out of support as we all know, does that mean we cannot upgrade the 1511 (and even 1607) Windows 10 to 1803 via WSUS? I think the answer is yes, and wanted to check with the experts as I am sure we are not the only company that lets say a little slow to the Windows 10 upgrade party.
>
> Let me know if you need any more information and thanks in advanced"

3.5.6 Vendor Behaviour (n=12, 3.4%)

We observed information regarding Microsoft as a company, its direction, or its policies related to patching. As a sysadmin, one must navigate the relationship with the vendor, as press statements and announcements of intent are indicators of potential shifts and alterations in the patching landscape. Keeping aware of company announcements and internal changes is useful as it allows for anticipation of the changes as opposed to being caught in the cold. We found that these discussions would routinely cover what was believed to be inadequate responses of Microsoft, with many using these to justify their beliefs and patching strategies. The responses were overwhelmingly negative, and it was clear in our community that Microsoft is not seen as a communicative partner. For example, through our error codes we saw passing remarks regarding the state of Microsoft's testing of patches, with some stating, "Do they even test these?!"

3.5.6.1 Windows/Microsoft Info (n=10, 2.8%)

This code captured the sharing of information regarding the behaviour of Microsoft. Announcements made by Microsoft and were regularly found to be unsatisfactory. The sender would react negatively to a decision or response from Microsoft, and would appear to vent regarding the "disregard" Microsoft was showing towards them and other admins. For example:

> "[URL to MS TechCommunity Windows 10 Ask Me Anything]
>
> IMHO the question really wasn't answered.
>
> So if you were in charge of patching/servicing and testing at Microsoft, how would you gain back the trust?"

Senders would often point to decisions being made, and often use them to justify their stance on certain patching services, as seen here:

> "Here is a reason to not deploy LTSB:

> [URL to MS TechCommunity Windows 10 Ask Me Anything - LTSB for Pro Users]
>
> We've recently announced that Office will no longer be supported on LTSC in the future (today, it is still supported on certain LTSC versions). So please keep this in mind. Here's the blog with that announcement:
>
> [URL to MS Technet Blogs on Windows IT Pro, Changes to Office and Windows]
>
> ..."

3.5.6.2 Windows/Microsoft Query (n=2, 0,6%)

In this code, the sender would reach out to the community for clarification regarding the actions of Microsoft as a company or Windows as a service. These queries would often follow from announcements or observations made regrading adjustments made to update services or related products. More often than not, senders would be angry or upset with the proposed change, often the discontinuation of a product used in patching. Those affected are pushed to reach out the community, to create a discussion as to why Microsoft will have made this decision. For Example:

> "Talking to an engineer about Bitlocker and he asked why I don't have an Azure presence yet? I say I don't need one. He says you do know you will be required to have one in the very near future. I scratch my head. Is this truly the path Microsoft is taking and will I be required to have an AD-Azure presence even if I don't want one?
>
> I'll sit back and read your replies."

3.5.7 Off-Topic, List Rules, and Community (n=18, 5.0%)

The community is clearly a trove of data on patching related issues. However, we observed examples of emails that were not directly patching related, and were considered off-topic, list information, or community related.

3.5.7.1 List Info (n=3, 0.8%) and List Query (n=3, 0.8%)

These two codes captured emails which informed community members of the rules and running of the mailing list, with List Info focused on sharing this information. This could include moderators stating they would be away for a while, or GDPR's affect on the list, as seen here:

> "Since

a. this list is voluntary

b. this list is not selling anything

c. this list is not selling email addresses

d. it is merely a peer resource

Therefore this list is not covered by GDPR rules and thus does not need to ask everyone from the EU to re-opt back in.”

The opposing code, List Query, captured queries regarding what was appropriate content for the list, such as the example below:

> “Am I mistaken to say that this list primarily refers to servers? I work in the desktop environment and follow this list to help determine what to release.”

3.5.7.2 Off Topic Info (n=5, 1,4%)

Not all emails coded were directly related to patching or the patching cycle, but could still be related in some way. For example, we saw the list share tools or information regarding sysadmin tasks, such as Amazon Web Services (AWS), seen here:

> “For those on AWS – [URL to Github Hammer Project]
>
> [URL to Medium article on Dow Jones Hammer Tool]
>
> Today, Dow Jones Tech is pleased to open source Dow Jones Hammer, a DevSecOps tool that lets you identify and proactively fix misconfigurations in cloud workloads.
>
> ...”

Moreover, we saw examples of information source sharing, which took the form of members posting their go-to Twitter followers:

> “Hi Folks,
>
> This is me and my Twitter Following List.
>
> **Microsoft People**
>
> [List of 10 Twitter Accounts]
>
> **Microsoft**
>
> docs.microsoft.comVerified $@docsmsft$
>
> The PowerShell Team $@PowerShell_Team$
>
> Windows InsiderVerified $@windowsinsider$
>
> .NET Foundation $@dotnetfdn$

Visual Studio Team ServicesVerified *@VSTS*

.NET TeamVerified *@dotnet*

Windows Defender Security Intelligence *@WDSecurity*

Microsoft SecureVerified *@msftsecurity*

Security ResponseVerified *@msftsecresponse*

Microsoft SupportVerified *MicrosoftHelps*

MicrosoftVerified *@Microsoft*

MS Windows IT ProVerified *@MSWindowsITPro*

Microsoft NewsVerified *@MSFTnews*

WindowsVerified *@Windows*

Microsoft Channel 9Verified *@ch*9

Media, NetCasters & Others

[List of 22 Twitter Accounts]

..."

3.5.7.3 Community (n=7, 2.0%)

This code captures the camaraderie shown amongst list members, primarily in the form of thanks, or appreciation to the list for its help.

> "I just want to give a HUGE **THANK YOU** to everyone in this group for helping me *SO* much over the last years. The information provided here is invaluable, and the time and effort everyone takes to submit and SHARE information is, well breathtaking. I've learned SO much and appreciated folks stepping up and answering my (sometimes neophyte) questions
>
> As it turns out, it's time for me to ride off into the sunset and enjoy retirement. ...
>
> It has truly been my honor being part of such a magnificent group."

Thanks were the most common example, but we also found emails that were part of community wide efforts, such as a list moderator thanking the community for help in the writing of an open letter to Microsoft [106].

> "[URL to Computer World article on Open Letter to Microsoft]
>
> My deepest thanks to all who participated.
>
> Bottom line the fight for change has just begun."

3.6 Discussion

3.6.1 Online Community of Practice

Upon completion of our analysis and development of the codes and themes, we took a bottom up approach to understand our results and better articulate the findings within the context of system administration. Reading through the work of Kandogan et al. [5], they introduced the concept of a Community of Practice (CoP). This observation crystallised many of the codes, themes and observations that I had made during the creation of our codebook and ongoing investigations into the PatchManagement.org mailing lists, hence we use this concept to more accurately articulate the impact of our results. CoP is a concept that emerged from the social theory of learning proposed by Lave and Wenger [107], and was further developed in 1998 by Wenger [108].

Essentially, in a CoP, a group of people who share a craft or profession build a community where members can share information and experiences as well as learn from each other. CoP's can either form naturally or be explicitly created and grow through the discussions of community members around topics such as best practices. A CoP can also be virtual [109]. A CoP has three key characteristics:

- **Domain** - a body of knowledge which allows for mutual understanding amongst the members of the community, and guides the learning and goals of the community;
- **Community** - a community feeling provides the social framework for learning. A strong and accepting community fosters discussions and encourages further learning opportunities; and
- **Practice** - the members of the community are practitioners in the domain, therefore through their social interactions they gain an insight into best practices for their chosen domain.

The **domain** of PatchManagement.org is software patches – release, intended and unintended side effects, and mitigation of problems introduced by the patches. Members are mostly sysadmins, and therefore **practitioners** in the domain – part of their role is to apply patches in a way that is both timely and minimally disruptive. Our qualitative analysis, in particular the smaller themes, clearly demonstrate that the mailing list also has the third element of **community**. There are well enforced rules that allow list members to debate the advantages and disadvantages of each patch openly.

3.6.2 Tasks are Highly Complex and High Risk

Typical of contexts that generate Communities of Practice, patch management is a highly complex task. It requires sysadmins to manage potentially large networks of computers, each of which have a different set of requirements and setups. Then, with the release of each patch, they must balance the risks and benefits of installing it [13], not just globally but on each computer or set of computers. The information that sysadmins have at release is skewed and incomplete. They know what issues the patch is supposed to fix, but not what new issues the patch is likely to cause across the entire network. Therefore, they need a way of monitoring emerging issues for data that they can use in their own decision processes.

The online CoP of PatchManagement.org is a safe space where sysadmins can access up to date information and obtain expert advice on their decisions, as shown in the themes of Errors and Troubleshooting, and How-To and Tools.

Since sysadmins may be isolated in their own companies or work in very small teams, it is invaluable to have access to the "wisdom of the crowd" when it comes to a high risk activity such as patch management. Errors, such as installing a buggy patch, or not installing a patch for a security vulnerability that is then exploited, can take large numbers of computers that are vital to the functioning of an organization offline very quickly [42]. Often it is not possible to mitigate all the associated risks, so instead sysadmins must manage their risk appetite and decide how much and what type of risks they will take. The open discussion of questions such as how quickly a patch should be deployed, or whether difficult-to-patch computers should be protected behind a firewall, help the community come to a consensus on what best practice mitigations are "enough".

3.6.3 Time Pressures Require Prioritisation of Work

Looking more specifically at the monthly patch cycle, one large source of complexity is the time pressure to get patches installed quickly [13, 43]. Most official guidance on patching, such as the UK's Cyber Essentials [3], recommends installing patches as soon as possible to avoid potential compromise. Practically, however, installing all patches at once is not possible as each patch should be tested and then deployed. Some months as many as 60+ patches can be released by Microsoft alone in one day. To handle the overload, sysadmins use available information to prioritize patches that have serious security implications over those that do not [28].

We saw evidence of the PatchManagement.org community openly discussing what patches needed to be prioritised and which could or should be delayed. The community’s wisdom was drawn upon to prioritise not only the patches themselves, but how severe the vulnerabilities they impacted were and what systems would be impacted.

3.6.4 Verification is Hard to Impossible

Verifying that a patch is “safe” or “working” is a challenging (and occasionally impossible) problem partially because there is no good definition of “safe” or “working” [110]. After installation, sysadmins “test” the patch by doing everything from observing a lack of errors, conducting basic actions like opening email, deploying it to beta testers, or running a full battery of automated tests on a dedicated testing environment. However, these activities do not produce definitive proof that the patch is good, just that no errors have been found, therefore the decision to move to production systems is always a gamble.

Patch failures can also have several sources beyond the patch itself. The purpose of a patch is to change how software works, that change can then have side effects for other software or react badly to specific configurations [5, ch3]. As a result, when a patch is seen to fail, the first question is what and/or who is the cause rather than assuming the patch itself is problematic.

The PatchManagement.org community actively shared information about observed and potential patch problems. The list allowed members to collect together information they found across the Internet to bring together a picture of what patches were causing issues drawn not only from their local experience, but also from those of other related communities. For members who were currently struggling with post-patch issues, the list offered a place to openly discuss the problem and gain not only solutions but also learning about how systems like Windows work.

3.6.5 Fixing Requires Evidence

Getting problematic patches fixed can also be difficult. There are many possible sources of issues beyond the patch itself. So getting a vendor, such as Microsoft, to fix the patch requires providing evidence that the problem being observed is really caused by the patch and not something else.

In this regard, the list served as a collective method of contacting vendors and a source of multiple cases to provide to the vendors as evidence. When a problem was

identified, list members would comment that they “had a ticket with Microsoft” and promised to report back to the list with the response. Some members had elevated Microsoft Support contracts and could use them to get better support which was then passed on to the list.

3.7 Conclusion

This Chapter 3 represents the first analysis of an information source known to be used by sysadmins when tasked with patching their systems [28, 29]. We found that these sources represent a vast range of distinct topics related to patching, with the most prominent being related to issues caused by patching or with community members. Additionally, we found that these threads often were the most popular and contained a number of links, highlighting the citation behaviour of other information sources. Finally, we identified that this mailing list represents a Community of Practice which relies on the collaboration of its users to provide a centralised patching information source and access to veteran patchers and their experience.

Chapter 4

‘To Patch or not to Patch?: A Case Study of SysAdmins’ Online Collaborative Behaviour[1]

4.1 Introduction

Patching is a regular event [28], and yet each release contains new patches, new vulnerabilities, new features, and, therefore, new potential risks. Yet we do not understand how the efforts of online communities make the risks of a particular patch apparent in a timely manner given the race to close systems’ window of vulnerability.

Both academia and industry have stated the importance of pushing updates as quickly as possible [2, 3], however sysadmins whose day-to-day tasks will include updating [60] may also be juggling a number of other diverse, complex, and highly collaborative IT maintenance tasks [47, 5]. Sysadmins must remain aware both the risks of avoiding a patch, and the potential risks of patching [28, 12, 29]. Therefore, many may be motivated to simply “wait it out”, as the errors and bugs of released patches are found and remedied, before deciding to patch [13].

Deciding to patch hinges on balancing risk. Push a patch too early and you may inadvertently break business essential services. Push too late, and you may provide ample time for exploits to be developed and targeted at your systems. Identifying this optimum time is a difficult process for sysadmins [13], requiring a number of stages

[1]The following chapter is composed of work conducted and compiled into a research paper. The ethnographic approach was conducted by myself, and the case selected and construction done in collaboration with my supervisor, Kami Vaniea. The first draft of the paper was written by myself and edited in collaboration with all authors.

and contextual information gathering to even consider patching [28, 29, 30]. The previous Chapter 3 highlighted that one such source used by sysadmins, focused a large component of their collaborative efforts regarding issues cased by patches. Furthermore, the work showed the recurring behaviour of citing information from a number of different sources, hinting that patch information and knowledge is being generated by a number of distinct communities. To explore this observation further we decided to focus on the following research questions:

RQ3 How does the knowledge of risks associated with a patch develop over time?

RQ4 How do the various communities support the collation and synthesizing of available information for use by sysadmins and other patching communities?

Given the clear dependence of sysadmins on both internal [5] and external information sources [28], we address our research questions through an in-depth, descriptive case study to illuminate the online information life-cycle of two Microsoft patches. We track these patches across a number of online sources, from initial release to a final all-clear, "time to patch" announcement by a key community, almost a full month after release. Our case study highlights the scale of collaborative work found within and across communities of practice, which provides sysadmins with advice and data which is integral to deciding when to patch. Our case indicates areas where we could not only aid, but reduce the time necessary for information searching by sysadmins needed for making their patching decisions.

We find that a number of information sources compile and collect knowledge from work performed elsewhere on the Internet. The use of forums and mailing lists also facilitates discussion and detective work that generates necessary information to inform patch decisions. These results reflect a number of individual Communities of Practice [25], which collaborate to share knowledge and propagate information through out a wider Network of Practice [27].

4.2 Methodology

We use a case study [111] approach to gain an in-depth understanding of the types of information and processes sysadmins must deal with when searching for information about patches. By narrowing our focus to a pair of patches, we were able to do an exhaustive search of available information and fully trace when information became

available, where it was posted, how it was incorporated, assimilated, referenced, cited, and used by other people in their own posts, and how it ultimately impacted the growing recommended action consensus within different communities.

A case study also allows us to develop a detailed case-level understanding of our two research questions around the progressive release of new information (RQ3) and the different ways online communities support information collation and synthesis (RQ4). While this understanding will consequently be case-specific, the lessons learned can inform later more general investigations.

4.2.1 Case selection

We decided to focus on Microsoft patches for three key reasons. First, their products are a major feature of the corporate world and the associated support communities are large and well developed. Second, Microsoft patches are released on a regular one month cycle that the community is very familiar with communal processes around patch release. Last, Microsoft labels all patches with a unique Knowledge Base (KB) number, which is then logged in the Microsoft Knowledge Base which is a public repository containing articles, relating to products or user-encountered problems. The KB number is heavily used by sysadmins in their online discussions [87] making such posts easy to accurately associate with a specific patch.

We selected our patch case with the following features, which exemplify many of the issues sysadmins are known to struggle with [87, 28, 29]: is security critical; has undocumented problems; has a temporary workaround; involves interaction with vendor (Microsoft); and addresses a "typical" vulnerability.

Security critical patches are reported to have the highest priority by sysadmins [28, 29] since the vulnerabilities they patch can be serious and lead to compromise if not quickly corrected.

Undocumented problems, *temporary workarounds*, and *interaction with Microsoft* are all events that can happen during a patch month. All three are future events that are impossible for sysadmins to predict at the time of patch release but can be problematic later. Undocumented problems are the largest issue since they are problems that may appear without warning damage system functionality. Workarounds are inconvenient, but may allow an admin to keep a problematic patch installed rather than uninstalling it so they are valuable to know about. Finally, interactions with Microsoft provide official guidance and corrections but may require manual intervention to install.

A "typical" vulnerability patch was targeted for research and practical reasons. While famous vulnerabilities like Meltdown and Spectre can be quite interesting to study, the media frenzy around them also leads to non-standard behavior. Organizations might allocate more resource or allow sysadmins to take more risks than normal. The amount of media attention also increases the number of people posting about the patches and effectively drowns out the voices of the people we most want to study. Additionally, while some patches become famous in hind-sight, any critical patch has the potential to later develop into a WannaCry level global disaster. So for this case study we focused on finding a patch case that is relatively unknown to the general public.

To find potential cases we searched the PatchManagement.org [26] email archive which we previously showed to be an online Community of Practice focused on sharing patch information [87]. We used regular expressions to find email threads containing KB numbers and that looked to have heavily discussed problems, yielding 5 potential cases. We then took a more in-depth look at these cases to determine how well they met our other requirements, including looking at the KB articles, following links in the emails, and searching other communities like TechNet for the KBs. Ultimately we selected KB4034679 and KB4034664 which were released in August 2017. Two patches were selected because Microsoft releases patches in pairs, with one containing only security changes (KB4034679) and the other containing all fixes (KB4034664).

4.2.2 Data collection and Analysis

To collect data we took an approach similar to digital ethnography [112]. Our goal was to collect the information about our patches that sysadmins were actively looking at and engaged with. So we started with the PatchManagement.org email list and searched for all occurrences of the two KBs. We then looked through those email threads and followed all the links in the posts. If the resulting web page was related to the KBs, we recorded it, and then followed all the shared links on that web page. We continued this activity till all such links had been followed. We then took a look at the different sites this activity lead us to and created a list of those that were clearly aimed at sysadmins and patching, including AskWoody and TechNet. We then searched these sites for the two KBs and followed the same process as for PatchManagement.org. The result was 42 web pages (URLs) that contained content related to our KBs, several of these were forum threads which collectively contained a total of 489 posts or comments.

For data analysis, we chose to focus on the chronology of events as our interests are

in how knowledge of risks associated with the patches develops over time (RQ3). Prior work has also shown the patching process to be very time dependent and sequential in nature [28, 29, 74, 87]. The first author read through all 42 web pages including all the posts on them to construct a chronology of events, particularly instances where new knowledge was posted, shared onto a new community, or collated by a member in a larger post or email. They then walked through the chronology with the second author discussing the evidence and any points of uncertainty. Both researchers then worked together to group the detailed events into higher level conceptual groups using the previously identified stages of patch management [28, 74, 29] as a guide.

4.2.3 Limitations

We specifically selected a pair of patches that exhibited multiple types of problems that admins must handle including: rare problems, a work around, a hotfix which also had problems, Microsoft pulling the hotfix, and a second hotfix. While we feel that these patches exemplify common patching situations that sysadmins must handle, we also acknowledge that it is uncommon for one patch to have this many issues associated with it. We provided an early draft of our results to community members for feedback and validation of our observations. All community members provided consent to being nnamed explicitly and additionally provided positive feedback, stating that although our case is an outlier in some aspects, the behaviours demonstrated are reflective of the actions and contributions made by the respective communities and contributors.

A second limitation of our work is in how we found the posts and communities to analyze. We specifically started from the PatchManagement list and used the link structure in posts to find new relevant posts. While this approach allowed us to find all the information considered important enough to link to for our patches, it also excluded information that was less likely to be cited or simply not relevant for the patches we examined. Previous work has shown that professional mailing lists [28, 29, 30], and this particular community [87], are a prominent source of patch information for sysadmins.

4.3 The Cast: Places & Players

We start by providing some context about the prominent communities discovered from our initial community, PatchManagement.org.

4.3.1 AskWoody.com

The AskWoody forum, which is sometimes known as “The Wailing Wall of IT”, was started in 2004 by Woody Leonhard, as a place for news and discussions surrounding Windows and Office. The site itself, has multiple features related to patching, including designated Master Patch Lists and forum threads for specific platforms. It also features the Microsoft Patch Defense Condition Level [113], or MS-DEFCON system for short, which is similar in nature to the DEFCON system used by the US army. The MS-DEFCON system, shown in Figure 4.1, is a quick reference guide for community members on the current state of released MS patches, giving them a quick way to decide when it is safe to patch, with the scale ranging from ”Patches causing issues, avoid deploying” to ”All clear, apply patches.”

A notable feature of AskWoody is the use of the terms Groups A, B, and W [114] to describe people with different patching philosophies loosely linked to their willingness to trust Microsoft. The group terms were suggested in reaction to the shift by Microsoft from individual patches to cumulative patches as well as some patching disasters that were a result of Windows 10 auto upgrades or that led to installation of “tracking” elements into the OS [115].

