International Journal of Cyber-Security and Digital Forensics (IJCSDF) 4(2): 354-379
The Society of Digital Information and Wireless Communications, 2015 (ISSN: 2305-0012)

Characterising File Sharing and its Protection, the Insider Threat Case

Rakan Alsowail and Ian Mackie
School of Engineering and Informatics,
University of Sussex, Falmer, Brighton, BN1 9RH, United Kingdom
ra216@sussex.ac.uk, i.mackie@sussex.ac.uk

ABSTRACT

File sharing has become an indispensable part of our daily lives. Some of the shared files are sensitive, thus, their confidentially, integrity and availability should be protected. This paper investigates the protection requirements and the activities of file sharing from the perspective of the insider threat problem. It addresses three fundamental questions to the design of a protection mechanism against insider misuses: who is the insider, what are the insider misuses, and how the activity of file sharing can be performed. This paper proposes a new approach for classifying the insider threat problem into different categories and then focuses on one category that is related to file sharing. It characterises the protection required by the shared files against different types of insiders misuses and characterises the activity file sharing based on two factors: how files can be propagated and how they can be accessed after their propagation.

KEYWORDS

Insider threat, security, information security, file sharing, data confidentiality, file dissemination.

1 INTRODUCTION

File sharing has been a topic of interest in computer science right from the beginning–ever since files were created. The prevalence of file sharing activity nowadays is attributed to the existence of variety of methods that simplified such activity to be performed. These methods have gone through several stages until they reached the maturity at the present time to become fundamentals to any Internet user. At the beginning, no actual storage medias existed, where the only way to transfer information from one computer to another is to type them manually. Later on, the first magnetic storage media emerged which could contain data, however, moving around these magnetic storage were very difficult [1]. The first time file sharing became an easy task to perform was in 1971 when the 8-inch floppy disk was developed by IBM [2, 3, 4, 5]. However, spreading of files went slowly as the files had to be moved physically from one place to another. Users were able to share files online by utilising their phone lines in 1978 when the the first online bulletin board system (BBS) emerged. This was followed by various methods of sharing such as Usenet in 1979, FTP in 1985, Napster in 1999. From 2000 up to the present time, a wide variety of peer-to-peer file sharing system emerged such as Gnutella, eDonkey 2000, Kazaa, BitTorrent as well as web-based file sharing services such as Dropbox, GoogleDocs, youSENDit, Streamfile, Wikisend, 4shared and social networking sites such as Facebook, YouTube, Instagram and Flickr.

The existence of these methods nowadays encouraged more people to share files with each other. Some of the shared files might be confidential content that needs to be protected. Such confidential content raised the awareness of people to the security concept *share but protect*. Depend-

ing on the nature of the content, the shared files might need to be protected against unauthorised disclosure, modification, or withholding. Generally speaking, such protections stem from three distinct fields of security due to the fact that data can be in three different states. First, data can be stored, and protecting it is the main concern of a field named Perimeter security which prevents attacks on data stored inside a trusted internal network. Second, data can be in transit, and protecting it is the main concern of a field named Communication security which prevents attacks on data transmitted over a network. Third, data can be in use, and protecting it is the main concern of a field named Insider security which prevents attacks on data by those who have authorised access.

According to the 2011 CyberSecurity Watch Survey, conducted by the U.S. Secret Service, the CERT Insider Threat Center, CSO Magazine, and Deloitte [6], 58% of the attacks are caused by outsiders (those unauthorised to access network systems or data) while 21% of the attacks are caused by insiders (those authorised to access network systems and data), and 21% from unknown sources. Even though the percentage of insider attacks is less than the external attacks, the consequences of insider attacks can be more severe. The survey indicated that 33% of the respondents consider insider attacks to be more costly and damaging. Consequently, insider attacks merit the same attention as external attacks.

Protecting the shared files from the perspective of Insider security is a challenging problem. It has always been recognised that preventing policy violation by authorised users is more challenging than those who are not. Authorised users have access privileges that make it hard to prevent or detect policy violation. Additionally, providing a mechanism to protect the shared files from insiders requires an investigation into three fundamental questions which we address in this paper.

- First: What is the insider problem?