Group A people regularly install the cumulative updates. They may skip an update due to bugs or issues with the patch, but their philosophy is that patching is important and that new features are wanted. *Group B* people regularly install security-only updates, which are not cumulative. They value stability and do not want to install extra features which might bring unwanted elements into their systems. Similar to Group A, they may skip a buggy patch even if it is security related. *Group W* people do not believe in patching at all and generally advocate avoiding patch installation whenever possible. This group’s philosophy is not recommended by the community, but is respected as existing. In our case, Group A would install the cumulative KB4034664, Group B would install the security-only KB4034679 and Group W would install nothing.

On AskWoody, the different groups are represented through the use of tags with the group name, or in text as a convenient way of referring to a patching approach. For example, recommendations may state that a patch is suitable for Group A to install, but that Group B might want to hold off because it adds some extra features.

Figure 4.1: Microsoft Patch Defense Condition Level (MS-DEFCON)

4.3.2 Other Communities & Contributors

While our discussion focuses mainly on Patchmanagement.org and AskWoody, there are a number of other important partners in the Network of Practice, as we saw many users migrating between forums to discuss or declare issues once patches have been pushed. While linked, we believe that the behaviour exhibited on each respective site and forum is different. For example, the posts which are found on *Technet*, and its sister site *Answers*, appear to be more focused on getting Microsoft's attention, in comparison to the detective work found on AskWoody.

Günter Born is a freelance journalist and blogger whose work is captured in this case study. Born began as an engineer, before slowly transitioning into writing and blogging. He writes on a number of subjects, predominantly focused on technology, ranging from books for children and computing, to those dedicated for elderly users. He continues to post on the topic of Windows through his blog, "Born's IT and Windows Blog". Winner of the Microsoft Community Contributor Award (MCC) in 2011, he is also regularly recognized as a Microsoft Most Valuable Professional (MVP). He can also be found on a number of communities, such as AskWoody, and is a voluntary moderator and wiki editor for the Microsoft Answers forum. Microsoft hosts its own official web platform for the discussion of IT related products and issues, called *TechNet*. The site's main function is to provide an area for IT professionals to gain access to information, documentation and have discussions. It has a number of features, such as its own wiki, and documentation library. It also allows for blogs, which are contributed to by Microsoft employees.

Several other sites offer patch-related content such as curated lists of available patches and blog posts about vulnerabilities. These lists are created by a wide variety of organizations including: *Ghacks* [116], *Zero Day Initiative* (ZDI) [117], and the *SysAdmins, Audit, Network, Security (SANS) institute* [118]. News-oriented sites like *Krebs On Security* [119] also publish Patch Tuesday overview articles.

4.3.3 KB4034679 & KB4034664

Our patches relate to Windows 7 and Windows Server 2008 R2 platforms, which at the time of our case were still in their extended support cycles. Official support for these platforms expired in January of 2020. As mentioned previously (see section 3.3), Microsoft patches are released in pairs on Patch Tuesday where admins are expected to install one or the other but not both.

The first is the *monthly rollup*, which contains all available patches including patches from past months, both feature and security orientated. The second is the *security only*, which contains only the security fixes for that month.

The monthly rollups are available through Windows Update, Windows Server Update Services (WSUS), and the Windows Update Catalog. The security only updates are only available through WSUS and the catalog.

Below we present the two communities' perspectives on our two patches KB4034679 (security-only) [120] and KB4034664 (cumulative) [121] from when the communities first learn about them to when they are considered safe to install. At a high level, this story starts with Patch Tuesday release announcements, followed by community testing which flags problems, detective work to find the nature of the problems, a hotfix released by Microsoft to address the problems, more testing highlighting new problems, more detective work, Microsoft removing the buggy hotfix, Microsoft releasing a new hotfix, more testing, and finally a "safe to install" announcement.

4.4 Results

4.4.1 It's Patch Tuesday!

Our case begins on the evening of August 8th, 2017, 17:00 UTC with the standard "Patch Tuesday" announcement of released patches by Microsoft. August's collection of patches contained 41 security patches, covering all supported versions of Microsoft Windows. Patches were also released for Microsoft Edge, Internet Explorer, Microsoft SharePoint, Microsoft SQL Server, and Adobe Flash Player. The total number of vulnerabilities addressed in Windows on this Patch Tuesday was 48, with 2 vulnerabilities for Adobe Flash Player.

For each patch, Microsoft also releases an associated KB article with a standard set of information including *improvements and fixes* addressed by the patch, as well a summary of *Known issues in this update*. The information is updated as issues crop

Patch	CVE	Severity	Impact	Product
KB4034664 and KB4024679	CVE-2017-8624	Important	Elevation of Privilege	Windows 7 SP 1 and Windows Server 2008 R2 SP 1
KB4034664 and KB4024679	CVE-2017-8633	Important	Elevation of Privilege	Windows 7 SP 1 and Windows Server 2008 R2 SP 1
KB4034664 and KB4024679	CVE-2017-0174	Important	Denial of Service	Windows 7 SP 1 and Windows Server 2008 R2 SP 1
KB4034664 and KB4024679	CVE-2017-0293	Critical	Remote Code Execution	Windows 7 SP 1 and Windows Server 2008 R2 SP 1
KB4034664 and KB4024679	CVE-2017-8593	Important	Elevation of Privilege	Windows 7 SP 1 and Windows Server 2008 R2 SP 1
KB4034664 and KB4024679	CVE-2017-8620	Critical	Remote Code Execution	Windows 7 SP 1 and Windows Server 2008 R2 SP 1
KB4034664 and KB4024679	CVE-2017-8666	Important	Information Disclosure	Windows 7 SP 1 and Windows Server 2008 R2 SP 1
KB4034664 and KB4024679	CVE-2017-8668	Important	Information Disclosure	Windows 7 SP 1 and Windows Server 2008 R2 SP 1
KB4034664 and KB4024679	CVE-2017-8691	Important	Remote Code Execution	Windows 7 SP 1 and Windows Server 2008 R2 SP 1
KB4034664	CVE-2017-8635	Moderate	Remote Code Execution	Internet Explorer 11
KB4034664	CVE-2017-8636	Moderate	Remote Code Execution	Internet Explorer 11
KB4034664	CVE-2017-8641	Moderate	Remote Code Execution	Internet Explorer 11
KB4034664	CVE-2017-8653	Moderate	Remote Code Execution	Internet Explorer 11
KB4034664	CVE-2017-8669	Moderate	Remote Code Execution	Internet Explorer 11

Table 4.1: Vulnerabilities addressed by KB's 4034679 and 4034664.

up and are reported by the community, and includes information about proposed and known workarounds.

Security patches are often associated with the Common Vulnerabilities and Exposure numbers (CVEs), which is a unique identifier for a specific security vulnerability. This information is also provided by Microsoft, but on a separate page called the Security Update Guide [8]. The guide provides information about the CVE number, severity, and impact. Our cumulative patch is associated with 14 CVEs and the security-only patch addresses 9 CVEs and can be seen in Table 4.1

Patch awareness.

Both official and community curated, annotated lists of the newly available patches (c.f. Section 4.3.2) are quickly announced and shared through email lists and forums by community members, with citations and credit often given to the original authors/team. These announcements are important because they help admins prioritize their limited time and resources to focus on the patches that are most critical for their organizations. While summary information often has high overlap between sites, there are also distinctive attitudes and tones to each.

The Ghacks website, for example, focuses heavily on providing information about the content of the different patches providing, sometimes lengthy, change-log style descriptions of adaptions. For our patches, Ghacks only notes "Security updates to Windows Server, Microsoft JET Database Engine, Windows kernel-mode drivers, Common Log File System Driver, Microsoft Windows Search Component, and Volume

Manager Driver" [116]. Ghacks also provides an Excel spreadsheet for download with information about: product impacted, platform, link to KB article, type (cumulative, security, etc.), severity, type of vulnerability addressed, and CVE numbers. Because platforms, patches and CVE numbers have a many-to-many relationship, the August's Excel table had 712 rows for 35 unique KB numbers (patches), and 49 CVEs.

In contrast, ZDI's Patch Tuesday summary focuses on the CVEs being patched, including the number of security related patches, and the severity of the vulnerabilities addressed. For our patches, the author draws attention to two selected remote code execution vulnerabilities. Most notable is CVE-2017-8620, which essentially allows a malicious actor to commandeer a target system through construction of a malicious Server Message Block (SMB). It is also noted that this vulnerability is similar to a previously known bug which was exploited in the wild. The author therefore dubs this vulnerability to be "by far the most critical bug of the month". Following the in-depth look at particular vulnerabilities, a table of all the CVEs addressed in the patches is provided (c.f. Figure 4.2).

On AskWoody's forum, an admin creates a forum thread regarding August's patches entitled, "Lots and lots of patches" [122]. The post cites the Ghacks blog summary of the security patches released, and provides a snippet of the executive summary for the community members. Within 10 minutes, a member is quick to share the curated list provided by SANS.

On PatchManagement.org, a moderator sends an email out [123], reminding everyone that it is Patch Tuesday. Within this email, they provide a links to official KB articles from Microsoft. This thread also experiences a similar rapid response from the community members, with a user supplying the ZDI collated vulnerabilities list. Again, later that evening we see the SANS list shared on the email thread.

Update identification and prioritization

One of the first tasks an admin needs to accomplish on Patch Tuesday is to identify and prioritize the patches that they will need to install. The announcements and curated lists described above help by providing the necessary information in a single source. This is essential to reduce time spent information seeking, as patch information can be widely dispersed across a number of sources [28].

The AskWoody's announcement thread for that month's patches is also quickly used to discuss triage-type issues such as number of patches, the severity, the systems impacted, and the number and type of vulnerabilities. Such centralised information

CVE	Title	Severity	Public	Exploited	XI - Latest	XI - Older
CVE-2017-8620	Windows Search Remote Code Execution Vulnerability	Critical	Yes	No	1	1
CVE-2017-8627	Windows Subsystem for Linux Denial of Service Vulnerability	Important	Yes	No	3	N/A
CVE-2017-8633	Windows Error Reporting Elevation of Privilege Vulnerability	Important	Yes	No	1	1
CVE-2017-0250	Microsoft JET Database Engine Remote Code Execution Vulnerability	Critical	No	No	3	3
CVE-2017-0293	Windows PDF Remote Code Execution Vulnerability	Critical	No	No	2	2
CVE-2017-8591	Windows IME Remote Code Execution Vulnerability	Critical	No	No	2	2
CVE-2017-8622	Windows Subsystem for Linux Elevation of Privilege Vulnerability	Critical	No	No	3	N/A
CVE-2017-8634	Scripting Engine Memory Corruption Vulnerability	Critical	No	No	1	N/A

Figure 4.2: Exert of Zero Day Initiative's (ZDI) curated patch blog post showing a list of addressed CVEs.

aggregation allows experts to identify and elaborate on patches which are particularly security critical. In essence, this advice informs sysadmins of the most pressing security risks addressed in the patch release.

Digging into our two patches a bit more, a sysadmin would learn that the patches correct a total of nine unique vulnerabilities, with the cumulative patch correcting an additional five vulnerabilities relating to Internet Explorer, as detailed in Table 4.1. As the table shows, they range from moderate to critical in nature, as described by Microsoft themselves. The aforementioned "most critical bug of the month" is contained within both of our patches, likely resulting in the patch rising to the top of sysadmins' lists of patches to apply for the respective Windows versions.

4.4.2 "All code is guilty, until proven innocent"

Self reporting initial tests

One of the well documented reasons sysadmins delay or avoid patching is the risk that the patch may introduce new bugs into the system [13, 12, 42]. There are two common ways of mitigating the risk: 1) test the patches to find the ones with problematic errors, 2) wait for others to find the errors.

Sysadmins who have test environments, and consequently lower risk, install patches in those environments and report findings on forums and email lists. Similarly, sysadmins who perceive either the risk of not patching to be quite high, or the risk of problems quite low, will patch immediately and report the results. Looking at the

AskWoody's forum thread [122], users begin to share initial test results, while stating their patching philosophy (here: Group A), the exact system used, and the exact patch installed:

> "Group A, Win 7, SP1, X64 Home Premium installed [Cumulative Update KB]. No problems so far. I only use this computer for print, email, internet."

In a more comprehensive post the following day, 9th notes an unexpected change to their settings:

> "Enabled and started Windows Update on my Win 7 virtual machine. It ran a couple of minutes and reported 2 important and 2 optional updates available... I chose to hide the recurring optional KB2952664 "telemetry" update again.
>
> [Two screenshots of the patching UI.]
>
> The updates went in smoothly, the reboot was clean, no new errors or warnings in the System Event Log. A check for changes: BITS service was changed from DEMAND_START to AUTO_START. Further testing is planned."

A later poster asks if the setting changes caused issues, but is told by a third member that the change is unlikely to cause issues and is easily reversed. On this particular thread, no community member identifies potential warning signs regarding our updates.

Discovery of errors and issues

When sysadmins do run into issues, posters tend to report it to places like PatchManagement.org or Woody's Forums by creating a new thread, with a heading along the lines of: "I applied patches X and Y happened. My system is running Z etc. Is this happening to anyone else?" [87]. Each thread tends to contain key contextual data, including which patch was installed, what platform it was installed on, and exactly what problem is being observed or discussed. The posts are requests to the various communities to help troubleshoot issues. The bug reporting behaviours appear to happen simultaneously in multiple corners of the Internet. They are also not limited to sysadmins, many end-users also report on the outcome of installed patches, particularly if there are problems that they need help to correct.

For example, one of the earliest indications that our patches have issues is a post to a support forum for IfranView [124], which is a a freeware graphic viewer and editor

for Microsoft Windows. Similar to what we see elsewhere, the user creates a new thread, in which they report their issues. Posted on 10.08.2017 at 05:42pm, it states:

> "I'd like to report a bug, I have encountered with the 32-bit version 4.37 and the 64-bit version 4.44. It was encountered directly after installing Windows update [Cumulative Update KB], and solved immediately after de-installing said update.
>
> Computer data: OS: Win7 Home Premium SP1, 64 bits, i3 core, 16gb ram, Video Card: AMD Radeon R7 200 Series
>
> The Bug: Going fullscreen on my 2nd monitor causes the images to lag. Meaning if I go to the next image the old image remains on screen. In the case of a slide show, the screen remains black, but images are recovered after dragging an alternate window across it, and in its trail the original image temporarily returns but never fully ..."

In essence, the application of the cumulative patch creates a rendering bug on applications displayed on a 2^{nd} monitor, which, although not debilitating, could easily interrupt or restrict the work-flow of end-users.

4.4.3 Understanding Patches and their Errors

Synthesising similar bug reports

Key members of the community keep tabs on patch-related bug reports, even when they are posted on non-patching related forums, like in the IfranView example above. Those with many years of experience seem to have become patch soothsayers, picking up on subtle vibrations from the Web, which they then synthesize into analysis blogs and forum posts for consumption by the community. These posts consolidate the disparate information which is being posted all across the Internet, and put it into a single comprehensive form. Interestingly, these posts also make heavy use of links and attribution, citing the bug reports that lead to that conclusion and, where appropriate, acknowledging contributors by name.

For our patches, we first see this type of summary post on Günter Born's blog [125], on the 12^{th}, 4 days following Patch Tuesday. Born says that his interest was peaked through a post on Heise Online's forums [126], a German based media company focused on the IT sector. Born provides links to all of his gathered evidence, and in the case of the German language posts, provides translations into English. Born's investigative work is impressive in that he finds four other similar issues, and his citations include Matlab forums [127], TechNet [128], Nvividia forums [129], and a singular

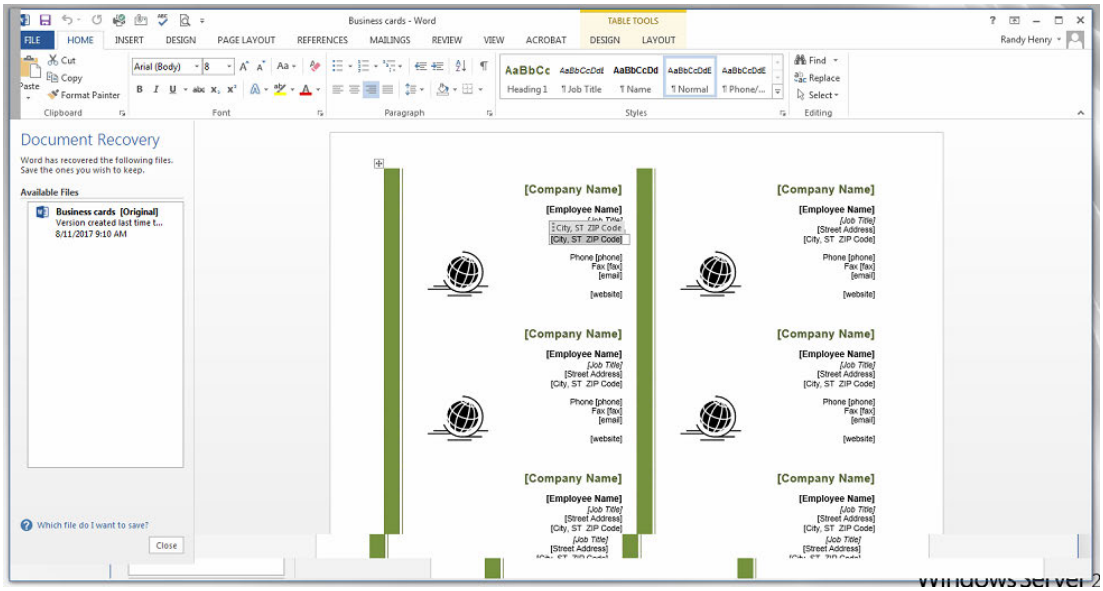

Figure 4.3: TechNet Post showing the rendering issue at the bottom of the image.

comment on a news post on Softpedia [130]. These cited posts are all similar to the IfranView post above, with varying degrees of information provided. The TechNet post in particular provides a large amount of detail supplemented with screenshots of the bug, as seen in Figure 4.3.

After highlighting the evidence that this is not a one-off event, Born provides known workarounds, which were provided by the users of the Heise forums. The first is to uninstall the guilty patches, which removes the issues, but also results in the system remaining unpatched, and therefore insecure. The second is to enable Desktop Composition, and Born provides a link to an online tutorial [131] which explains how to do so. Although Born posts this on the weekend, the work is noticed by an AskWoody admin, who reacts to the blog by creating a forum thread [132]. They begin the thread by immediately citing and thanking Born, before inviting discussion.

Proof of concept code

By the 14th, we see that another community member has provided a link to their blog post [133] on the AskWoody topic thread. Within the personal blog, they detail all known workarounds, which includes upgrading to Windows 10. The important contribution of the post is the inclusion of proof-of-concept code, which replicates the render-bug issue. This post is immediately shown appreciation from AskWoody members, and a moderator makes an announcement through their ComputerWorld column [134].

These posts act as a community wide summation of the current state of knowledge produced from contributions of many differing communities and their members. The work of others is fully acknowledged, with citations, links, and thanks given to all key contributors, including the summation posters themselves. The posts also provide a quick description of the role that screen coordinates play in creating the bug, although the solution advocated by the author is to uninstall both of the offending patches.

4.4.4 Waiting on Microsoft

Following this summary and background work, a possible option is waiting for an official fix from Microsoft. Enough work and information has been gathered to decide that the issue is related directly to our patches. Therefore, sysadmins and users can now justify waiting for an official response from Microsoft. However this may not be suitable for all, leaving some admins in a bind. Should they push these patches, which are known to cause issues, potentially increasing workload by manually applying workarounds such as screen coordinate adjustment, and impacting the workflow end-users? Or should they avoid patching for now and mitigate through other means? Remember, our patches fix a vulnerability which was seen as the most critical of the month. This is where the unique make-up of each sysadmin's context will contribute to their decision process.

Each of the forum posts continue to have some level of interaction as time passes, mostly confirming that the issue persists for them and providing more data-points and observations. On the TechNet forum post mentioned earlier, which is an official forum and channel to Microsoft, many users indicate that the issues are not unique to the original poster and the amount of users and activity on the post gives an indicator to Microsoft that it must react and address the issue. This is a logical approach for Microsoft, as it has been argued that online reviews and comments could be used to ascertain software quality [135].

4.4.5 Hotfix spotted Saturday

Monitoring developments, or lack thereof, is part and parcel for patch management. Given the security implications, sysadmins are expected to react to any developments and must be able to adapt. Moreover, their work does not finish at the weekend as systems can be attacked at anytime, and malicious agents do not seem to take vacations. Therefore, it is clear that community members must remain aware of related threads

and responses from stakeholders involved.

Two weeks later, and 18 days since the original release date, we see a response from Microsoft in the form of hotfix, KB4039884 [136]. An anonymous poster comments on the AskWoody's thread [132], on the 26th, at 3:08 pm:

> "Microsoft has acknowledged the bug in their Known Issues for August and releases a patch! (Not yet tested since I could never reproduce the issue...but can collaborate with co-worker on Monday.)"

This post happens on a Saturday, but spurs the AskWoody community into action immediately. A separate thread is created [137], detailing the hotfix's release, and allowing for space to host detective work. Moreover, Born also notices this and, although slightly behind the AskWoody community, creates a blog post [138] on the following day. However, all is not well as by Sunday morning it becomes apparent that there are issues with this hotfix, with one community member posting:

> "FWIW I installed the said 'fix' and then Windows Update came back and told me I had over 20 important updates and a few recommended ones that needed to be installed. Apparently some of the files that are installed are rather old versions (dating back to aug of 2016), just a few are actually newer than 20 aug 17 (according to autoruns). It does change many dll's [Dynamic Link Libraries]. (autoruns' 'compare' facility is great!)"