The problem with the insider security literature is that there is no a widely accepted definition of what is an insider and there is no a clear distinction between insiders and outsiders. What is considered an insider for someone might be an outsider for someone else. Therefore, protecting the shared files from insiders without knowing who is the insider is meaningless. Bishop and Gates [7] pointed out that there exist many definitions of insider and insider threat in the literature that complicated the research in insider threats as one solution to the insider problem might not be applicable to another insider problem. Also, Hunker [8] stated that although there exists a large body of work in the literature to address the insider threat problem, little progress has been made due to the absence of clear answers to fundamental questions such as "What is an insider threat". We believe that the insider problem should be classified into several categories which can be defined, studied and solved independently, and later combined to solve the problem as a whole.

- Second: What are the insider misuses?

Defining the insider problem and the insider precisely is the first step towards protecting the shared files from insiders. What more important is identifying the misuses that can be performed by insiders. Misuses are actions taken by the insider which violate the confidentiality, integrity or availability of a particular asset. by knowing the misuses that the insider can perform on the shared files, we can derive the different types of protections that are required to protect the shared files.

- Third: How the activity of file sharing can be performed?

While the first two questions are related to the insider security, this question is related to the activity of file sharing. However, similar to the insider problem, the activity of file sharing is not clearly identified. Most of the research on file sharing is focused on specific domains and applications while little research studied file sharing more broadly [9]. The term *file sharing* is rarely defined in the literature, and if defined, it is tailored to a specific method of sharing. One exception is the study by Whalen et al. [9] who defined file sharing as "the activity of making specified

file(s) available to an individual or group, with the option of granting specific right (e.g., ability to view, edit, delete) over those files". However, a general characterisation of how the activity of file sharing can performed is currently missing.

Protecting the shared files is a topic that have been studied from two different fields with different interests, namely, information sharing and security. The former focuses on facilitating information sharing and provides sharing tools that are suitable for various tasks of sharing but not secure. The latter focuses on securing information sharing and provides sharing tools that are secure but not suitable for every task of sharing. Considering both fields will help us to design a protection mechanism that will not only protect the shared files from insiders but also allow owners of files to share their files as desired. Therefore, in addition to identifying the different types of insider misuses, we investigate how the activity of file sharing can be performed.

In this paper we study the protection requirements and the activities of file sharing from the perspective of the insider threat problem by providing answers to the above three questions. The rest of this paper is structured as follows. In Section 2 we review the literature and related work on insider threat and file sharing, respectively. In Section 3, we propose a new approach for classifying the insider threat problem and focus on one category that is related to file sharing. In Section 4, we give our first contribution to characterising the protection required by the shared files against different types of insiders. In Section 5, we give our second contribution to characterising file sharing based on two factors: how files are propagated and accessed after their propagation. In Section 6, we define a framework to classify the activity of file sharing. Finally, in Section 7, we conclude the paper with our future work.

2 RELATED WORK

2.1 Insider Security

Several definitions of insider and insider threat exist in the literature. Some authors have focused on the trust relationship when defining the term insider. For instance, RAND report [10] defined the insider as "an already trusted person with access to sensitive information and information systems". Bishop [11] defined the insider as "a trusted entity that is given the power to violate one or more rules in a given security policy". Other authors have focused on the abuse of given access privileges. For instance, Chinchani et al. [12] defined the insider as "legitimate users who abuse their privileges". CERT report [13] defined the insider as "individuals who were, or previously had been, authorised to use the information systems they eventually employed to perpetrate harm". Others defined the insider very broadly. For instance, Predd et al. [14] defined the insider as "someone with legitimate access to an organisation's computers and networks". RAND report [10] defined the insider again as "anyone with access, privilege, or knowledge of information system and services". The former definition might include masqueraders who stole the credential of a legitimate user to get access to the computer or the network. The latter definition eliminates the need of trust and includes those who have knowledge of the system or the service even if they do not have access privileges. In 2008, a cross-disciplinary workshop on "Countering Insider Threats" [15] concluded that

> "an insider is a person that has been legitimately empowered with the right to access, represent, or decide about one or more assets of the organisation's structure"

With regard to insider threat, Predd et al. [14] defined insider threat as "an insider's action that puts an organisation or its resources at risk". RAND report [10] defined it as "malevolent (or possibly inadvertent) actions by an already trusted

person with access to sensitive information and information systems". Hunker and Probst [16] defined it as "an insider threat is [posed by] an individual with privileges who misuses them or whose access results in misuse". The CERT Insider Threat Center's current definition of insider threats as follows:

> "A malicious insider threat to an organisation is a current or former employee, contractor, or other business partner who has or had authorised access to an organisation's network, system, or data and intentionally exceeded or misused that access in a manner that negatively affected the confidentiality, integrity, or availability of the organisation's information or information systems". [17]

Due to the differences and contradictory definition of insider and insider threats that complicated the problem to be solved, many authors are urging the community to establish a framework or taxonomy for distinguishing among different types of insider threats [15, 16]. They mentioned that each determining factor for an insider can be used for a taxonomy, for example based on distinctions between: Malicious and accidental threats; Doing something intentionally (for malice, or good reasons which nonetheless may result in damage) versus events that occur accidentally; Obvious and stealthy acts; Acts by masqueraders (e.g, an individual with a stolen password), traitors (malicious legitimate users) and naive or accidental use that results in harm; A combination of factors such as access types; aim or intentionality or reason for misuse; level of technical and the system consequences of insiders threats.

Bellovin [18] identified three different types of insider attack which are misuse of access, defence bypass, and access control failure. He stated that access control failure attacks can be prevented by purely technical means, while the other two attacks require combination of technical and non-technical means. Hunker and Probst [16] identified three different approaches, which current works in the field revolve around, to solve the insider threat problem. These approaches are technical approach, socio-technical approach, and sociological approach. The authors noted that technical approaches are focused on policy languages, access control and monitoring, while socio-technical approaches are focused on policy, monitoring and profiling, prediction, forensics and response work. Sociological approaches are focused on motivation, organisational culture, human factors and privacy and legal aspects. Silowash et al. [19] analysed cases of insider threat from the CERT insider threat database, which contains more than 700 cases of insider threat, and observed that malicious insider activities can be classified into four classes as follows.

- IT sabotage: an insider's use of IT to direct specific harm at an organisation or an individual. Example of this are destroying critical data, planting logical bomb to delete data at critical times, etc.
- Theft of IP: an insider's use of IT to steal IP from the organisation. This category includes industrial espionage involving outsiders. Examples of usually stolen IP assets are proprietary software, business plans, product details, and customer information.
- Fraud: an insider's use of IT for the unauthorised modification, addition, or deletion of an organisation's data (not programs or systems) for personal gain, or theft of information that leads to an identity crime (e.g., identity theft or credit card fraud).
- Miscellaneous: cases in which the insider's activity was not for IP theft, fraud, or IT sabotage.

2.2 File Sharing

A wide variety of file sharing methods exist, and they differ from one another in the way that they allow users to control the what, how, and with whom to share [20]. Various studies have been conducted to investigate these properties.

Olson et al. [21, 22] conducted a pilot study

and a more formal survey to explore preferences for general information sharing by investigating what information people are willing to share, and with whom. Their findings indicated that people willingness to share is different from one another and it depends on who they are sharing the information with, therefore, a one-size-fits-all permissions structure for sharing is inappropriate. They found that people deal with particular types of information similarly when assessing whether or not to share it with others (example classes include work email and telephone number, pregnancy, health information, email content, credit card number). Also, they found that people deal with particular types of individuals similarly when assessing whether or not to share information with them (example classes include spouse, manager, trusted coworker, the public and competitors). The authors believe that their findings can provide guidance to the design of access control and interfaces.

Voida et al. [20] conducted a survey and follow-up interviews at medium-sized research organisation to explore users' current practices and needs around file sharing. The authors stated that the understanding of what, with whom and how of sharing will lead us to understand users current sharing practices of file sharing. The result of their study indicated that almost third of the respondents shared files with groups or classes of individuals, and in many cases these classes mapped directly onto categories identified by Olson et al. [21, 22]. Also, their survey respondents reported sharing files at work regularly with an average of 7 individuals or groups. With respect to the types of files are shared, their respondent reported 34 different types of files or electronic information they share, which range from business documents and paper drafts to music, ideas, schedules, and TV shows. In terms of how the sharing is taking place, they found that Email is the most common method used for sharing files by their respondents (43% of all responses), followed by shared network folder (16%), followed by posting content to a website (11%).