In summary, applying the official hotfix had reverted the system's Dynamic Link Library (DLL) files to previous versions, making them out-of-date and potentially reintroducing previously-fixed vulnerabilities back into the "patched" system. This observation was quickly verified by another member of the community, providing further evidence in the form of screenshots. The solution suggested is to uninstall the patch. It should be noted that all of this activity and detective work takes place before the official documentation appears on Microsoft Support, and only once the weekend is over do we begin to see the official KB article and related documentation being circulated through PatchManagment.org, through an announcement email similar to those seen at the beginning of this patch cycle [139] on the 28th.

A moderator summarises the AskWoody community's findings in a column for ComputerWorld [140]. They are quick to alert readers of the unsafe nature of the hotfix, using direct quotes from the forum thread created to discuss the hotfix using the post highlighted earlier.

4.4.6 Hotfix Vanishes: "WTH..."

At some point during the night and into the early hours of the 29th, Microsoft removes the patch from the Update Catalogue, but the KB article remains on Support. Effectively, the associated information regarding the patch remains, however when users attempt to download the patch they are informed that it does not exist. This results in a release of frustration and resentment across all forum and email threads. One example from PatchManagement.org thread on the hotfix release is:

> "WTH.... I just imported it into WSUS last night and added it to my Software Update Groups. I guess I'd like to know if it breaks something else now....."

Betrayal of trust

It appears that the unique relationship between a software vendor and sysadmin hinges on a complex form of trust. Sysadmins will find themselves in situations where they must wait for the vendor's response to an issue, so information that is provided is essential in their decision process. Be it data points on uneventful testing, or the detective work on what exactly a patch does, information is king. It is therefore no surprise that the removal of a patch by a vendor without informing sysadmins is tantamount to betrayal. A more detailed response, which epitomizes this dynamic is given by a furious poster on PatchManagement.org:

> "Is anyone else disgusted with this policy that Microsoft eventually scrubs its bad patches from existence? It is like a great cover-up every time they make a mistake. This does not help me as a customer that is looking at a patch wondering if I am supposed to be using it to fix a problem. If I did this at work or with my dealings with the government I would be fired or in jail. A better and more respectful solution would be to state the error, and elaborate on the plan going forward."

4.4.7 What should we do now?

With the removal of the "solution", we now find the community of sysadmins adapting to the new landscape. Given the security critical nature, sysadmins must be responsive, similar to the flexible and reactive nature of their security counterparts [18, 53]. For those sysadmins who were not aware of the hotfix, or were unable to catch it before it vanished, not much has changed. However, for those with access to the hotfix, they can choose to uninstall, again reintroducing the second monitor errors, or stick with

the hotfix. As we have discussed, the hotfix itself is potentially security threatening, but one could sink time into updating all the DLL files to remove the reintroduced vulnerabilities. The discussion begins to contribute towards which solution is best at the given time. One poster is adamant that the workarounds suggested are not realistic:

> " >The monitor issue is easily resolved by reordering the
> >monitors in the "normal" order, you don't need a patch to resolve
> >this problem.
>
> Spoken like a true geek with years of experience. Most home users wouldn't have easily thought of that solution. I know this to be true, because I have done desktop support for a very long time. And by the way, a patch is what broke it; why shouldn't the user expect a patch to fix it?"

A moderator of AskWoody also announces the removal of the hotfix through their ComputerWorld Article [141]. This article includes further quotes from forum users that detail findings from independent investigations, again illuminating the problems with the original hotfix.

> "I tested [Hotfix KB Number] in a Windows 7 x64 virtual machine last updated in Sept. 2016. [...] It replaces some files with older versions ... for some files [Hotfix KB Number] installs GDR versions of files, replacing existing LDR versions of files. The good news is that uninstalling [Hotfix KB Number] seems to undo these issues, and I recommend doing so if you already installed [Hotfix KB Number]. Some of the old file versions that are installed by [Hotfix KB Number] likely have security vulnerabilities that were fixed in newer versions. Thus, installing [Hotfix KB Number] probably exposes your computer to fixed security issues."

The lack of clear communication as to why the patch was pulled left many other online communities in the dark, with the reason only being revealed once the article is shared, as is the case with PatchManagement.org [139] where a member shares the contextual information. Hence, some communities become aware of just what is going on thanks to the work of the communities they are networked with.

4.4.8 "It's baaaaack."

The Hotfix is then re-released on the 30th, and the communities, again, rapidly notice and inform their respective members. On PatchManagement, a member [139] shares the re-released patch:

> "This update is now back, now dated August 30. And I'm seeing it in the catalog dated the 30th.
>
> Definitely need a tad more communication."

An AskWoody forum thread is created for discussion for this new version of the hotfix [142], which quickly descends into further detective work and discussion of Microsoft's communication policy. Many commentators note that the only hint of a previous version from Microsoft is that the KB article contains the instruction:

> "Before you install this update, you must uninstall any previous version of [Hotfix KB Number]. Then install [Cumulative KB Number] or [Security-only KB Number]KB4034679 before installing this current update of [Hotfix KB Number]."

4.4.9 Safe to patch . . . carefully

As discussed in Beattie et al. [13], as time passes, the probability of a patch causing a business essential service to fail reduces, while at the same time, the risk of attack through a vulnerability increases. Therefore, those admins who wish to avoid errors are justified in waiting for the dust to settle. In fact, Beattie et al. showed that the majority of faults in patches are found within the first 30 days, justifying this risk adverse behaviour of waiting to patch.

Since we now have a working patch, all parties should be satisfied. However, it is not until the 5th of September that we see an "all-clear" message from the AskWoody moderator on ComputerWorld [143]. This is four weeks following our chosen Patch Tuesday, and a week before September's official Patch Tuesday. The article summarizes the issues of the month, and breaks down update guidance by platform type. An interesting point is the continual focus on addressing those who wish to avoid Microsoft's "snooping" and the author's insistence to turn automatic updates off.

Through our digital ethnography observation of online information about a pair of Microsoft patches we show the different types of information available to admins and how that information evolves over time. We also discuss the role the community plays in both providing information to admins as well as generating information.

4.5 Discussion

4.5.1 Patch Information Evolves

System administrators consistently report using online information sources to gather information about their systems and to help troubleshoot unexpected system states [5, 28]. Prior work has also shown that they share patching information with each other

online [87]. In this work we dive into how the online information available changes as time progresses. We also highlight that the sources of this information are quite varied and shift over time. Initial information was provided by Microsoft, followed by annotated versions being provided by other trusted groups like Ghacks. But then we see information start flowing from a myriad of untrusted sources, namely people on web forums. This information might be suspect, but instead of ignoring it we see members of the community working to explain, verify, and find workarounds for what is being observed. These observations are then compiled and fed upward via comprehensive posts by more trusted community members till the issues become well enough accepted that Microsoft itself responds by posting a Hotfix to address the issue.

What is notable here is the location and flow of information that is occurring. With more official sources like Microsoft and trusted organizations providing timely information on patch release, but then being slower to provide updates about potential or confirmed issues with the patches. To fill the information gap, smaller forum- and email-based communities step in to report potential issues as they happen. It is well accepted that patching quickly is the best approach from a security perspective [43, 144]. But looking at the reality of online information, it would be extremely difficult for an admin to reliably learn about issues from online sources at the time of patch release or really even a few days later. Our patches required 18 days between patch release and Microsoft officially acknowledging the problem by releasing a Hotfix, which turns out to have security issues itself. So an admin who is following only the official sources might be quite rational information-wise in patching a month behind schedule. Patching any earlier would require a comprehensive testing infrastructure, or the willingness to spend a great deal of time reading forum posts to understand the problems. And even if an admin decided to take the risks and just patch at the time of patch release, they still risk having to spend a great deal of time online to troubleshoot issues and find workarounds as this information exists only in the community curated posts, not the larger official ones.

4.5.2 Communities and Networks of Practice

This case study further highlights the role Virtual Communities of Practice play in patch management through facilitating troubleshooting and issue verification between members. They also provide an informal central repository of the current community-generated status of each patch.

Communities of practice [25] traditionally focus on people who interact regularly to learn together about a shared domain, which can range from producing new art to crisis management. Such communities are also found in online virtual spaces (e.g. [145, 146, 87]).

In this case we further observe that patch management information seems to be built on a network of practice, rather than a single community. Many of the key network members, such as Woody, clearly belong to multiple communities and the communities actively reference work from each other. This indicates that these Virtual Communities of Practice (VCoP) can be understood as the constituent parts of a larger, loosely connected *Network of Practice* [27]. The links within this network are established through fully attributed information that is taken, constructed, reported, and explained uniquely within each respective VCoP. The final judgement, to patch or not to patch, slowly filters through each community to give the entire network much needed insights.

4.5.3 Trust in my Community

Our findings demonstrate the complex trust dynamic which exists in patching between the consumer and the vendor. While VCoP often include representatives of the vendor, our case study has shown why sysadmins may rightly be wary of trusting them. Therefore, they need other trustworthy sources [147, 148], which they can access through the Network of Practice.

Trust is a key aspect of sysadmin collaborative work [5, p.197-227]. It is clear from the responses following the pulling of the patch that many sysadmins do not trust Microsoft. This erosion of trust has resulted from past bad experiences which have been discussed and shared within the community, and are archived on their respective sites. The effect of this breakdown of trust echoes the findings of Vaniea et al. [75] and Mathur et al. [82] with end users, who take into account their relationship with a vendor and past bad experiences when deciding if they should update.

The observed Communities of Practice garner their community member's trust, similar to what others have observed [149]. Through access to the experiences and knowledge of others who are familiar with the particular pressures and challenges that sysadmins face, members are mentored and guided through an uncertain patching landscape. Consistent access to useful information makes individual Virtual Communities of Practice credible and trustworthy. This stands in contrast to vendors with the lack

of consistent patch information and standards [30, 150, 151].

4.5.4 Community Influencers

Of particular importance in establishing credibility are social influencers, or big names and moderators, within these communities. These people build credibility by providing summations of the work of others and provide unsure sysadmins with clear patching targets. This parallels the findings of Das et al. [152], who showed that social influencers can directly impact the adoption of the security features found on Facebook. Additionally, the work with software developers by Xiao et al. [153] shows that security recommendations from sources with high Internet reputation are viewed as being nearly as trustworthy as that of their peers.

The social nature of these Virtual Communities of Practice facilitates critical discussion of all information. If a community member contributes incorrect information then the 'many eyes' of the community will be able to spot and correct this mistake or misunderstanding, or at least ask for clarification. Therefore the community is quick to correct any knowledge which may be harmful or incorrect, which makes it more trustworthy. Essentially, the advice given, especially when issues have begun to surface, should not be considered as the advice of a single unknown individual, but the collective knowledge of many committed community members all seeking the highest quality guidance and assurances regarding their own patching decisions.

Additionally, similar to the work regarding Developers and Stack Overflow [154], we find that much of this work is facilitated through a smaller group of key members who routinely provide summaries or condensed lists of instructions following work by their respective communities. These community members could therefore be seen as informal Security Champions [155, 156] who encourage admins to push or avoid patches depending on the information gathered through out the community. Our case study highlights that these summations happen at key times with the flow of information being sourced, shared, and cited from one community to another, allowing for one community to greatly benefit from the work of another. It is clear that their efforts are appreciated within these communities and provide vital data which allows for lessons on the practice of patching to be widely accessible. Future research could be conducted to identify the motivations of these informal security champions, and identify and design systems which support these members and assist them in providing expert knowledge to numerous "apprentice" experienced sysadmins.

4.5.5 Known Issues are Dynamic

Our case demonstrates the dynamic nature of patch information, particularly the development of crucial information such as Known Issues [30]. We see that the information sources that sysadmins use are built upon the collaborative effort of community members, who aid in identifying critical updates and share patch quality data. Although our case may not be representative, the case lasted nearly a full patch cycle (4 weeks) which is close to the 30 days found by Beattie et al. [13] to be an optimal time to patch. One reason patching may take time, is linked to the time it takes for further issues to be identified, investigated, and finally remedied by the vendor. Individual communities have evolved unique practices for sharing and generating information, which then spreads through out the network of practice, and pushes vendors to respond. Future work should be aim to support community efforts by designing platforms to facilitate this collaborative behaviour in finding and fixing patch issues. Working with these communities can direct attention towards issues which affect the wider patching community reducing potential waiting times. Similar approaches of using user reviews to promote updates to end-users have been investigated [84] and has potential within the patching context for sysadmins.

4.6 Conclusion

This Chapter represents an in-depth descriptive case study of two security critical patches from Microsoft. We found that a number of communities, bloggers, and social influencers work in-conjunction to provide details regarding updates, including detailed proof-of-concept code of errors, and discussion on suggested workarounds and their suitability. We highlight that many members put considerable levels of effort in providing this information, and that distinct communities compliment and cite this work to provide high level guidance to their own members. Finally, our work highlights the level of effort and time it takes for the risks associated to a single pair of patches to be discovered and shared through out the wider community, representing a Network of Practice.

Chapter 5

It's hard. It's not "just do it.": Deciding to Patch in Context

5.1 Introduction

Information is not the only thing admins consider when deciding to patch or not, as they consider their wider context, policies they work under, and resources available to them. We have limited understanding of how these circumstances are factored into their patching process [29, 28].

The results from the previous two chapters come exclusively from observational data from a patch orientated mailing list (Chapter 3) and dedicated forums (Chapter 4, and therefore we wish to validate our results through self-reported data directly from sysadmins. More specifically, our observations are driven from active members of the communities that were studied online, and are therefore based on an active subsection of the entire community. By using a survey we can reach a wider audience of sysadmins in an attempt to confirm and generalize our findings.

In particular, Chapter 3 highlighted the themes and prominence of patching related information shared within PatchManagement.org, with the majority of emails being related to "Errors and Troubleshooting", indicating that admins' information gathering needs extends beyond solely for awareness of a patch's existence and that they may reach out for additional help if and when an error is found. Furthermore, Chapter 4 detailed the online information life cycle regarding security related patches, with community members continually sharing the results of testing conducted. Unlike previous research [28] which implied the existence of dedicated testing environments, our observations raised questions regarding the prevalence of informal approaches. We observed

admins reporting test results from ad-hoc tests on personal machines or gauging patch quality through use of the community's consensus and feedback, potentially impacting their decisions to patch. Furthermore, Velasquez and Durcikova's findings state that the need for verification information is significantly impacted by the task complexity [61] and our previous results highlight the convoluted nature of patch risk information and its development online. Clearly the unintended issues introduced by some patches drive community engagement to collectively troubleshoot and resolve, indicating that these online communities may provide patching related information necessary for post-deployment issues. The systematic literature review of patch management research conducted by Dissanayake et al. [44] adds further justification for extending the scope for sysadmins' information gathering needs in these latter stages, highlighting related socio-technical issues such as "post-deployment patch verification" challenges and "patch testing" challenges.

To address the aforementioned gap in research on sysadmins' information seeking behaviours, I detail the design and findings of an online survey which is intended to elaborate on and provide further details regarding the established patching process used by sysadmins found by Li et al. and validated by Tiefenau et al. [28, 29]. The survey aimed to extend our understanding of what testing environments sysadmins are using, and to explore whether our communities and their collaborative information gathering practices are more influential in Small or Medium-sized Enterprises (SMEs) when compared to large organizations. Previous work attempted to investigate the impact the size and of organization, stating that larger organizations would potentially have greater available resources for their patch testing needs, and may also have more stringent and mature patching policies [29, 28]. Specifically, this chapter aims to answer the following research questions, as listed in Chapter 1:

RQ5 What contexts do the system administrators studied work in? In particular: what types of systems do they administer, what size of organization do they work in, and finally what tools and resources do they have at their disposal?

RQ6 How often do sysadmins engage in the various commonly known patching behaviors that prior work has identified as being associated with the six patching stages?

RQ7 How does the context of a sysadmin (e.g. organization size, systems supported) impact their approaches to patching?

Patching Stage	Question Used
Awareness	When new patches become available, I learn about them through:
Prioritisation	When deciding which patches to prioritise for installation or testing, how often do you engage in each of the following?
Deciding	Who makes the final decision about installing, not installing, or waiting to install a patch?
	In your opinion, how much say do you have in if a patch will or will not be installed?
Preparation	When preparing to install patches on a system, how often would you do the the following actions?
Testing	When testing patches which of the following test setups do you use?
	What most motivated your testing setup
Installation	The deployment of patches is:
Post-Installation	Once a patch has been deployed, to validate that it is working as expected, I will:
	When you detect an error after testing or deployment, what actions would you perform?

Table 5.1: List of patching stages and the related survey questions. Each of the above questions was close-ended, using three formats: multiple choice (radiobox), multiple answer (checkbox), or a set of Likert options. When appropriate, an "other" option was also provided that allowed for free text entry.

5.2 Methodology

We constructed an online quantitative survey based on our observations regarding the use of online communities of sysadmins and behaviours identified by related work [28, 29, 30]. These common behaviours were then grouped into the related stages [28, 29] of the patching process to assess how widespread they are within the broader sysadmin community. A complete copy of the final survey apparatus used can be found in Appendix A

5.2.1 Survey Structure

We decided to use a survey methodological approach because previous research has indicated that sysadmins and other IT Security professionals have hectic working practices making them hard to source for studies [31, 32]. Therefore, our focus was to construct a survey that would take participants no longer than 10 minutes to complete. The majority of the questions were arranged by relevancy to the stages of the patching process found by related work [28, 29]. Where it was suitable, we used Likert scales to capture the respondents beliefs regarding how likely they engage in each stated behaviour during a typical patching schedule. Table 5.1 displays the stages and the related questions used.

Our survey started with demographics of their organization (e.g size, sector, number of other sysadmins working with). To ensure our survey was appropriate for as many sysadmins as possible, we included demographic options to cover those who

were currently unemployed or managed patching for multiple organizations (i.e. IT outsourcing companies). Once these initial organization demographics had been taken, we asked participants to select the patching process of the organization, system, and Operating System (OS) types they felt most confident in recalling, henceforth refereed to as the Baseline set-up. For respondents that were unemployed, we asked that they answer based on their most recent sysadmin job and with the outsourcing admins we asked they answer based on one of the organizations they supported. We then asked for more detailed demographics regarding their chosen organization such as the number and types of machines, OS types, and software components that they managed. They were then asked to focus on one specific system type, and answer the stage questions with respect to their patching process for that system. This was done to allow us to compare between these groups to investigate differences in their behaviours and reduce noise introduced by discussing specific machines as opposed to their full system. Once all the respondents had completed the questions on each stage, we ended our survey by collecting participant's demographics. To ensure validity we placed a simple attention check question that appeared in a random position within one of the radiobox questions. The questions simply asked respondents to select "Sometimes".

5.2.2 Participant Recruitment

To collect the widest sample of professional sysadmins we recruited through a number of channels known to have previously been successful in attaining participants [28, 29, 30]. This included using a number of relevant subreddits, such as r/sysadmin, r/windows, r/linuxadmin, r/windows10 and r/windows11. We actively sought approval from moderators from all subreddits used before posting our survey, preventing us from breaking any community norms. Additionally, we shared our survey with the moderators of PatchManagement.org, who circulated the survey through the mailing list. Finally, we posted our survey on Twitter and encouraged it to be shared by relevant popular sysadmin accounts. The survey was active for 2 full weeks, from the 24th of August to September 7th, with these dates being selected as it was outwith the patching cycle for Windows Patch Tuesday. It was believed that doing so would encourage greater engagement as sysadmins would not be currently engaged with patching Windows machines. During the time that the survey was live we received a total of 362 respondents began the survey, and was completed by 224 (61.5%). In total we had 220 valid responses for the survey once we removed any responses which failed the

attention check question. The median time taken to complete the survey was 8 minutes and 13 seconds.

5.2.3 Statistical Analysis

Comparisons between groups were made using scipy.stats package, version 1.3.1 [157]. We used the non-parametric Mann-Whitney tests to compare questions using Likert scores such as for sysadmins' reports of frequency of behaviours in the Awareness, Prioritisation, and Preparation stages of the patching process, as seen in Table 5.1. For Likert-based questions, which consisted of 7-9 statements, we used Bonferroni Correction to adjust for multiple comparisons. We will only report results which are statistically significant at the ($p < 0.05$) level, which corresponds on an uncorrected level of $p < 0.005$). To compare groups for questions which had allowed participants to select a single option, we used Chi-square tests.

5.3 RQ5: The Contexts in Which Sysadmins Work

Below we will detail the demographics of the participants, the organizations that they represent, and their working experience and contexts.

5.3.1 Participants

A summary of the participants demographics can be found in Table 5.2. The vast majority of our responses came from male sysadmins (195/220), with only 13 female, 2 non-binary individuals, and 1 participant choosing to self identify. The most common age band reported by sysadmins was 25-35 (71/220), followed closely by the age bands 35-45 (61/220) and 45-55 (40/220). The majority of participants had received some form of higher education with a total of 173 reporting at least attending college, with a Bachelor's degree being the most common qualification attained (86/220). The majority of our sample was from North America (125/220), with the two largest groups following this being the UK and Northern Ireland (34/220), and Europe (43/220). Our sample reports to be very experienced with the practice of system administration, as over half of our respondents reported having more than 11 years of experience (112/220), with 46 reporting between 6 and 10 years, and 36 reporting 3 to 5 years.

		N	% (2d.p.)
Age	18-25	20	9.13
	25-35	71	32.42
	35-45	61	27.85
	45-55	40	18.26
	55-65	11	5.02
	Over 65	4	1.83
	Prefer not to say	12	5.48
Gender	Male	190	86.76
	Female	13	5.94
	Non-Binary/ Third Gender	2	0.91
	Self Identifying	1	0.46
	Prefer not to say	13	5.94
Location of Work	North America	125	57.08
	UK and Northern Ireland	34	15.53
	Europe	43	19.63
	Asia	2	0.91
	Oceania	8	3.65
	Central America	2	0.91
	Prefer not to say	5	2.28
Education	Secondary School or Less	1	0.46
	High School	24	10.96
	College but no degree	55	25.11
	Bachelor's	86	39.27
	Master's	29	13.24
	PhD	3	1.37
	Other	15	6.85
	Prefer not to say	6	2.74
Sysadmin Experience	$<$ 1 year	3	1.36
	1-2 years	14	6.36
	3-5 years	36	16.36
	6-10 years	46	20.91
	11+ years	112	50.91
	Prefer not to say	9	4.91

Table 5.2: Table summarising details of respondents. N=220

		N	% (2d.p.)
No. of Organizations	Single Organization	168	76.36
	2-5 Orgs.	21	9.55
	6-10 Orgs.	8	3.64
	11+ Orgs.	21	9.55
	Unemployed	2	0.91
Organization's Sector	Capital Goods	4	1.82
	Commercial & Professional Services	17	7.73
	Communication Services	7	3.18
	Consumer Staples	15	6.82
	Education	31	14.09
	Energy	4	1.82
	Financial	17	7.73
	Government	28	12.73
	Healthcare	20	9.09
	Manufacturing	13	5.91
	Other	18	8.18
	Technology	36	16.36
	Transportation	6	2.73
	Prefer not to say	4	1.82
Organization Size	$<$10 employees	13	5.91
	10-49 employees	23	10.45
	50-249 employees	46	20.91
	$\geq$ 250 employees	138	62.73

Table 5.3: Table summarising details of respondent's organizations. N=220

5.3.1.1 Organizations

We have provided a summary of the organizations that admins work for in Table 5.3. The overwhelming majority of sysadmins reported managing the patches for a single organization (N=168), however 50 respondents (22.7%) reported patching for a number of organizations, which we believe represents employment at an IT outsourcing company. 4 respondents were currently unemployed at the time of the survey. Just under two thirds of organizations had over 250 employees (N=138; 62.7%), and 46 reported supporting 50 to 249 employees. A range of sectors were reported with the most common being technology (N=36), education (N=31), government (N=28), and healthcare (N=20).

		N	% (2d.p.)
No. Sysadmins work with	0	31	14.09
	1-5	120	54.55
	6-10	26	11.82
	>10	43	19.55
No. of Machines	1-100	38	17.27
	101-250	32	14.55
	251-500	41	18.64
	501-1000	27	12.27
	1000+	82	37.27
Machine Type	Client Machines (Laptops, Desktops etc.)	173	78.64
	Servers	207	94.09
	Mobile Devices (Phones & Tablets)	82	37.27
	Routers/Network Appliances	133	60.45
	Embedded devices/Internet of Things	54	24.55
	Other	12	5.45
Software Managed	Operating System	216	98.18
	Applications	197	89.55
	Custom/Bespoke inhouse programs	79	35.91
	Software that is no longer supported by Vendor	61	27.73
	Other	7	3.18
Operating System Managed	Mac	57	25.91
	Windows	200	90.91
	Linux	138	62.73
	iOS	63	28.64
	Android	55	25
	ChromeOS	7	3.18
	Other	26	11.81

Table 5.4: Table summarises the working context of respondents. N=220

5.3.1.2 Managed Machines and Software

A summarization of the working contexts of the respondents is shown in Table 5.4. The majority of sysadmins reported working with colleagues, with only 31 respondents being the sole admin for their organization. Additionally, just over a third of respondents reported managing over a thousand machines in their patching duties (N=82) Sysadmins in our survey managed a range of software types, with only 39 respondents reporting management of one distinct type of machines and the majority of these admins, 31 out of 39 (80%) dealt with patching of servers. Servers were the most popular managed system with 207 respondents reporting this, with client machines the next largest group (N=173). When asked to provide details regarding the types of software that they managed on their systems almost all admins reported OS (N=216), with Applications also receiving a large proportion of respondents (N=197). Interestingly, just over a quarter of respondents reported the existence of End-of-life software (i.e. software no longer supported by the vendor) on the machines they managed. The most popular OS managed was Windows (N=200), with Linux the next most popular with 138 respondents.

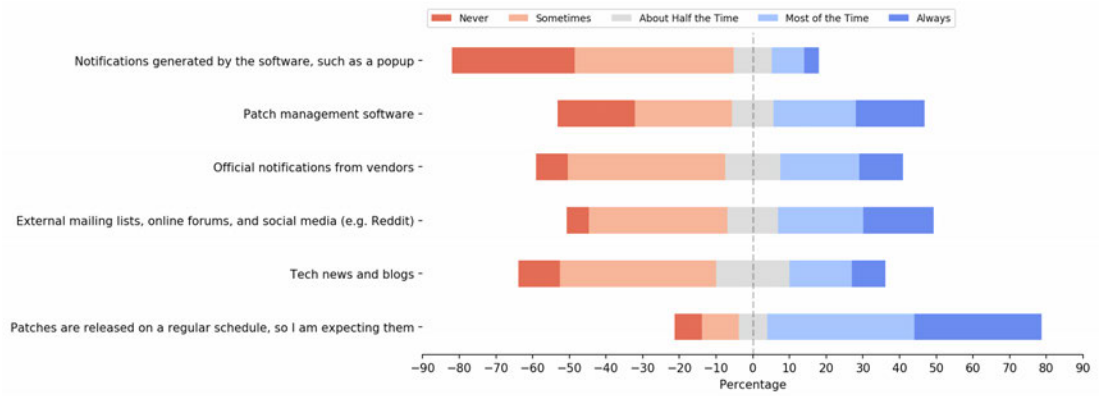

Figure 5.1: Patch Awareness response percentages N=219.

5.4 RQ6+7: Patching Behaviours and the effect of Organisation and Machine Type

To compare impact of organization size our sample was split into two groups; the first being Large organizations (i.e. over 250 employees; N=138) and Small and Medium enterprises (SMEs) were combined into a single group (N=82). Henceforth we will refer to these groupings as Large and SMEs respectively. To compare differences between systems we took the data from our Baseline question and compared the responses from Client machines (N=109) and Servers (N=104). Our results for each stage below will initially be discussed in terms of the full data set before detailing the aforementioned two splits.

5.4.1 Awareness

We begin by accessing the sources of patch awareness and respondents believed frequency of engagement during a typical patching cycle. Previous research [28, 29], including the work in the previous chapters(4, 3), has shown the breadth of sources used to become aware of a patch's existence. We asked our respondents to report the frequency of different potential sources of patch awareness using a Likert scale, the results are shown in Figure 5.1. The most frequent response from all respondents was the regularity with which patches are released with just under three quarters of stating most of the time or always (74.9%; N=164). Given that one of the largest software vendors, Microsoft, has a regular release point, Patch Tuesday, it is evident that admins have become accustomed to this schedule and therefore have built their processes around it. Another interesting response was that the majority of admins are not made

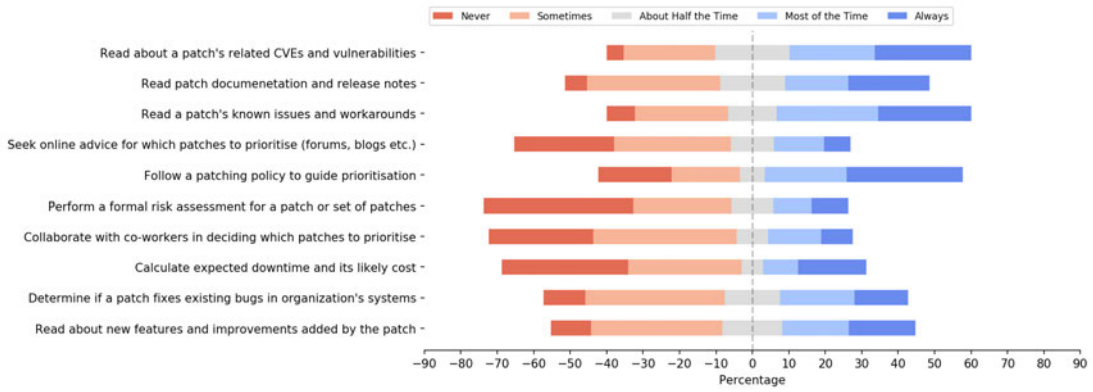

Figure 5.2: Patch Prioritization response percentages N=219

aware of patches' existence through the software itself, with 76.7% stating never or sometimes (N=168).

Organization There are very few differences between organisation size. Sysadmins for SMEs are more likely to become aware of patches through notifications from the software itself than sysadmins for Large organizations (Mann-Whitney U(137,82)=4259, $p<0.005$). Instead, sysadmins for Large organizations were more likely to notice patches that were released during regular patching schedules (Mann-Whitney U(137, 82)=4410.5, $p<0.005$). This suggests that Large organisation may have specific patching policies around key patch release dates, while sysadmins for SMEs are more sensitive to irregular patches such as hotfixes.

Machine Type We found no significant differences between Client and Server sysadmins when it came to awareness of patches.

5.4.2 Prioritization

Once aware of a patch, sysadmins must use a number of sources of information in order to identify the patches they must prioritise. To do so, they will look at sources externally in conjunction with their organizations needs and risk appetite. Hence our question was designed to capture this range of sources in a likert scale, and displayed in Figure 5.2. A majority of responses stated that they rarely perform formal risk assessments for patches, with 68.0% of all respondents (N=149). Additionally, admins reported that calculating potential downtime of systems was not a concern with around two thirds stating that it was an irregular occurrence (65.8%; N=144). Interestingly, the

prevalence of patching policies was lower than expected. Only half of all respondents (54.3% N=119) stated that policies played a role, whereas 38.8% (N=85) indicating that it did not. When it comes to integrating external information, 59.4% (N=130) respondents were rarely likely to prioritise based on guidance from social media or forums.

Organization When comparing the scores based on organization sizes, little differences were observed. However, the only example which showed some differences was that of the influence of patching policy, with a larger proportion of admins from Large organizations (60.6%; N=83) versus admins from SMEs (43.9%; N=36). Patching policy is more likely to affect patch management in Large organizations than in SMEs (Large median = 4 (often), SMEs median = 3 (half and half); Mann-Whitney $U(137,82)=4300$ $p<0.005$). Sysadmins for SMEs are somewhat more likely than those for Large organizations to factor in downtime of affected systems (Large median = 2, SMEs median = 2; Mann-Whitney $U(137,82)=4463.5$ $p<0.005$), but in both cases, they are relatively unlikely to do so (median answer for both 2, corresponding to sometime).

Machine Type There were two main differences between server and client sysadmins. 71.2% of all Server admins (N=74) did not rely on external sources to identify and aid in prioritisation, as opposed to 47.7& of Client admins (47.7% ; N=52). Admins of clients were also more amenable to seeking online advice regarding which patches to prioritise (Client median = 3 (half and half), Server median = 2 (rarely); Mann-Whitney $U(109,104)=3927.5$ $p<0.005$). This may be due to the popularity of the Windows OS making patching related information more accessible with user reports on social media or to the vendor directly. Furthermore, factors in decisions related to server patches appear to be more contextual with general server advice potentially being less applicable to themselves and their unique servers.

5.4.3 Decision

We were interested to see who and what was considered when making the final decision to install or delay a patch, hence we ask admins to select all options that were applicable. The overwhelming response from all admins was that it was themselves, or their team who made the final decisions with regards to a patch (78% ;N=173). It was believed that patching policies would dictate or guide decisions regarding patches

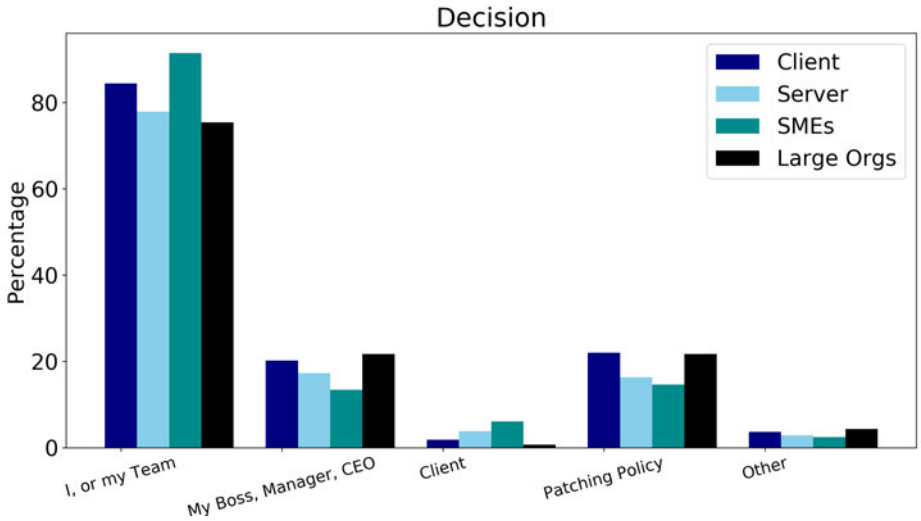

Figure 5.3: Patch Decisions input by response split

however, only 19% of admins reporting that a formal patching policy played any role in their decision process. Additionally, we found that only 18% reported having input from their boss, manager, Chief Information Security Officer (CISO), or executive from their organization. This was further compounded when asking respondents to gauge the amount of say they have in the decisions regarding a patch, with over two thirds of admins stating that they had a 'great deal' of input (N=151) and only 19 respondents stating they had little to no say in their organization's patching decisions. All responses are displayed in Figure 5.3.

Organization SME sysadmins stated that they or their team had responsibility for patching decisions (91.5% , N=75) as opposed to sysadmins in Large organizations (75.4% N=104), indicating that admins from SMEs may have more responsibility with regards to patching decision compared to their counterparts in Large organizations. Decisions in Large organizations appear to be only slightly more centralised and driven by policy: 21.7% (N=30) of sysadmins in large organizations reported an influence of their managers, compared to SMEs with 13.4% (N=11). Furthermore, 21.7% (N=30) of sysadmins for Large organizations identified a strong role for patching policy, compared to SMEs' 14.6% (N=12). Some SME sysadmins reported client influence on patching decisions (6.1%;N=5), while this was almost never the case in Large organizations (0.7%; N=1). There was no difference in sysadmins beliefs regarding the level of input into patching decisions between sysamdins of Large organizations and those of SMEs (Mann-Whitney $U(138, 82) = 5145$ $p > 0.05$).

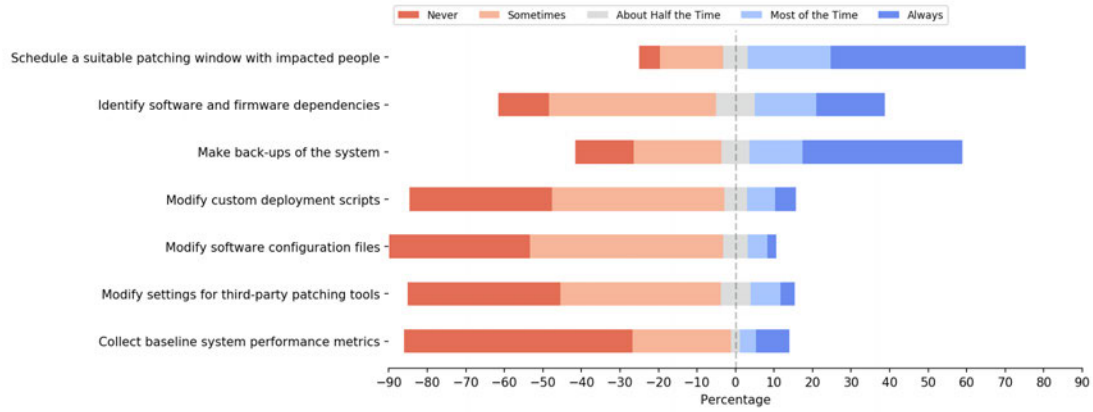

Figure 5.4: Patch Preparation response percentages N=219

Machine Type Similar to the previous split, the slightly more favoured responses from Clients and Servers was them or their team having the final say in patching decisions. Figure 5.3 shows that for all options available, percentages of Clients were slightly higher than Sever responses except for the 'Client' option, indicating that Servers may have more input regarding updates from clients and users than users of Client machines. This may be due to the fact that greater coordination to agree what and when these updates should be installed on Servers. Again, there was no difference in perceived input in patching decisions scores between that of Client sysadmins and Server sysadmins (Mann-Whitney $U(109,104) = 5244$ $p > 0.05$).

5.4.4 Preparation

We asked respondents to provide their frequency of actions related to the preparation stage, with the responses indicating that identifying a suitable patching window with impacted users was a regular occurrence with 72.1% of all responses (N=158). Interestingly however many of the other actions which involved changing or modifying components were considered rare events, with the majority of respondents selecting 'Never' or 'Sometimes'. This included the modification of deployment scripts (81.7%; N=179), configuration files (86.8%; N=190), or settings for third party patching tools (81.3%; N=178). Surprisingly, the action which appeared to split the respondents was the frequency of making back-ups of their systems, with only just over half of respondents indicating this was a regular occurrence (55.3%; N=121), as seen in Figure 5.4

Organization We compared our results based on the organization size, finding that there appeared to be very little differences between the frequency of reported actions

by admins from Large organizations and SMEs with their score distributions found not to be significantly different for any individual questions.

Machine Type When comparing our responses based on the machine types we observed a noticeable difference in responses for backing up the system before patching, with a larger proportion of Client admins reporting this to be a rare occurrence (50.5%; N=55) compared to Server admins who stated this was regular occurrence (66.3%; N=69). This was found to be significantly different with admins of clients (Median=2; Sometimes) backing up systems less often than admins working with servers (Median=3.5 ; Mann-Whitney U(109,104)= 4059.5, $p < 0.005$). This may be due to the fact that admins working with servers may have greater access than admins updating client machines, which may include work laptops taken to work from home, limiting direct access and leaving it up to the individual user.

5.4.5 Testing

Previous research [28, 29] and government guidance [3, 37] routinely states that testing patches before installation is best practice. This testing could come in the form of

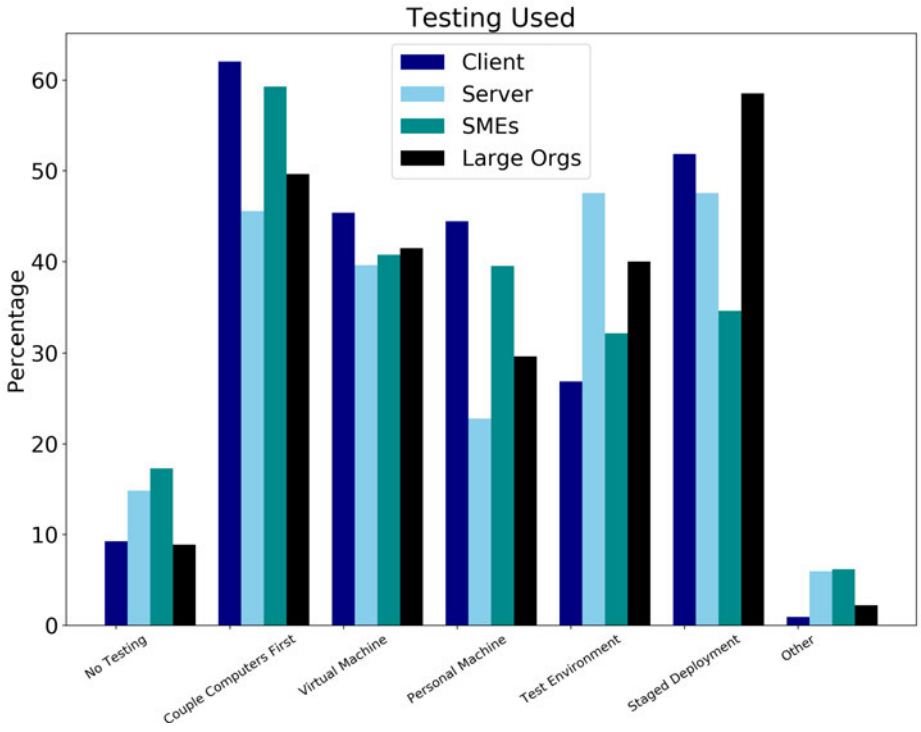

Figure 5.5: Testing set-up used by respondents percentages

staged deployment, or the use of a dedicated testing environment, however we explicitly asked admins to report any and all methods used. Our results indicate that around a half reported using staged deployment (N=107) as part of their testing process. Furthermore, dedicated test-environments are not as prominent as previously thought with just over a third of admins reporting access to one (37.2%; N=80) as part of their testing process. Overall, we found that only 12% (N=26) reported that no testing was involved. As a follow up question we asked admins to select a reason as to what motivated their testing set-up, with "Resource availability" being the most popular option with 26.8% of all admins (N=57). Additionally, 22.5% stated that their patching policies motivated (N=48) the set-up and 23.5% reported that they were motivated by the rarity of patches causing their organization errors.

Organizations When we compare the results based on organization size, see Figure 5.5, a greater proportion of admins from Large organizations (58.5%;N=79) report using staged deployment when compared to SMEs (34.6%;N=28). Another clear difference can be seen in motivations for testing set up, with a slightly greater percentage of admins of Large organizations stating that patching policies dictated their approach (28.4%;N=38) compared to SMEs (12.7%;N=10). Moreover, 31.6% (N=25) of SME admins reported the fact that patches rarely caused errors as a motivator compared to large organizations with only 18.7% (N=25). Organization size was found to influence in the reasons selected by admins to justify their chosen testing set up with a significant difference between selections made by admins of SMEs and those of Large organizations ($\chi^2(4, N=213) = 10.4, p < 0.05$).

Machine Type When we compare the reports by system, we see that there is a clear difference between Clients and Servers use of a dedicated test environment with the latter only reporting around 27% (N=29) compared to almost half of Servers (47.5%;N=48). An interesting observation is that their is only a slight difference in the reports of no testing taking place were higher for Servers (14.9%;N=15) than Clients with only 9.3% (N=10). The machine type administered by admins did not differ in motivations selected choosing for their chosen testing set-up was found not to exist ($\chi^2(4, N=207) = 2.2, p > 0.5$). This result, coupled with the previous comparison of organizations size implies that reasons for admins selecting their testing set-up is influenced more by the organization size rather than machine administered. This makes sense given the potential differences in budget and resources available to administrator and the formality of

their IT management processes.

5.4.6 Installation

We asked participants to estimate the level of automation they used in the deployment of patches with the vast majority of responses stating that this was "Mostly Automated" with 59.7% (N=129) of responses. The least popular option was "Fully Manual" with only 4.6% (N=10) of subjects selecting this option. This highlights the growing trend of automation, and the fact that respondents were less likely to select "Fully Automated" may not be fully applicable for the deployment of patches due to the need for input from a human expert decisions [5, 44].

Organizations Comparing the selection made by the organization splits we can see that both groups are overwhelmingly "Mostly Automated" with 67.4% (N=91) of Large organizations and 46.9% (N=38) of SMEs. However we can see slight differences across options as more admins from SMEs report that their deployment is "Roughly Same" with 19.8% (N=16)compared to only 4.4% (N=6) of Large organizations. It was found that there was a statistical differences between the level of automation in patch installation for Large and SMEs, however they both reported as mostly automated (Median for both was 4; Mann-Whitney $U(135,81) = 4280.5$, $p < 0.05$)

Machine Type Examination of the responses by the machine type being administered the same similarities were found, with both Clients and Servers overwhelmingly selecting "Mostly automated". When performing statistical comparisons based on the level of automation used in patch deployment, there was no statistical differences found between the admins of clients and respondents administering servers.

5.4.7 Post-Installation

The results in the previous Chapter, Chapter 4, highlighted that some members of these communities would remain vigilant to online sources regarding issues with patches, and also use these communities to aid in troubleshooting. Hence, we were interested in understanding how sysadmins judge that a deployed patch is performing as expected. The most popular response from respondents was to monitor user reports with 81.7% (N=174), and was closely followed by monitoring online sources for reported issues

with 68.5% (N=146) of responses. What is notable is that a lower proportion of the respondents were reliant on vendor actions such as announcements (45%;N=96), or changes in patch documentation (24.4%; N= 52). Our final question aimed to identify troubleshooting strategies once an error has been identified within their systems, with the most popular response being to investigate the cause and scope of the error with 79.3% (N=169). Checking patch documentation (69.5%; N=148) and reading or asking for help from online sources (69%;N=147) were also similarly popular actions. Interestingly, only 23% (N=49) of all responses reported immediately uninstalling the offending patch, as seen in Figure 5.6.

Organization When we compared responses across our organization groupings, we found that for the question regarding validating a patch was working, admins of SMEs and Large organizations were only slightly different regarding "Monitoring Vendor Announcements", with 41.5% (N=32) of SMEs syadmins compared to 47.0% (N=64)of admins from Large organizations. In the second question, there are noticeable differences, as raising a ticket with the vendor was reported by only 24.7% (N=19) of SMEs admins compared to 53.7%(N=72) of Large admins. Additionally, there was also very little difference between the proportion of SMEs admins (77.9%; N=60) reporting use of online forums for help and advice compared to sysadmins from large organizations (64.9%;N=87).

Machine Type When looking at systems there was also a discrepancy between admins of Clients (33.6%;N=36) and those of Servers, with only 51.0% (N=51) reporting the use of raising a ticket with the vendor, and asking colleagues for help with 37.4%(N=40) compared to 47%(N=46) respectively. A consistent difference between these groups found in both questions was the higher proportion of Client admins reporting the use of online sources to both monitoring for issues (76.6%, N=82 , versus 62.0%, N=62 , of Servers), and seek patching advice (74.5%, N=80, versus 66.0%, N=66, of Servers). These results indicate that there may be potential differences regarding machine type and use of online communities. Client admins appear to be more likely to look to these online communities for patch information regarding errors and post-installation issues compared to Server admins, and this is possibly due to the wider availability of general client machine advice which may be applicable to their client set-up, since a server's set-up and maintenance is likely more contextual than client desktop machines.

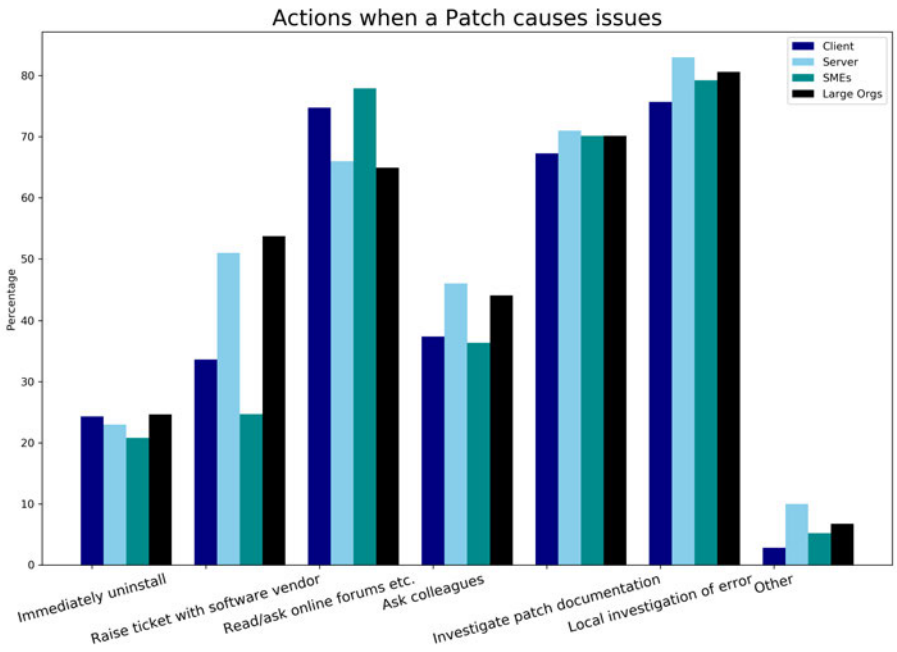

Figure 5.6: Actions performed by respondents when an error is caused by a patch.

5.5 Discussion

The aim of this survey was to determine the prevalence of patch management behaviours, both observed and reported in previous research [87, 28, 29], and expand upon areas within the patching process. To the best of our knowledge, we have procured the largest sample of sysadmins to date, and were able to investigate the testing set-ups used by admins, and their motivations for them. Additionally, we were able to expand upon the post patch deployment state, and identify how admins become aware of patch issues and errors. Our work highlights that the breadth of sources used to become aware of a patch's existence can also be reached and utilised in other stages of the patching process, particularly in troubleshooting errors where uninstalling is not the default it is believed to be.

Our results highlight a number of interesting findings, including the contexts in which administers work (RQ5). Sysadmins are responsible for the patching of a range of systems, and the applications that run on those systems. We found that a quarter of sysadmin surveyed must manage software that is no longer supported, adding further complexity to patch management. Our results highlight the prevalence of previously identified behaviours [29, 28] and observed behaviours [87] (RQ6), such as the testing set-ups used by admins, with dedicated testing environments and the use of staged

deployment less common than previously theorised. Additionally, we illustrated that contextual factors such as organizations size may lead to differences in approaches taken in the patching process, and in the level of input these admins will have in their respective organizations patching decision (RQ7). We also highlighted that admins of servers may only be slightly less likely to engage with advice shared online, indicating that server related decision may be more reliant on their unique context than the advice given regarding clients.

5.5.1 Information gathering does not stop at awareness

Previous work indicated that admins may struggle to coordinate their patching process due to the wide ranging number of sources of information, from internal systems, notifications, mailing lists, and news or tech blogs [28, 29]. Our survey, and the work in previous chapters (Chapters 3 4), highlights that sysadmins do not halt this information gathering process at the initial stages of a patch's release, and in fact the process of patching is an ongoing search for information. For example, we showed that admins will monitor online sources for hints and suggestions at errors within patches. Doing so will allow admins some understanding of the risk associated with a patch, since even if a patch passes all tests, or initially runs on a system with no issues, reports of errors elsewhere give a sense of the patches quality and may cause issues with their individual systems. Keeping on top of these potential issues prevents admins from being shocked or surprised when an error suddenly appears.

What is surprising however is that our results indicate that such integral information is rarely given from patch management software with admins reporting that they do not rely on such software for patch awareness. However, we could extend such systems, like Microsoft's System Center Configuration Manager (SCCM) to incorporate other sysadmins patch reviews, similar to the work of Tian et al. [84], allowing admins to identify which patches may be causing issues elsewhere. Additionally, it appears that admins may have adapted to this spread of information by forming their own online communities which through their collaborative information gathering efforts have essentially created a proxy centralised source for their patching needs. Additionally, we have seen a growth in sites orientated around aiding developers with their practice such as the Q&A site StackOverflow [34], which is known to influence the practices of developers. A similar site has been created for sysadmins called SuperUser [158], and future work should investigate these sites to better understand the issues that face

sysadmins outwith the patching process.

5.5.2 Testing may be ad-hoc

The testing of patches is a key stage in the patching process, with previous work indicating that use of staged deployment and dedicated test environments aid admins in identifying issues before they make their way into production systems. However, our work indicates that these formal processes may not be as wide spread as previously thought. Only 37% of respondents indicated that they had such resources available, with the majority of testing apparently taking a much more informal and ad-hoc approach. One reason for this approach may be the fact that having a separate testing environment which accurately imitates a real production system will only result in doubling the workload of admins as they must now maintain both distinct systems. Hence, the approaches taken by admins are streamlined, with many admins testing initially on their own personal machines or through the use of Virtual Machines. Hence future work should be conducted to investigate these informal testing practices which appear to be more prominent in SMEs. Additionally, it appears that there are some differences between the testing strategies used by admins for larger organizations who may have more resources available to allow for the use of dedicated testing environments and staged deployment techniques. Furthermore, we saw deviations in the reasoning behind the strategy used with a larger proportion of admins from large organizations identifying patching policies as the reason for their approach. This may be due to the fact that larger organization may have more matured procedures due to their size and availability of resources. Furthermore it may explain why we see SMEs having a larger percentage of responses stating that patches rarely cause errors as since they are smaller, they may have less complex systems with fewer additional programs and software, reducing the chances of potential conflict with newly introduced patches.

5.5.3 Uninstalling is not default

One of the main aims of this survey was to expand upon the latter stages of the patching process, and in particular expand upon the troubleshooting of errors. Previous studies identified that the most common response to a patch causing an error was to remove the offending patch [28, 29], however our results indicate that this may not be the most common response, with a large majority indicating that work is done to understand the scope and impact of an error, with additional help coming from online forums or

blogs. These strategies makes sense given the highly collaborative and problem solving nature of system administration [5], and to simply remove an error only delays an issue until the patch is fixed, often in the form of a hotfix (see Chapter 4) Additionally, this approach of carrying out one's own internal investigation before or while attempting to gain help from online communities could be seen as putting the correct effort in. With all sysadmins having their own systems to manage, any requests for help may be ignored if the request does not show adequate attempts to solve the issue themselves. Our results show that admins may raise tickets with the vendors to get them to fix the issue, since when an error occurs the fault can either lie with the patch or the admins specific system. What is interesting however is that limited number admins from SMEs reported raising a ticket with vendors, and this may be due to their scale, with only issues affecting larger organizations and therefore impacting more users and therefore more likely to receive a response. Future work should aim to identify and expand on these online troubleshooting strategies which appear be prominent throughout all groups investigated.

5.5.4 Limitations

The responses from the survey participants are self-reported, and hence may not be reflective of actual behaviours due to social desirability bias [159]. However, the survey was structured to ground participants by guiding respondents to reflect upon their patching process based on known stages [28, 29] and by providing a baseline question focused on a particular system (i.e Client, Server, Router etc.). It was believed that doing so would reduce the complexity when recalling the frequency of their patching actions as they would not need to involve the wide range of technologies they are tasked with managing [5]. Additionally, the channels for recruitment we used have been also been used in previous research [29, 28, 30], but to guard against potential invalid responses a simple attention check question was placed randomly into our survey. The demographics of the sample collected is dominated by North American respondents who make up more than half of the sample. However, given the findings of Tiefenau et al. [29], whose sample was predominantly European and the similarities in findings when compared to the Li et al's [28] largely US sample indicate that although there are cultural differences between these regions, the practice of patching may be universal. Furthermore, reddit is known to both be primarily used in Western Nations, particularly the United States, and is also skews towards male users [160]. The over-

whelming majority of admins surveyed are male, and although inline with previous work [5, 28, 30, 29], future work should attempt to engage with admins of different genders. Finally, our questions regarding their reliance on online sources may be biased due to the fact that we directly recruited from such platforms, and therefore may not be reflective of the larger population of system administrators. However, given the fact that research into the potentially comparable user group of developers has also investigated the influence of online forums and Q&A platforms [34], we believe our research and findings are applicable for sysadmins.

5.6 Conclusion

This Chapter presented the design, analysis, and results of an online survey shared with a number of online communities of sysadmins to identify the impact of context on patching behaviours. We provide the research community with the largest sample of admins to date, giving us greater confidence in the results found. Analysis shows that there exists a number of differences in the approaches used by admins working for SMEs compared to Large organizations indicating that factors, such as patching policies are more relevant in larger organization. Additionally, we have expanded upon our understanding of testing set-ups used by admins, with more informal approaches being more popular than previously anticipated. Finally, our work highlights that uninstalling offending patches is not the default option of many sysadmins and instead time and effort is placed in scoping the impact of errors, highlighting the problem solving nature of patch management, and the general practice of system administration in general.

Chapter 6

Discussion & Conclusion

In this Chapter, I will discuss my original research questions from Chapter 1 in light of the studies presented in the previous Chapters, highlighting the contributions made as part of this thesis. Additionally, I will reflect on the body of work through the lens of Communities of Practice, which will aid in my analysis and aid in generalising some of my observations regarding sysadmins and patching to the wider practice of system administration and cyber security. I will then go on to consider on the original high level question, "Why do sysadmins avoid updating?" and reflect on the results of this thesis to suggest potential answers to this questio. Finally, I will take lessons learned and apply them to other research conducted with sysadmins within this space to provide insights and potential future directions for expanding research involving both patching and system administration.

6.1 Discussion

The work in this thesis represents a triangulation of data to provide the research community some understanding of the information sources used and behaviours demonstrated during the patching process. The clearest implication of my work is that sysadmins remain reliant on their communities [87] as sources for adapting best practice to their individual context. Additionally, these communities provide information that is used within all stages of the patching process, with the issues introduced by sub par patches requiring further verification of information [61].

As demonstrated in Chapter 3, the largest identified theme was that of "Errors and Troubleshooting" with community members asking for advice around discovered errors or providing the mailing list with indications of problematic patches. This re-

sult highlighted that sysadmins may reach out to known information sources for help and guidance, similar to the approach taken by developers [34, 24]. Furthermore, the breadth of themes identified indicated the extent of information gathering that may be necessary as part of making and informed decision regarding a particular patch and its potential impact when deployed. Previous work by Li et al. [28] and Tiefenau et al. [29] highlighted the expansive spread of sources that admins reported using for becoming aware of a patch's release, with Tiefenau et al. suggesting that the creation of a centralised source for sysadmins to share patching information would improve the situation. This thesis, however, illustrates that the need for a centralised source has been satisfied, at least in part, through these necessary communities. An active need or *potential* existed within the practice of Patch Management due to the complexities in calculating the risks of a particular patch, as my case study shows. My results also highlight that such CoPs also provide their members with access to veteran sysadmins who can provide advice and knowledge drawn from their experiences that aids less experienced sysadmins with the practice of system administration and patch management. Since Communities of Practice facilitate situational learning [25], allowing members to participate in community lead discussion similar to how apprentices learn from their master. Given that the initial work is focused solely on a mailing list and forums which facilitate discussion to a greater degree than Q&A platforms, such as StackOverflow [34].

My detailed case study, Chapter 4, provides rich data on a single pair of Microsoft patches where issues are found and analysed by a number of communities in advance of the information propagating through out the wider community. The results are similar to the work surrounding both news aggregators and online forums and their impact on software developers [161, 162], with forums providing sysadmins with a space to share knowledge and provide complementary information to official released update documentation. The implications being that vendor release notes are not sufficient for admins to make their patching decisions, and these communities provides complementary and additional informal support without which can extend systems' windows of vulnerability. The case study also identifies that one explanation for admins delaying the installation of an update are reports of patch quality from community members. In our case, admins required the patches to be fixed by the vendor in the form of a hotfix, waiting two full weeks for an initial response from Microsoft. The community actively work together to drill down on one single question, are the errors my system's fault or is it the patch? Once the communities have their answer they can work as a

collective to gain the attention of the software vendor, in our case Microsoft, providing evidence that a fix must come from them and that the issues are not a one off. Having a collective effort working towards the goal of developing an understanding of the risks of a patch, potentially leads to a more direct response as is it happened in our case. By providing crowdsourced information regarding the quality of a patch, sourced through reports of errors from community members the benefits are twofold as we can both highlight the patches that are working for sysadmins, and those that need remediation from the vendors themselves, potentially reducing the amount of time admins wait for such responses [13].

Considering the level of support that communities provided in understanding a patch's risk, and the results of Chapter 5 illustrating a lack of admins engagement with information on vulnerabilities, CVEs, and their critically. The results indicate their may be a need to change how vulnerabilities are reported and scored [163], with the need for scoring to take into consideration more contextual factors and information that is used by admins to quantify risks. This thesis provides a basis for building a better understanding of what sysadmins actually consider when they make their patching decisions.

6.1.1 Limitations

This work is predominantly informed by the observations made within one particular community, that of PatchManagement.org. My initial exploratory qualitative codebook work (Chapter 3) and the work conducted as part of the descriptive case study (Chapter 4) were based on online observations made, with the latter highlighting the connections present between a number of online communities. The case study work encouraged me to combine my findings with results of others [28, 29, 44], and attempt to gather a sample of sysadmins across those discovered communities to generalise my observations (Chapter 5). However, there still remain a number of distinct features of the community observed that will result in differing practices from other sysadmins and their respective patching behaviours and strategies.

The largest clear example is that of the focus on Microsoft Windows OS. Although my findings have a great deal of merit given the scale of usage of Windows OS, there still remains intrinsic difference from the delivery of patches and updates when compared to other examples. The most notable of this is that of Linux, which is an open source software (OSS). This results in distinctly different approaches to the mainte-

nance of security for the platform, and therefore will result in different approaches to the release of patches. This in turn will produce different behaviours [164] and differing security related issues [165]. Additionally, the chosen medium of a mailing-list will also impact the types of community work that is available to them. If we consider comparing the features of the mailing-list to those of forums or even Reddit, we can see that each platform provides different potentials for collaboration and how praise and appreciation is expressed (up-votes versus ”thank you” emails). Finally, we may consider that the platforms where we sourced the survey members from will already be a group who are engaging heavily with the community. We may have missed out on those admins who work in larger more internally focused organizations where patching decisions take very little stock from the work of online communities.

Despite these differences, my work attempted to focus in on their behaviours within their working context, and the results of my survey suggested that the general practice of patch management may be universally agreed upon. However, information gathering behaviours and sources of information used will differ between platforms, and operating systems.

6.2 System Administrators and Communities of Practice

The theory of Community of Practice (CoP) was initially developed in the early nineties, after anthropologist Jean Lave and educational theorist Étienne Wenger conducted several ethnographic studies investigating apprenticeships and how they may be applied to a general theory of learning [107]. Their observations bore from butchers, midwives, and other practitioners, highlighting to them that learning is not only facilitated within a classroom, but can be found in informal gatherings of professionals, often of ranging skill levels (novices to experts): this concept was defined as *situated learning*. In their work they loosely defined these social phenomena as a CoP and the concept remained of interest and has continually evolved [166]. Wenger himself remained keen to elaborate on the theory and published works where the concept took centre stage, allowing CoPs to be considered in the advancement of a general theory of learning [25].

In the following sections I will discuss the definitions and components of CoPs and link the theoretical underpinnings to my work, observations to develop a deeper understanding of the impact of such CoPs and the practice of Patch Management, as well

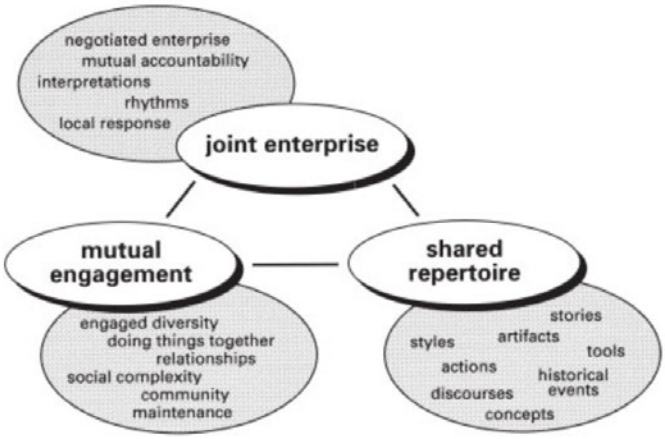

Figure 6.1: Dimensions of practice as the property of a community [25]

as the greater practice of System Administration. As stated previously, the concept of CoP has been applied to System Administration before [5] and by using the domain of Patch Management, future research can benefit from viewing my work from the same theoretical lens. Doing so will aid in understanding how system administrators tackle their work, and their cooperative approach with external communities for informing their decisions. This is especially the case within security, as security sysadmins are distinct from other admins as they are more reliant on collaborative working when tackling security issues and problems due to their intrinsic complexity [53, 18].

6.2.1 What is a Community? The Joint Enterprise of Patch Management

Wenger describes the dimensions upon which a community can be defined in association with practice [25], and they can be seen in Figure 6.1. I detail the 3 dimensions below and relate them to the work detailed within this thesis.

Mutual engagement is essentially how a community functions, as it is through engagement with the practice that membership to a community is obtained. Hence an essential component in the creation of CoPs is the ability for members to engage in the practice and the affordance of engagement in mutual relationships with other community members. These community members and participants represent a diverse range of skills and knowledge, hence they range from the fresh-faced novice, to the battle-hardened veterans of the practice. **Joint enterprise** is the communally negotiated goal of the community and is the result of the full complexities entailed by mutual

engagement and is continually renegotiated by its members. The goal is defined by the very act of pursuing it and the negation is dependent on each member's contexts or situations. This creates a **shared repertoire** is the communities pooled resources (routines, artifacts, vocabulary, etc) that they develop over time. It also combines the reificative and participative aspects of the community and can be seen as the resources necessary for continual mutual engagement.

Throughout my work, my observations and data were originally derived from the sourced mailing-list and then later through communities found to exist on online forums, message boards, and Reddit. The growth of new social media platforms has greatly facilitated the ease of engagement and the range of channels present and may be representative of the changing demographics of the profession of system administration. As novices join, we can see the use of the platform Reddit, with the sysadmin subreddit being established in 2008. Even with the ongoing turmoil of Twitter (also an often cited resource), we have seen the development of distinct Mastodon instances focused on Information Security, an umbrella domain containing patching, vulnerability management and other security sysadmin related duties. Compare this to the age of PatchManagement.org (est. Nov. 2003), or the older origins of AskWoody.com, which originally started as a newsletter, Woody's Windows Watch, in 1998. The integration of new platforms is a symptom of both the ease of mutual engagement, the availability of these platforms, and the consumption preferences of their respective members. Additionally with some platforms having their own version of 'prestige' systems in the form of likes, upvotes, or replies, we can see the quantification of mutual accountability. The practice of patching necessitates engagement with these CoPs when calculating risks as Chapter 5 indicates, given the previously presumptuous accessibility of dedicated testing apparatus.

The development of these platforms has facilitated a need that is clearly there when sysadmins are handling the complexity of both systems and the patches that they must introduce. Patches are fundamental changes in software, and - as the case study demonstrates - can introduce errors, introducing an element of risk that is yet unknown. Through the joint enterprise of calculating the potential risks, sysadmins derive great insight and evidence for their own unique systems and circumstances. This individual knowledge of each member works well within the context of understanding one clear question when troubleshooting: is the error me and my system, or is it the patch? Multiple reports of issues provide clear evidence for the latter, or the opportunity to problem solve with others aiding if the former.

These communities will also have a growing and unique shared repertoire which is heavily influenced by the vendors and the OS that they handle. I would argue that some processes found through the initial focus on Microsoft Windows platform, would also be applicable to other communities such as those focused on Linux. However, they will still be distinguishable by the differences observed in such things as structure of the two OS or even the initial patch notes provided on release [30]. This is evident in how many of the behaviours we investigated had limited differences in approaches by systems but did hint at potential differences when the organizational context was examined.

6.2.2 Boundaries and Brokering by Broker Technicians: Community Overlaps and Landscapes of Practice

Wenger argues that "participation and reification can both contribute to the discontinuity of a boundar" [25]. Some boundaries can be explicit, such as being a subscriber to the mailing-list or to subreddits, or less explicit such as the existence of lurkers [167] or followers on Twitter. The work shown in the Case Study (Chapter 4) and Survey (Chapter 5) spanned a number of different communities, and during my analysis this observation came directly from the change in platform, i.e., mailing-list, to subreddit, to forum, and back again. This complexity of interaction made it inherently difficult to state where one community may end and another began. At the time, I related this to what was defined as a Network of Practice [27]. However, a limitation of this concept is that the authors originally conceived of this being based purely within one organization. This does not reflect the complexities detailed in my thesis, which traverses a range of distinct platforms, yet all seem highly motivated by the previously mentioned joint enterprise of patching successfully.

Wenger would develop and elaborate on the concept of a Landscape of Practice (LoP) [168], which better represents the nature of the phenomena observed within my thesis as its shows a "weaving of boundaries and peripheries between related communities" [169]. That of multiple platforms sharing, exchanging, and digesting the work and construction of knowledge between one another, which is a fundamental aspect of CoPs [170]. Each platform can be viewed as a distinct CoP with connections between them being fostered by 'brokers' or influencers as I previously stated. A related term would be that or a *network broker* [171, 172] or bridge, who connects two social networks together by translating the work or information from one group

to another. These individuals are positioned in such a way that they can observe the flow of information between groups, and are therefore "early to learn about activities in other groups and are often the person introducing to one group information from another" [173].

These brokers share the communal work of their CoP to those on another CoP through use of boundary objects [174]. For example, the use of Proof-of-Concept code detailed in Chapter 4, is a clear example of such an instance. A 'broker', Günter Born, took the posts of others and reified these mumblings into a coherent blog post, from which several other members were able to contribute. One being Woody with a post on Computerworld.com and their forum, and another being the construction of replication of the error. Both of which were again fed back to other communities and their members, igniting further sparks of discussion and detective work. The process of translating and transferring knowledge is one that sysadmins have previously been associated with, hence the coining of broker technicians [62]. The very practice of system administration is rife with constant problem solving [5], causing sysadmins to be accustomed to working closely with their peers, their organizations, and, as my work indicates, even further afield to online communities of professionals. Essentially, this sharing of knowledge and expertise is a natural phenomenon found throughout the working experiences of sysadmins.

6.2.3 The longevity of PatchManagement.org, Microsoft's Patching Cycle, and the rise of alternative Communities

A body of work exists around the creation of CoPs [175], within the domains of education [176] or in healthcare [177]), but the instances I have examined were formed without direct interaction from a larger player, like a vendor such as Microsoft. This occurs when there is *potential* for one, or a 'need', driven by professionals experiencing difficulties or problems framed around their practice [178]. Patching and software updates have been ever present in software, originally coming in the form of paper tape or punched cards which were added to the original tape or patched in (hence the name) over the updated sections. Hence, patching has always been a part of systems and software and will likely forever remain a part of it. Therefore, I argue that such communities will remain active (see Figure 6.3) for the foreseeable future. What will change is technology which continually develops, and as I mentioned in previous sections, we can see across the history of patching through the distinct platforms and the

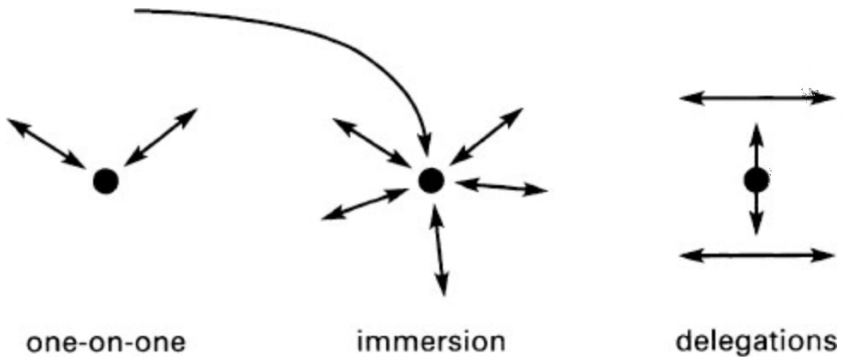

Figure 6.2: Types of boundary encounter [25] - The Proof-of-Concept code is an example of *delegation*, where the creation of a blog by Günter Born spurs the action of one community to investigate the root cause of the observed error. This causes further responses from AskWoody's forum members to discuss the potential work around and approaches to fixing the issue. Each community remains distinct and separate, yet both work to construct meaning from the original blog post of observations, ultimately aiding both in informing their patching decisions.

affordances for interaction that they offer. Consider how PatchManagement.org has a single email thread to represent the Microsoft updates for a month and now the similar approach taken on r/sysadmin with a pinned monthly Patch Tuesday Megathread.

Microsoft Update itself has gone through several platforms, OS variants, delivery mechanisms and documentation, yet the community remains ever present as the need remains constant. Additionally, I would argue that the unique nature of the patching cycle that occurs every month allows the routines and approaches of communities to become readily adaptable and responsive, finding the best practices since new instances are less than a month away. System administration is fundamentally concerned with the adaption and break down of new technology and the practices that they introduce [5] . Already on the horizon we can see the introduction of new alternative approaches from the legacy patching mindset, namely that of DevOps [179]. This approach argues. "Treat servers like cattle, not pets" and claims to be the new and improved way of managing the security and efficiency of one's systems. As my survey shows, the uptake of such expensive (in terms of systems, services, and re-raining) new technologies may not be as prevalent and cost free as the proponents claim. However, once it is the industry standard, I am confident that existing CoPs will adapt or new instances sprout into existence, as some already have with r/devops on Reddit.

Consider the security aspects of patching, we should question whether these CoPs are able to remain resilient in the face of loss of a core group or key members, which can be a reason for CoPs' failure [180]. There remains a need for many sysadmins to use such CoPs as resources to inform their patching decisions, and we already see the passing of key roles and activities. This is due to an awareness of the necessity of such activities to the maintenance of communities and the demands of their members. For example, Woody, the creator of AskWoody, has since retired and has passed the baton over to Susan Bradley, the main moderator of PatchManagement.org. Susan has become a lead moderator for the forum and is now a publisher on the Newsletter, indicating that roles will be filled due to the understanding that their activities remain integral to many admins remaining informed when making risky patching decisions.

6.3 Future Work

6.3.1 Sysadmins' Tools and their impact on Security

The tools used by sysadmins can have a great deal of impact on the security of their organizations and users [21, 23, 64]. Future work should expand on this thesis by looking at the impact of patching related tools, potentially extending designs to include the information that this thesis illustrates as important to sysadmins. Additionally, an argument could be made to extend the scope of the investigation of tools, such as the prominent use of ticketing systems [59]. Communication is key to successful system administration [18], and as this thesis has shown user reports are an important source for sysadmins when patching. Hence, by investigating their current usage we may be able to design future tools to better facilitate the hand-offs between distinct services and their teams. By designing these systems to better support the accurate transfer of information we can improve the security practices of sysadmins within organizations. For example, in Althobaiti et al. [59] work, I helped illustrate the importance of these ticketing systems in handling end-user reports of phishing, finding that IT service management is a Distributed Cognitive [58, 45] process that relies heavily on the correct information being shared at the right time with the appropriate teams. By investigating potential extensions to this design we may be able to improve the working conditions of sysadmins, and benefit their security practices.

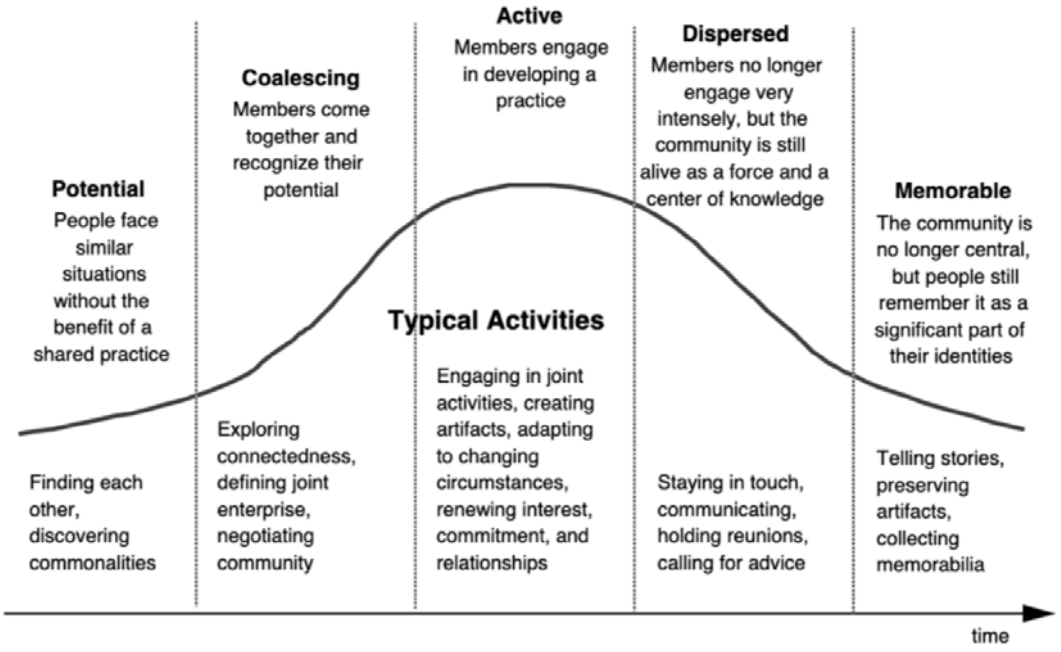

Figure 6.3: Stages of development for a CoP [108] - My observations highlights that the joint enterprise of Patch Management remains *active*, and the results from my qualitative codebook indicates that the admins within PatchManagement.org remain ever vigilant for changes in the mechanisms behind both patch delivery and the OS itself ('Vendor Behaviour' theme). Any changes made by Microsoft to these aspects would result in the need for adaption of current patching strategies. This readiness to evolve the practice can also be applied to new features of the platforms that these communities find themselves on. For example PatchManagement.org was originally a simple mailing list which has now moved to a Google groups as it "was no longer meeting the needs of our user base". [181]

6.3.2 Sysadmins and their Communities

This thesis demonstrates the reliance that sysadmins have on their communities, a finding that is consistent with previous research [5]. Future research should continue to pursue this line of investigation by examining how security information and advice spreads within these networks, similar to the work on developers [161, 34, 162]. Through better understanding of these communities we may be able to identify how admins learn their security practices [148], potentially identifying security misconceptions similar to end-users [182]. Another future direction could be to investigate how these communities handle headline-driven security incidents, which often do not match the scale of harm anticipated [183]. The research community should better utilise these communities, making use of participatory design methods with willing participants sourced from respective populations of interest (i.e. CoP). This thesis, particularly the survey in Chapter 5, greatly benefited from the involvement of and engagement with these communities, which also demonstrates the perceived importance of these issues to the communities themselves. One particular area which may be a fruitful direction is the design of patch release notes [30], which have no current standard or template [150].

6.3.3 Extension Beyond Microsoft

The majority of this work was focused on the Microsoft orientated communities of sysadmins, which although this is a large and important part of the software ecosystem, future work should investigate the communities and the patching schedules of other prominent software vendors. For example, the survey involved a number of Linux admins, however the sample size was still dominated by Windows user. The case study example, Chapter 4, illustrated that even within the scope of Windows there exists a number of distinct communities, with their own unique social norms and community practices. By using the approaches of digital ethnography we can extend the scope of research and investigate these other thriving communities, such as the Linux kernel [184], and extend our understanding of the full patching landscape.

6.4 Conclusion

In this thesis I have investigated information sources used as part of sysadmins' patching process, their beliefs and attitudes regarding patching actions, and the impact of

their organizational context on patching decisions. I found that sysadmins remain reliant on their Communities of Practice, seeking aid regarding patching information. These communities continue to provide advice that relates to both best practice in patch management, and the tools and techniques necessary for a successful patch management strategy. I have shown how these communities collaborate to produce detailed investigations of specific patches and their risk. This suggests that vendors, and patch management tool do not provide the critical information that sysadmins need to make informed patching decisions.

Finally, I have presented the results of an online survey which identifies common patching behaviours and use of information sources in different stages of the patching process. This work highlights that information gathering around patching is an ongoing process, extending into post-installation and troubleshooting of errors. This thesis is one of the first investigations solely focused on system administrators and the security critical process of patch management, and provides a strong basis for future research into both patching, and this vital user group.

Bibliography

[1] Software vulnerabilities increase by 20% in 2021. https://www.hackerone.com/press-release/software-vulnerabilities-increase-20-2021, 2021.

[2] Robert W Reeder, Iulia Ion, and Sunny Consolvo. 152 simple steps to stay safe online: Security advice for non-tech-savvy users. *IEEE Security & Privacy*, 15(05):55–64, 2017.

[3] Cyber essentials. https://cyberessentials.online/cyber-essentials-patch-management-explained/, Nov 2022.

[4] 2017 data breach investigations report. https://www.ictsecuritymagazine.com/wp-content/uploads/2017-Data-Breach-Investigations-Report.pdf, 05 2017.

[5] Eser Kandogan, Paul Maglio, and Eben Haber. *Taming information technology : lessons from studies of system administrators*. Series in human-technology interaction. Oxford University Press, Oxford, 2012.

[6] IT spending and staffing benchmarks 2021/2022. https://www.computereconomics.com/page.cfm?name=it-spending-and-staffing-study, 2021.

[7] Nicole F. Velasquez and Suzanne P. Weisband. Work practices of system administrators: Implications for tool design. In *Proceedings of the 2nd ACM Symposium on Computer Human Interaction for Management of Information Technology*, CHiMiT '08, New York, NY, USA, 2008. Association for Computing Machinery.

[8] Microsoft. Security Update Guide. https://portal.msrc.microsoft.com/en-us/security-guidance. Accessed Aug 27, 2021.

[9] Microsoft. Windows 7 support ended on January 14, 2020. https://support.microsoft.com/en-gb/windows/windows-7-support-ended-on-january-14-2020-b75d4580-2cc7-895a-2c9c-1466d9a53962.

[10] William Smart. Lessons learned review of the wannacry ransomware cyber attack. https://www.england.nhs.uk/wp-content/uploads/2018/02/lessons-learned-review-wannacry-ransomware-cyber-attack-cio-review.pdf, February 2018.

[11] Dan Goodin. Failure to patch two-month-old bug led to massive equifax breach. https://arstechnica.com/information-technology/2017/09/massive-equifax-breach-caused-by-failure-to-patch-two-month-old-bug/, 09 2017.

[12] Adam Shostack. Quantifying patch management. *Secure Business Quarterly*, 3(2):1–4, 2003.

[13] Steve Beattie, Seth Arnold, Crispin Cowan, Perry Wagle, Chris Wright, and Adam Shostack. Timing the application of security patches for optimal uptime. In *LISA*, volume 2, pages 233–242, 2002.

[14] Paul Kocher, Jann Horn, Anders Fogh, , Daniel Genkin, Daniel Gruss, Werner Haas, Mike Hamburg, Moritz Lipp, Stefan Mangard, Thomas Prescher, Michael Schwarz, and Yuval Yarom. Spectre attacks: Exploiting speculative execution. In *40th IEEE Symposium on Security and Privacy (S&P'19)*, 2019.

[15] Moritz Lipp, Michael Schwarz, Daniel Gruss, Thomas Prescher, Werner Haas, Anders Fogh, Jann Horn, Stefan Mangard, Paul Kocher, Daniel Genkin, Yuval Yarom, and Mike Hamburg. Meltdown: Reading kernel memory from user space. In *27th USENIX Security Symposium (USENIX Security 18)*, 2018.

[16] Tom Warren. Microsoft halts amd meltdown and spectre patches after reports of unbootable pcs. https://www.theverge.com/2018/1/9/16867068/microsoft-meltdown-spectre-security-updates-amd-pcs-issues, Jan 2018.

[17] Josh Taylor. Global microsoft outage brings down teams, office 365 and outlook. https://www.theguardian.com/technology/2020/sep/29/major-

microsoft-outage-brings-down-office-365-outlook-and-teams, 09 2020.

[18] Eser Kandogan and Eben M Haber. Security administration tools and practices. *Security and Usability: Designing Secure Systems that People Can Use*, pages 357–378, 2005.

[19] Russell Brandom. Former equifax ceo blames breach on a single person who failed to deploy patch. https://www.theverge.com/2017/10/3/16410806/equifax-ceo-blame-breach-patch-congress-testimony, 10 2017.

[20] Tianyin Xu and Yuanyuan Zhou. Systems approaches to tackling configuration errors: A survey. *ACM Computing Surveys (CSUR)*, 47(4):70, 2015.

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

[22] Sascha Fahl, Yasemin Acar, Henning Perl, and Matthew Smith. Why eve and mallory (also) love webmasters: a study on the root causes of ssl misconfigurations. In *Proceedings of the 9th ACM symposium on Information, computer and communications security*, pages 507–512. ACM, 2014.

[23] Katharina Krombholz, Wilfried Mayer, Martin Schmiedecker, and Edgar Weippl. " i have no idea what i'm doing"-on the usability of deploying https. In *Proc. of the 26th USENIX Security Symposium, ser. USENIX Security*, volume 17, pages 1339–1356, 2017.

[24] Matthew Green and Matthew Smith. Developers are not the enemy!: The need for usable security apis. *IEEE Security & Privacy*, 14(5):40–46, 2016.

[25] Etienne Wenger. *Communities of practice: Learning, meaning, and identity*. Cambridge university press, 1999.

[26] Patchmanagement.org. http://www.patchmanagement.org/default.asp.

[27] John Seely Brown. *The Social Life of Information*. Harvard Business School Press, Boston, 2002.

[28] Frank Li, Lisa Rogers, Arunesh Mathur, Nathan Malkin, and Marshini Chetty. Keepers of the machines: Examining how system administrators manage soft-

ware updates for multiple machines. In *Fifteenth Symposium on Usable Privacy and Security ({SOUPS} 2019)*, 2019.

[29] Christian Tiefenau, Maximilian Häring, Katharina Krombholz, and Emanuel von Zezschwitz. Security, availability, and multiple information sources: Exploring update behavior of system administrators. In *Sixteenth Symposium on Usable Privacy and Security ({SOUPS} 2020)*, pages 239–258, 2020.

[30] Florin Martius and Christian Tiefenau. What does this update do to my systems?–an analysis of the importance of update-related information to system administrators. In *6th Workshop on Security Information Workers*, 2020.

[31] Pooya Jaferian, Kirstie Hawkey, and Konstantin Beznosov. Challenges in evaluating complex it security management systems. In *Symposium on Usable Privacy and Security (SOUPS)*, 2010.

[32] Andrew G Kotulic and Jan Guynes Clark. Why there aren't more information security research studies. *Information & Management*, 41(5):597–607, 2004.

[33] Dennis G Hrebec and Michael Stiber. A survey of system administrator mental models and situation awareness. In *Proceedings of the 2001 ACM SIGCPR conference on Computer personnel research*, pages 166–172. ACM, 2001.

[34] Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. Understanding privacy-related questions on stack overflow. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

[35] Olivier Crameri, Nikola Knezevic, Dejan Kostic, Ricardo Bianchini, and Willy Zwaenepoel. Staged deployment in mirage, an integrated software upgrade testing and distribution system. In *ACM SIGOPS Operating Systems Review*, volume 41, pages 221–236. ACM, 2007.

[36] Vulnerability management. `https://www.ncsc.gov.uk/guidance/vulnerability-management`, 09 2016.

[37] Peter Mell, Tiffany Bergeron, and Dave Henning. Creating a patch and vulnerability management program, 2005-11-16 2005.

[38] Iulia Ion, Rob Reeder, and Sunny Consolvo. "... no one can hack my mind": Comparing expert and non-expert security practices. In *SOUPS*, volume 15, pages 1–20, 2015.

[39] Omid Setayeshfar, Junghwan "John" Rhee, Chung Hwan Kim, and Kyu Hyung Lee. Find my sloths: Automated comparative analysis of how real enterprise computers keep up with the software update races. In Leyla Bilge, Lorenzo Cavallaro, Giancarlo Pellegrino, and Nuno Neves, editors, *Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 215–236, Cham, 2021. Springer International Publishing.

[40] Frank Li and Vern Paxson. A large-scale empirical study of security patches. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, CCS '17, page 2201–2215, New York, NY, USA, 2017. Association for Computing Machinery.

[41] Thomas Duebendorfer and Stefan Frei. Why silent updates boost security. *TIK, ETH Zurich, Tech. Rep*, 302:98, 2009.

[42] David Moore, Colleen Shannon, et al. Code-red: a case study on the spread and victims of an internet worm. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 273–284. ACM, 2002.

[43] Leyla Bilge and Tudor Dumitraş. Before we knew it: an empirical study of zero-day attacks in the real world. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 833–844. ACM, 2012.

[44] Nesara Dissanayake, Asangi Jayatilaka, Mansooreh Zahedi, and M. Ali Babar. Software security patch management - a systematic literature review of challenges, approaches, tools and practices. *Information and Software Technology*, 144:106771, 2022.

[45] Paul P Maglio, Eser Kandogan, and Eben Haber. Distributed cognition and joint activity in collaborative problem solving. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 25, 2003.

[46] Rob Barrett, Yen-Yang Michael Chen, and Paul P Maglio. System administrators are users, too: designing workspaces for managing internet-scale systems. In *CHI'03 Extended Abstracts on Human Factors in Computing Systems*, pages 1068–1069. ACM, 2003.

[47] Rob Barrett, Eser Kandogan, Paul P Maglio, Eben M Haber, Leila A Takayama, and Madhu Prabaker. Field studies of computer system administrators: analysis

of system management tools and practices. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 388–395. ACM, 2004.

[48] Rob Barrett. People and policies: Transforming the human-computer partnership. In *null*, page 111. IEEE, 2004.

[49] Rob Barrett, Paul P Maglio, Eser Kandogan, and John Bailey. Usable autonomic computing systems: The system administrators' perspective. *Advanced Engineering Informatics*, 19(3):213–221, 2005.

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

[51] John Bailey, Eser Kandogan, Eben Haber, and Paul P Maglio. Activity-based management of it service delivery. In *Proceedings of the 2007 symposium on Computer human interaction for the management of information technology*, page 5. ACM, 2007.

[52] Eben M Haber, Eser Kandogan, and Paul P Maglio. Collaboration in system administration. *Communications of the ACM*, 54(1):46–53, 2011.

[53] Eben Haber and Eser Kandogan. Security administrators: A breed apart. *SOUPS USM*, pages 3–6, 2007.

[54] Herbert H Clark. Arranging to do things with others. In *Conference Companion on Human Factors in Computing Systems*, pages 165–167. ACM, 1996.

[55] Susan E Brennan. The grounding problem in conversations with and through computers. *Social and cognitive approaches to interpersonal communication*, pages 201–225, 1998.

[56] Gary Klein, Paul J Feltovich, Jeffrey M Bradshaw, and David D Woods. Common ground and coordination in joint activity. *W.R. Rouse & K.B. Boff (eds), Organizational Simulation*, pages 139–184, 2005.

[57] Edwin Hutchins. How a cockpit remembers its speeds. *Cognitive science*, 19(3):265–288, 1995.

[58] David Botta, Kasia Muldner, Kirstie Hawkey, and Konstantin Beznosov. Toward understanding distributed cognition in it security management: the role of cues and norms. *Cognition, Technology & Work*, 13(2):121–134, 2011.

[59] Kholoud Althobaiti, Adam Jenkins, and Kami Vaniea. A case study of phishing incident response in an educational organization. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work and Social Computing*. ACM, 2021.

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

[61] Nicole F Velasquez and Alexandra Durcikova. Sysadmins and the need for verification information. In *Proceedings of the 2nd ACM Symposium on Computer Human Interaction for Management of Information Technology*, page 4. ACM, 2008.

[62] Nicole F Velasquez and Suzanne P Weisband. System administrators as broker technicians. In *Proceedings of the Symposium on Computer Human Interaction for the Management of Information Technology*, page 1. ACM, 2009.

[63] Constanze Dietrich, Katharina Krombholz, Kevin Borgolte, and Tobias Fiebig. Investigating system operators' perspective on security misconfigurations. In *The 25th ACM Conference on Computer and Communications Security (CCS'18)*. ACM, 2018.

[64] Christian Tiefenau, Emanuel von Zezschwitz, Maximilian Häring, Katharina Krombholz, and Matthew Smith. A usability evaluation of let's encrypt and certbot: Usable security done right. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 1971–1988, 2019.

[65] Artem Voronkov, Leonardo A. Martucci, and Stefan Lindskog. System administrators prefer command line interfaces, don't they? an exploratory study of firewall interfaces. In *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*, pages 259–271, Santa Clara, CA, August 2019. USENIX Association.

[66] Sara Kraemer and Pascale Carayon. Human errors and violations in computer and information security: The viewpoint of network administrators and security specialists. *Applied Ergonomics*, 38(2):143 – 154, 2007.

[67] Rick Wash and Emilee Rader. Too much knowledge? security beliefs and protective behaviors among united states internet users. In *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*, pages 309–325, Ottawa, July 2015. USENIX Association.

[68] Alain Forget, Sarah Pearman, Jeremy Thomas, Alessandro Acquisti, Nicolas Christin, Lorrie Faith Cranor, Serge Egelman, Marian Harbach, and Rahul Telang. Do or do not, there is no try: user engagement may not improve security outcomes. In *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*, pages 97–111, 2016.

[69] Francesco Vitale, Joanna Mcgrenere, Aurélien Tabard, Michel Beaudouin-Lafon, and Wendy E Mackay. High costs and small benefits: A field study of how users experience operating system upgrades. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 4242–4253. ACM, 2017.

[70] Antonio Nappa, Richard Johnson, Leyla Bilge, Juan Caballero, and Tudor Dumitraş. The attack of the clones: A study of the impact of shared code on vulnerability patching. In *2015 IEEE symposium on security and privacy*, pages 692–708. IEEE, 2015.

[71] Arunesh Mathur, Nathan Malkin, Marian Harbach, Eyal Peer, and Serge Egelman. Quantifying users' beliefs about software updates. *arXiv preprint arXiv:1805.04594*, 2018.

[72] Rick Wash, Emilee Rader, Kami Vaniea, and Michelle Rizor. Out of the loop: How automated software updates cause unintended security consequences. In *Symposium on Usable Privacy and Security (SOUPS)*, pages 89–104, 2014.

[73] Michael Fagan and Mohammad Maifi Hasan Khan. Why do they do what they do?: A study of what motivates users to (not) follow computer security advice. In *Twelfth symposium on usable privacy and security (SOUPS 2016)*, pages 59–75, 2016.

[74] Kami Vaniea and Yasmeen Rashidi. Tales of software updates: The process of updating software. In *CHI 2016: Conference on Human Factors In Computing Systems*, 2016.

[75] Kami Vaniea, Emilee Rader, and Rick Wash. Betrayed by updates: How negative experience affect future security. In *CHI 2014: Conference on Human Factors in Computing Systems*, April 2014.

[76] Ofer Bergman and Steve Whittaker. The cognitive costs of upgrades. *Interacting with Computers*, 30(1):46–52, 2017.

[77] Sadegh Farhang, Jake Weidman, Mohammad Mahdi Kamani, Jens Grossklags, and Peng Liu. Take it or leave it: A survey study on operating system upgrade practices. In *Proceedings of the 34th Annual Computer Security Applications Conference*, ACSAC '18, page 490–504, New York, NY, USA, 2018. Association for Computing Machinery.

[78] Christos Gkantsidis, Thomas Karagiannis, and Milan VojnoviC. Planet scale software updates. In *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '06, page 423–434, New York, NY, USA, 2006. ACM.

[79] Armin Sarabi, Ziyun Zhu, Chaowei Xiao, Mingyan Liu, and Tudor Dumitraş. Patch me if you can: A study on the effects of individual user behavior on the end-host vulnerability state. In *International Conference on Passive and Active Network Measurement*, pages 113–125. Springer, 2017.

[80] Michael Fagan, Mohammad Maifi Hasan Khan, and Ross Buck. A study of users' experiences and beliefs about software update messages. *Computers in Human Behavior*, 51:504–519, 2015.

[81] Michael Fagan, Mohammad Maifi Hasan Khan, and Nhan Nguyen. How does this message make you feel? a study of user perspectives on software update/warning message design. *Human-centric Computing and Information Sciences*, 5(1):1–26, 2015.

[82] Arunesh Mathur, Josefine Engel, Sonam Sobti, Victoria Chang, and Marshini Chetty. " they keep coming back like zombies": Improving software updating interfaces. In *SOUPS*, pages 43–58, 2016.

[83] Kandha Sankarpandian, Travis Little, and W. Keith Edwards. Talc: Using desktop graffiti to fight software vulnerability. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, page 1055–1064, New York, NY, USA, 2008. Association for Computing Machinery.

[84] Yuan Tian, Bin Liu, Weisi Dai, Blase Ur, Patrick Tague, and Lorrie Faith Cranor. Supporting privacy-conscious app update decisions with user reviews. In *Proceedings of the 5th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices*, pages 51–61. ACM, 2015.

[85] Alisa Frik, Nathan Malkin, Marian Harbach, Eyal Peer, and Serge Egelman. A promise is a promise: The effect of commitment devices on computer security intentions. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–12, New York, NY, USA, 2019. Association for Computing Machinery.

[86] Gharad Bryan, Dean Karlan, and Scott Nelson. Commitment devices. *Annu. Rev. Econ.*, 2(1):671–698, 2010.

[87] Adam Jenkins, Pieris Kalligeros, Kami Vaniea, and Maria K. Wolters. "anyone else seeing this error?": Community, system administrators, and patch information. In *2020 IEEE European Symposium on Security and Privacy (EuroS P)*, pages 105–119, 2020.

[88] Sharoda A. Paul and Madhu C. Reddy. Understanding together: sensemaking in collaborative information seeking. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work, CSCW*, pages 321–330, Savannah, Georgia, USA, February 2010. ACM.

[89] Jérôme Segura. Fake spectre and meltdown patch pushes smoke loader malware. `https://blog.malwarebytes.com/cybercrime/2018/01/fake-spectre-and-meltdown-patch-pushes-smoke-loader/`, Jan 2018.

[90] Microsoft. Microsoft security intelligence report, volume 13, January – June 2012.

[91] Symantec Corporation. Internet Security Threat Report, Volume 18, 2013.

[92] National Audit Office. Investigation: Wannacry cyber attack and the nhs. https://www.nao.org.uk/wp-content/uploads/2017/10/Investigation-WannaCry-cyber-attack-and-the-NHS.pdf, Oct 2017.

[93] Eduard Kovacs. Heartbleed still affects 200,000 devices: Shodan. https://www.securityweek.com/heartbleed-still-affects-200000-devices-shodan, January 2017.

[94] Gilad Maayan. Five years later, heartbleed vulnerability still unpatched. https://blog.malwarebytes.com/exploits-and-vulnerabilities/2019/09/everything-you-need-to-know-about-the-heartbleed-vulnerability/, September 2019.

[95] MARC:Mailing list ARChives. Patchmanagement marc. https://marc.info/?l=patchmanagement&r=1&w=2.

[96] Mayank Parmar. Windows continues to be the dominant operating system on the desktop. https://www.windowslatest.com/2018/04/03/windows-continues-to-be-the-dominant-operating-system-on-the-desktop/, Apr 2018.

[97] Microsoft. Description of the windows critical update notification utility. https://support.microsoft.com/en-us/help/224420/description-of-the-windows-critical-update-notification-utility, 2007.

[98] Pie-Li Chao. Windows update and its derivatives - with a focus on sus. Technical report, SANS Institute, April 2003.

[99] Leonard Richardson. Beautiful soup documentation. *April*, 2007.

[100] Web filter categories. https://fortiguard.com/webfilter/categories, 2019.

[101] Peter C. Rigby and Ahmed E. Hassan. What can oss mailing lists tell us? a preliminary psychometric text analysis of the apache developer mailing list. In *Proceedings of the Fourth International Workshop on Mining Software Repositories*, MSR '07, page 23, USA, 2007. IEEE Computer Society.

[102] Amy X. Zhang, Mark S. Ackerman, and David R. Karger. Mailing lists: Why are they still here, what's wrong with them, and how can we fix them? In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing*

Systems, CHI '15, page 4009–4018, New York, NY, USA, 2015. Association for Computing Machinery.

[103] Ridley Jones, Lucas Colusso, Katharina Reinecke, and Gary Hsieh. r/science: Challenges and opportunities in online science communication. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, page 153. ACM, 2019.

[104] Johnny Saldaña. *The coding manual for qualitative researchers*. Sage, 2015.

[105] Mary McHugh. Interrater reliability: The kappa statistic. *Biochemia medica : časopis Hrvatskoga društva medicinskih biokemičara / HDMB*, 22:276–82, 10 2012.

[106] Woody Leonhard. An open letter to Microsoft management re: Windows updating. `https://www.computerworld.com/article/3293440/microsoft-windows/an-open-letter-to-microsoft-management-re-windows-updating.html`, July 2018.

[107] Jean Lave, Etienne Wenger, et al. *Situated learning: Legitimate peripheral participation*. Cambridge university press, 1991.

[108] Etienne Wenger et al. Communities of practice: Learning as a social system. *Systems thinker*, 9(5):2–3, 1998.

[109] Michael Barrett, Sam Cappleman, Gamila Shoib, and Geoff Walsham. Learning in knowledge communities:: Managing technology and context. *European Management Journal*, 22(1):1–11, 2004.

[110] P. K. Kapur, A. K. Shrivastava, and Ompal Singh. When to release and stop testing of a software. *Journal of the Indian Society for Probability and Statistics*, 18(1):19–37, Jun 2017.

[111] Robert K Yin. *Case study research : design and methods*. SAGE, Los Angeles, fifth edition.. edition, 2014.

[112] Helen Sharp, Yvonne Dittrich, and Cleidson RB De Souza. The role of ethnographic studies in empirical software engineering. *IEEE Transactions on Software Engineering*, 42(8):786–804, 2016.

[113] Ms-defcon system. `https://www.askwoody.com/ms-defcon-system/`.

[114] Woody Leonhard. 2000011: Group A, Group B and Group W - what's the difference? https://www.askwoody.com/forums/topic/2000011-group-a-group-b-and-group-w-whats-the-difference/, April 2018. Accessed Feb 19, 2019.

[115] Christina Zhao. Microsoft starts forcing Windows 7 and 8.1 users to update to Windows 10. https://www.independent.co.uk/life-style/gadgets-and-tech/news/windows-7-update-microsoft-81-download-windows-10-software-a7684256.html, April 2017.

[116] Martin Brinkmann. Microsoft Security Updates August 2017 release. https://www.ghacks.net/2017/08/08/microsoft-security-updates-august-2017-release, August 2017. Accessed Feb 19, 2019.

[117] Dustin Childs. The August 2017 Security Update Review. https://www.thezdi.com/blog/2017/8/8/the-august-2017-security-update-review, August 2017. Accessed Feb 19, 2019.

[118] Johannes B. Ullrich. Microsoft Patch Tuesday August 2017. https://isc.sans.edu/forums/diary/Microsoft+Patch+Tuesday+August+2017/22694/, August 2017. Accessed Feb 19, 2019.

[119] Brian Krebs. Critical Security Fixes from Adobe, Microsoft. https://krebsonsecurity.com/2017/08/critical-security-fixes-from-adobe-microsoft-2/, August 2017. Accessed Feb 19, 2019.

[120] Microsoft. August 8, 2017-KB4034679 (Security-only update). https://support.microsoft.com/en-us/help/4034679/windows-7-sp1-windows-server-2008-r2-sp1-update-kb4034679). Accessed Feb 19, 2019.

[121] Microsoft. August 8, 2017-KB4034664 (Monthly Rollup). https://support.microsoft.com/en-us/help/4034664/windows-7-sp1-windows-server-2008-r2-sp1-update-kb4034664. Accessed Feb 19, 2019.

[122] Woody. Lots and lots of patches. https://www.askwoody.com/2017/lots-and-lots-of-patches/, August 2017. Accessed Feb 19, 2019.

[123] Susan Bradley. It's patch day! https://marc.info/?t=150221746200001&r=1&w=2, August 2017. Accessed Feb 19, 2019.

[124] Irfanview Fullscreen bug version 4.44 & 4.37 with windows update KB4034664 . https://irfanview-forum.de/archive/index.php/t-11261.html, August 2017. Accessed Feb 19, 2019.

[125] Günter Born. Windows Update KB4034664 is causing trouble on 2nd screen. http://borncity.com/win/2017/08/12/windows-update-kb4034664-is-causing-trouble-on-2nd-screen/, August 2017. Accessed Feb 19, 2019.

[126] KB4034664 macht Probleme bei PDFXchange . https://www.heise.de/forum/heise-Security/News-Kommentare/Patchday-Windows-Suche-als-Einfallstor-fuer-wurmartige-Attacken/KB4034664-macht-Probleme-bei-PDFXchange/posting-30849369/show/, August 2017. Accessed Feb 19, 2019.

[127] Matlab 2016b command is black after Windows Update. https://de.mathworks.com/matlabcentral/answers/352339-matlab-2016b-command-is-black-after-windows-update?w.mathworks.com, August 2017. Accessed Feb 19, 2019.

[128] Office 2013 not rendering correctly on second monitor only after update. https://social.technet.microsoft.com/Forums/en-US/cc63be1e-0457-4da1-8d83-89b0f79fdddd/office-2013-not-rendering-correctly-on-second-monitor-only-after-update?forum=winserverTS, August 2017. Accessed Feb 19, 2019.

[129] KB4034664 causing issues with NVIDIA video drivers. https://forums.geforce.com/default/topic/1022095/geforce-drivers/kb4034664-causing-issues-with-nvidia-video-drivers/, August 2017. Accessed Feb 19, 2019.

[130] Microsoft Releases KB4034664 and KB4034681 Rollup Updates for Windows 7/8.1. https://news.softpedia.com/news/microsoft-releases-kb4034664-and-kb4034681-rollup-updates-for-windows-7-8-1-517349.shtml, August 2017. Accessed Feb 19, 2019.

[131] How to Enable or Disable Desktop Composition in Windows 7 and Vista. https://www.sevenforums.com/tutorials/127411-desktop-composition-enable-disable.html, November 2010. Accessed Feb 19, 2019.

[132] Woody. This month's Win7 patches KB 4034664, KB 4034679 causing second-screen problems . https://www.askwoody.com/forums/topic/this-months-win7-patches-kb-4034664-kb-4034679-causing-second-screen-problems/, August 2017. Accessed Feb 19, 2019.

[133] Nineberry. Graphics Bug in Windows 7 after installing August 2017 Security Updates. https://www.neunbeere.de/blog/2017/08/graphics-bug-in-windows-7-after-installing-august-2017-security-updates/, August 2017. Accessed Feb 19, 2019.

[134] Woody. This month's Win7 patches KB 4034664, KB 4034679 causing second-screen problems . https://www.computerworld.com/article/3215194/microsoft-windows/two-of-this-months-win7-patches-causing-second-screen-problems.html, August 2017. Accessed Feb 19, 2019.

[135] Eduard C. Groen, Sylwia Kopczynska, Marc P. Hauer, Tobias D. Krafft, and Jörg Dörr. Users - the hidden software product quality experts?: A study on how app users report quality aspects in online reviews. *2017 IEEE 25th International Requirements Engineering Conference (RE)*, pages 80–89, 2017.

[136] Microsoft. August 30, 2017-KB4039884. https://support.microsoft.com/en-us/help/4039884/windows-7-update-kb4039884. Accessed Feb 19, 2019.

[137] Woody. Microsoft patches buggy Windows 7 patch, KB 4039884 solves the dual-monitor rendering problem. https://www.askwoody.com/forums/topic/microsoft-patches-buggy-windows-7-patch-kb-4039884-solves-the-dual-monitor-rendering-problem/, August 2017. Accessed Feb 19, 2019.

[138] Günter Born. Windows 7: KB4039884 fixes dual monitor rendering bug. http://borncity.com/win/2017/08/27/windows-7-kb4039884-

fixes-dual-monitor-rendering-bug/, August 2017. Accessed Feb 19, 2019.

[139] Susan Bradley. KB4039884. https://marc.info/?t=150389768200001&r=1&w=2, August 2017. Accessed Feb 19, 2019.

[140] Woody. Microsoft repairs buggy Win7 security patch with buggy hotfix KB 4039884. https://www.computerworld.com/article/3219738/microsoft-windows/microsoft-repairs-buggy-win7-security-patch-with-buggy-hotfix-kb-4039884.html, August 2017. Accessed Feb 19, 2019.

[141] Woody. Microsoft yanks buggy patch of a buggy patch, KB 4039884 . https://www.computerworld.com/article/3220665/microsoft-windows/microsoft-yanks-buggy-patch-of-a-buggy-patch-kb-4039884.html, August 2017. Accessed Feb 19, 2019.

[142] Woody. Buggy KB 4039884 Win7 patch of a patch, returns with no explanation . https://www.askwoody.com/forums/topic/buggy-kb-4039884-win7-patch-of-a-patch-returns-with-no-explanation/, August 2017. Accessed Feb 19, 2019.

[143] Woody. It's time to install August Windows and Office patches - carefully. https://www.computerworld.com/article/3221371/microsoft-windows/its-time-to-install-august-windows-and-office-patches-carefully.html, September 2017. Accessed Feb 19, 2019.

[144] Verizon. 2020 data breach investigations report. Technical report, Verizon Trademark Services LLC, 2020. Also available as https://vz.to/3vKNI1K. Accessed Jun. 2020.

[145] Noriko Hara and Khe Foon Hew. Knowledge-sharing in an online community of health-care professionals. *Information Technology & People*, 20(3):235–261, 2007.

[146] Suzanne Riverin and Elizabeth Stacey. Sustaining an online community of practice: A case study. *Journal of Distance Education*, 22(2):43–58, 2008.

[147] Elissa M Redmiles, Amelia R Malone, and Michelle L Mazurek. I think they're trying to tell me something: Advice sources and selection for digital security.

In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 272–288. IEEE, 2016.

[148] Elissa M Redmiles, Sean Kross, and Michelle L Mazurek. How i learned to be secure: a census-representative survey of security advice sources and behavior. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 666–677. ACM, 2016.

[149] Nancy A Van House, Mark H Butler, and Lisa R Schiff. Cooperative knowledge work and practices of trust: Sharing environmental planning data sets. In *CSCW*, volume 98, pages 335–343, 1998.

[150] Surafel Lemma Abebe, Nasir Ali, and Ahmed E Hassan. An empirical study of software release notes. *Empirical Software Engineering*, 21(3):1107–1142, 2016.

[151] Emad Aghajani, Csaba Nagy, Olga Lucero Vega-Márquez, Mario Linares-Vásquez, Laura Moreno, Gabriele Bavota, and Michele Lanza. Software documentation issues unveiled. In *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*, pages 1199–1210, 2019.

[152] Sauvik Das, Adam DI Kramer, Laura A Dabbish, and Jason I Hong. The role of social influence in security feature adoption. In *Proceedings of the 18th ACM conference on computer supported cooperative work & social computing*, pages 1416–1426. ACM, 2015.

[153] Shundan Xiao, Jim Witschey, and Emerson Murphy-Hill. Social influences on secure development tool adoption: why security tools spread. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 1095–1106, 2014.

[154] Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. Understanding privacy-related questions on stack overflow. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–14, New York, NY, USA, 2020. Association for Computing Machinery.

[155] Ingolf Becker, Simon Parkin, and M Angela Sasse. Finding security champions in blends of organisational culture. *Proc. USEC*, 11, 2017.

[156] Trevor Gabriel and Steven Furnell. Selecting security champions. *Computer Fraud & Security*, 2011(8):8–12, 2011.

[157] Scipy. https://docs.scipy.org/doc/scipy-1.3.1/reference/, Sep 2019.

[158] superuser. https://superuser.com/.

[159] Robert J Fisher. Social desirability bias and the validity of indirect questioning. *Journal of consumer research*, 20(2):303–315, 1993.

[160] Michael Barthel, Galen Stocking, Jesse Holcomb, and Amy Mitchell. Seven-in-ten reddit users get news on the site. https://www.pewresearch.org/journalism/2016/02/25/seven-in-ten-reddit-users-get-news-on-the-site/, Feb 2016.

[161] Maurício Aniche, Christoph Treude, Igor Steinmacher, Igor Wiese, Gustavo Pinto, Margaret-Anne Storey, and Marco Aurélio Gerosa. How modern news aggregators help development communities shape and share knowledge. In *Proceedings of the 40th International Conference on Software Engineering*, ICSE '18, page 499–510, New York, NY, USA, 2018. Association for Computing Machinery.

[162] Leticia S. Machado, Igor Steinmacher, Sabrina Marczak, and Cleidson R. B. de Souza. *How Online Forums Complement Task Documentation in Software Crowdsourcing*, page 101–108. Association for Computing Machinery, New York, NY, USA, 2020.

[163] Jonathan Spring, Eric Hatleback, Allen Householder, Art Manion, and Deana Shick. Time to change the cvss? *IEEE Security Privacy*, 19(2):74–78, 2021.

[164] Shao-Fang Wen. Learning secure programming in open source software communities: A socio-technical view. In *Proceedings of the 6th International Conference on Information and Education Technology*, ICIET '18, page 25–32, New York, NY, USA, 2018. Association for Computing Machinery.

[165] Shao-Fang Wen, Mazaher Kianpour, and Stewart Kowalski. An empirical study of security culture in open source software communities. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '19, page 863–870, New York, NY, USA, 2020. Association for Computing Machinery.

[166] Linda C Li, Jeremy M Grimshaw, Camilla Nielsen, Maria Judd, Peter C Coyte, and Ian D Graham. Evolution of wenger's concept of community of practice. *Implementation science*, 4(1):1–8, 2009.

[167] Na Sun, Patrick Pei-Luen Rau, and Liang Ma. Understanding lurkers in online communities: A literature review. *Computers in Human Behavior*, 38:110–117, 2014.

[168] Etienne Wenger-Trayner, Mark Fenton-O'Creevy, Steven Hutchinson, Chris Kubiak, and Beverly Wenger-Trayner. *Learning in landscapes of practice: Boundaries, identity, and knowledgeability in practice-based learning*. Routledge, 2014.

[169] Igor Pyrko, Viktor Dörfler, and Colin Eden. Communities of practice in landscapes of practice. *Management Learning*, 50(4):482–499, 2019.

[170] Constantin Bratianu. Knowledge sharing and communities of practice. *Organizational knowledge dynamics: Managing knowledge creation, acquisition, sharing, and transformation. Hershey, PA: IGI Global*, 10:978–1, 2015.

[171] Ronald S Burt. Structural holes and good ideas. *American journal of sociology*, 110(2):349–399, 2004.

[172] Ronald S Burt. The social capital of opinion leaders. *The Annals of the American Academy of Political and Social Science*, 566(1):37–54, 1999.

[173] Ronald S Burt and Giuseppe Soda. Network capabilities: Brokerage as a bridge between network theory and the resource-based view of the firm. *Journal of Management*, 47(7):1698–1719, 2021.

[174] Susan Leigh Star and James R Griesemer. Institutional ecology,translations' and boundary objects: Amateurs and professionals in berkeley's museum of vertebrate zoology, 1907-39. *Social studies of science*, 19(3):387–420, 1989.

[175] Etienne Wenger, Richard Arnold McDermott, and William Snyder. *Cultivating communities of practice: A guide to managing knowledge*. Harvard business press, 2002.

[176] Paul A Kirschner and Kwok-Wing Lai. Online communities of practice in education. *Technology, Pedagogy and Education*, 16(2):127–131, 2007.

[177] Geetha Ranmuthugala, Jennifer J Plumb, Frances C Cunningham, Andrew Georgiou, Johanna I Westbrook, and Jeffrey Braithwaite. How and why are communities of practice established in the healthcare sector? a systematic review of the literature. *BMC health services research*, 11(1):1–16, 2011.

[178] Sasha Barab, Scott J Warren, Rodrigo del Valle, and Fang Fang. Coming to terms with communities of practice. *Handbook of human performance technology*, pages 640–664, 2006.

[179] Nicole Forsgren and Jez Humble. Devops: Profiles in itsm performance and contributing factors. *Forsgren, N., J. Humble (2016)." DevOps: Profiles in ITSM Performance and Contributing Factors." In the Proceedings of the Western Decision Sciences Institute (WDSI)*, 2016.

[180] Gilbert Probst and Stefano Borzillo. Why communities of practice succeed and why they fail. *European management journal*, 26(5):335–347, 2008.

[181] Susan Bradley. Moderator note: Just a kind reminder in 10 days this list will shut down. `https://marc.info/?l=patchmanagement&m=155474314429057&w=2`, April 2019. Accessed Nov 19, 2022.

[182] Rick Wash. Folk models of home computer security. In *Proceedings of the Sixth Symposium on Usable Privacy and Security*, SOUPS '10, New York, NY, USA, 2010. Association for Computing Machinery.

[183] Exploring the future of cyber reinsurance. `https://www.ajg.com/gallagherre/news-and-insights/2022/february/future-of-cyber-reinsurance/`.

[184] Linux kernel. `https://www.kernel.org/`, 2022.

Appendix A

Survey of Patching Behaviours

Participant Sheets and Consent Form

Patching Behaviour of System Administrators

What is the purpose of the study?

Our goal is to understand the prevalence of common Patching behaviours among System Administrators when carrying out patch management related tasks, and any deviations that may occur in some instances.

What will happen if I decide to take part?

You will be asked to provide some basic demographics, such as age, gender, and country of work and some demographic-like information regarding the organisation you patch for. Then we will ask you questions about your typical behaviour during different patching phases, based on the systems and machines you are most comfortable talking about.

Why have I been asked to take part?

We are advertising this study to people who are currently working as System Administrators, or where Patch Management is a regular part of their assigned duties.

Do I have to take part?

No – participation in this study is entirely up to you. You can withdraw from the study at any time, without giving a reason. Your rights will not be affected. If you wish to withdraw, contact the Principle Investigator, Dr Kami Vaniea (kvaniea@inf.ed.ac.uk), or lead researcher, Adam Jenkins (adam.jenkins@ed.ac.uk). We will stop using your data in any publications or presentations submitted after you have withdrawn consent. However, we will keep copies of your original consent, and of your withdrawal request.

Are there any benefits associated with taking part?

You will receive no compensation for participating in this study other than the knowledge that you have helped us gain a better understanding of the patching behaviour of admins, with the goal of ultimately improving the current state of Patch Management.

Are there any risks associated with taking part?

There are no significant risks associated with participation.

What will happen to the results of this study?

The results of this study may be summarised in published articles, reports and presentations. Quotes or key findings will be anonymized: we will remove any information that could, in our assessment, allow anyone to identify you. With your consent, information can also be used for future research. Your data may be archived for a minimum of 3 years.

How will my data be protected and kept confidential?

Your data will be processed in accordance with Data Protection Law. All information collected about you will be kept strictly confidential. Your data will be referred to by a unique participant number rather than by name or platform ID. Your data will only be viewed by members of the research team. All electronic data will be stored on password-protected encrypted computers, on the School of Informatics' secure file servers, or on the University's secure encrypted cloud storage services (DataShare, DataSync, or Sharepoint).

What are my data protection rights?

The University of Edinburgh is a Data Controller for the information you provide. You have the right to access information held about you. Your right of access can be exercised in accordance Data Protection Law. You also have other rights including rights of correction, erasure and objection. For more details, including the right to lodge a complaint with the Information Commissioner's Office, please visit www.ico.org.uk. Questions, comments and requests about your personal data can also be sent to the University Data Protection Officer at dpo@ed.ac.uk. For general information about how we use your data, go to: edin.ac/privacy-research

Who is conducting this research and who can I contact about it?

This survey is conducted by researchers from School of Informatics, University of Edinburgh. If you have any further questions about the study, please contact the lead researcher, Adam Jenkins (adam.jenkins@ed.ac.uk), or his supervisors Dr Kami Vaniea (kvaniea@inf.ed.ac.uk) and Dr Maria Wolters (maria.wolters@ed.ac.uk).

If you wish to make a complaint about the study, please contact inf-ethics@inf.ed.ac.uk. When you contact us, please provide the study title and detail the nature of your complaint. This study was certified according to the Informatics Ethics boards, with reference number 2021/95958.

Updated information
If the research project changes in any way, an updated Participant Information Sheet will be made available on: https://web.inf.ed.ac.uk/infweb/research/study-updates

Alternative formats

To request this document in an alternative format, such as large print or on coloured paper, please contact Adam Jenkins (adam.jenkins@ed.ac.uk)

Consent

Please read the following statements carefully:

- I confirm that I have read and understood the Participant Information Sheet for the above study, that I have had the opportunity to ask questions, and that any questions I had were answered to my satisfaction.
- I understand that my participation is voluntary, and that I can withdraw at any time without giving a reason. Withdrawing will not affect any of my rights.
- I consent to my anonymised data being used in academic publications and presentations.
- I understand that my anonymised data can be stored for a minimum of 3 year.

Do you agree to participate in this survey?

- ○ Yes, start the survey
- ○ No

Demography - Outsourcing

How many distinct organisations do you currently manage patches for?

- ○ Currently Unemployed
- ○ 1 Organisation
- ○ 2 - 5 Organisations
- ○ 6 - 10 Organisations
- ○ 11+ organisations

Demography - Information of the organisation

Please answer the following questions about a single organisation.

For the remaining questions, please select one of the organisations you manage patches for and answer based only on that organisation. If you work for an outsourcing or technology support organisation, please select one of your clients and answer the questions about that client.

Please answer the following questions about the last organisation you worked with.

For the remaining questions, please refer to the most recent organisations that you managed patches for and answer based only on that organisation.

What sector does your organisation work in?

How large is your organisation?

- ○ <10 employees
- ○ 10-49 employees
- ○ 50-249 employees
- ○ more than 250 employees

How many other system administrators do you work with?

- ○ 0
- ○ 1-5
- ○ 6-10
- ○ more than 10

Demography - Knowledge of the machine/OS/software

How many machines/devices do you manage?

- ○ 1 - 100 machines
- ○ 101 - 250 machines
- ○ 251 - 500 machines
- ○ 500 - 1000 machines
- ○ 1000+ machines

What type of machines/devices do you manage? (Select all that apply)

- ☐ Client Machines (Laptops, Desktops etc.)
- ☐ Servers
- ☐ Mobile devices (e.g. phones and tablets)
- ☐ Routers/network appliances
- ☐ Embedded devices/Internet of Things
- ☐ [ ] Other

What are the operating systems on the machines that you manage? (Select all that apply)

- ☐ Mac
- ☐ Windows
- ☐ Linux
- ☐ iOS
- ☐ Android
- ☐ Blackberry
- ☐ ChromeOS
- ☐ [ ] Other

What types of software components are you in charge of patching? (Select all that apply)

- ☐ Operating system
- ☐ Applications
- ☐ Custom/bespoke inhouse programs
- ☐ Software that is no longer supported by vendor
- ☐ [ ] Other

Baseline question

Please select one of the types of systems you support and then answer all following questions in regards to that system.

- ○ » Client Machines (Laptops, Desktops etc.)
- ○ » Servers
- ○ » Mobile devices (e.g. phones and tablets)
- ○ » Routers/network appliances
- ○ » Embedded devices/Internet of Things
- ○ [ ] » Other

What operating systems run on the type of system you selected above?
(Select all that apply)

- [ ] » Mac
- [ ] » Windows
- [ ] » Linux
- [ ] » iOS
- [ ] » Android
- [ ] » Blackberry
- [ ] » ChromeOS
- [ ] [ ] » Other

Behaviour - Awareness

When new patches become available, I learn about them through:

	Never	Sometimes	About half the time	Most of the time	Always
Official notifications from **vendors**	○	○	○	○	○
Patches are released on a **regular schedule**, so I am **expecting them**	○	○	○	○	○
Notifications generated by the software, such as a popup	○	○	○	○	○
External **mailing lists, online forums**, and **social media** (e.g. Reddit)	○	○	○	○	○
Tech news and **blogs**	○	○	○	○	○
Patch management software	○	○	○	○	○

Behaviour - Prioritisation

When deciding which patches to prioritise for installation or testing, how often do you engage in each of the following?

	Never	Sometimes	About half the time	Most of the time	Always
Read about **new features** and **improvements** added by the patch	○	○	○	○	○
Collaborate with co-workers in deciding which patches to prioritise	○	○	○	○	○
Seek online advice for which patches to prioritise (forums, blogs etc.)	○	○	○	○	○
Calculate **expected downtime** and its likely cost	○	○	○	○	○
Determine if a patch **fixes existing bugs** that are impacting the organisation's systems	○	○	○	○	○
Read patch **documentation** and r**elease notes**	○	○	○	○	○
Read about a patch's related **CVEs** and **vulnerabilities**	○	○	○	○	○
Read a patch's **known issues** and **workarounds**	○	○	○	○	○
Perform a formal **risk assessment** for a patch or set of patches	○	○	○	○	○
Follow a **patching policy** to guide prioritisation	○	○	○	○	○

Behaviour - Deciding

Who makes the final decision about installing, not installing, or waiting to install a patch?
(Select all that apply)

- ☐ The client
- ☐ My boss, manager, CISO, or executive level
- ☐ I, or my team, make the decision
- ☐ A previously agreed patch policy clearly states
- ☐ [ ] Other

In your opinion, how much say do you have in if a patch will or will not be installed?

- ○ A great deal
- ○ A lot
- ○ A moderate amount
- ○ A little
- ○ None at all

Behaviour - Preparation

When preparing to install patches on a system, how often would you do the the following actions?

	Never	Sometimes	About half the time	Most of the time	Always
Modify **software configuration files**	○	○	○	○	○
Schedule a suitable **patching window** with impacted people	○	○	○	○	○
Make **back-ups** of the system	○	○	○	○	○
Identify software and firmware **dependencies**	○	○	○	○	○
Collect **baseline system performance** metrics	○	○	○	○	○
Modify custom **deployment scripts**	○	○	○	○	○
Modify settings for **third-party patching tools**	○	○	○	○	○
This is an attention **check**, select sometimes	○	○	○	○	○

Behaviour - Testing

When testing patches which of the following test setups do you use?
(Select all that apply)

- ☐ **No testing** is done
- ☐ Testing on a **couple of computers** first
- ☐ Testing via **Virtual Machines (VM)**
- ☐ **Testing on a personal machine**
- ☐ **Testing on a dedicated test environment**
- ☐ **Staged deployment**
- ☐ [ ] **Other**

The deployment of patches is:

- ○ Fully Automated
- ○ Mostly Automated
- ○ Roughly the same
- ○ Mostly Manual
- ○ Fully Manual

What most motivated your testing setup?

- ○ Speed to patch
- ○ Resource availability for testing
- ○ Patching policies
- ○ Patches rarely cause errors
- ○ [ ] Other

When testing patches, what is considered by your organisation? Please rank the following options with 1 (top) being the most important.

Time to patch, getting patches deployed as quickly as possible

Stability and uptime of system

User experience

Efficient use of staff time

Efficient use of computing resource (e.g. VMs or development servers)

Behaviour - Post-Deployment State

Once a patch has been deployed, to validate that it is working as expected, I will:
(Select all that apply)

- ☐ **Compare** system performance **against baseline**
- ☐ Monitor **user reports**
- ☐ Monitor **online sources** (e.g. blogs, forums) for reported **patch issues**
- ☐ Monitor **vendor announcements**
- ☐ Monitor for changes in **patch documentation**
- ☐ [ ] Other

When you detect an error after testing or deployment, what actions would you perfrom?
(Select all that apply)

- ☐ Immediately **uninstall the patch**
- ☐ Raise ticket with **software vendor**
- ☐ Read/ask external **online forums or mailing lists**
- ☐ Ask **colleagues**
- ☐ Investigate **patch documentation**
- ☐ Local investigation of **cause and scope of error**
- ☐ [ ] Other

Demographics - Personal information

How many years old are you?

- ○ 18-25
- ○ 25-35
- ○ 35-45
- ○ 45-55
- ○ 55-65
- ○ over 65
- ○ I do not wish to disclose

Which country did/do you currently work in?

United Kingdom of Great Britain and Northern Ireland

What is your gender?

- ○ Male
- ○ Female
- ○ Non-Binary / Third Gender
- ○ [ ] Other/ Wish to self identify
- ○ Prefer not to say

Demographics - Education background

What is the highest level of education that you have completed?

- ○ Secondary school or less
- ○ High School
- ○ College but no degree
- ○ Bachelor's degree
- ○ Master's degree
- ○ Phd degree
- ○ [ ] Other

What is the subject area of your highest level of education?

Demography - Career situation

What is your current job title?

What was your job title?

For how many years did you work in that role?

- ○ < 1 year
- ○ 1-2 years
- ○ 2-5 years
- ○ 5-10 years
- ○ 10+ years

For how many years have you worked in this current role?

- < 1 year
- 1-2 years
- 2-5 years
- 5-10 years
- 10+ years

How much total experience have you had in the area of System Administration?

- < 1 year
- 1 - 2 years
- 3 - 5 years
- 6 - 10 years
- 11+ years

ANYTHING Else

Optional: Is there anything else about patching awareness, prioritisation, testing, deployment, or troubleshooting that you would like to tell us about?