Their findings indicated that there are three main classes of difficulties and breakdowns that people encounter in sharing, which are: forgetting what file had been shared with who; difficulties in selecting a sharing method with desired features that was also available to all sharing participants; and problem in knowing when new content was made available. They mentioned that their respondents usually fall back to use the most universal method, which is Email, in order to share their files when they are uncertain about the tools available to their intended recipients. Based on their findings, they identified a number of critical characteristics of file sharing methods including universality, addressing, visibility, notification, and the differentiation between push- and pull-oriented sharing. They developed a prototype of a set of user interface features called a sharing palette, providing a platform for exploration and experimentation with new modalities of sharing.

Whalen et al. [23] conducted an online survey and follow up interviews at a medium-sized industrial research laboratory to address the issue of users' experience of file sharing and access control by gathering information on how and why people share files; the types of information shared; and how, when and why people limit access to those files. The results of the survey showed that email attachments were the most commonly used method for sharing files (98% of all responses), followed by network files sharing (55%), followed by commercial content management system (25%) and removable media (25%). Also their result indicated that 37% of respondents protect their shared files from friends and colleagues, and the methods used for restricting access to their sensitive files are: passwords; permissions/access control lists; physical controls (e.g., safeguard in office or on person); encryption ; obscurity (e.g, given files innocuous names, hidden directories); and deleting/relocating sensitive files. Based on the results of the study, the authors suggest guidelines to improve methods for appropriate content protection.

In another study, Whalen et al. [9] conducted a web-based survey at a medium-size university to investigate the fundamental issues regarding how files are shared and the difficulties encoun-

tered when managing files in collaborative environments. From the result of their survey, they found that file sharing is a common activity, with over 70% of respondents share professional and personal files at least once per week . The file sharing methods used by their respondent are email attachments, physical devices (e.g., USB token, CD), networks file share, instant messenger (e.g., MSN, Yahoo), Web server (e.g., webpage, wiki), peer-to-peer (e.g., KaZaa), file copy protocol (e.g., scp,ftp). The most commonly-used file sharing method by their respondents is Email (42.7%) followed by network file share (14.7%) followed by peer-to-peer and file copy protocol (10.3%). This corresponds with the findings of Voida et al. [20] and their previous study [23]. Their results show that there are a number of positive and negative factors that have an impact on peoples choice of file sharing methods. The positive factors are: the convenience and the ease of use of the method, the widespread availability of the method in order to reach all recipients, and the suitability of the method to the organisation or task at hand. The negative factors are: the limit on file space or file size, lack of access control or security features and the inability to reach all recipients. Also, the result show that the majority of respondents share files between two and four groups, and 80% of respondents have sensitive files. These sensitive files were shared as the results indicated that 44% of respondents shared sensitive professional files and 11% of respondents shared sensitive personal files such as financial or medical information. The authors found that people utilise various methods to control access to their sensitive files, some are technical (passwords, permissions) and others are socially-controlled such as hiding files.

Unlike the study of Voida et al. [20] and Whalen et al. [23, 9] which focused on subjects within a single organisation, all of whom had access to similar, established file sharing methods, Dalal et al. [24] conducted in-depth interviews with respondents across various domains in their homes, home offices, or in cafes where people worked to examine how file sharing and access controls are used, not used or circumvented in order to get work done. The result of their study show that 80% of respondents shared files with overseas collaborators or clients in Europe and the Asia-Pacific region and 100% shared files with colleagues across the US. Their results showed differences between personal and professional sharing as they found that people in professional sharing concentrate on sharing files that are related to project work, such as shared documents included technical specification, meeting minutes and action items, proposals, reports. On the other hand, they found that people in personal sharing concentrate on sharing their experiences with others, and the content being shared (primarily multimedia) relational in nature, such as sharing photographs with with family members who live overseas. Email was used by all the respondents of their survey and 80% of them used various social software such as wiki, blogs, social networking sites (including MySpace and Facebook), public websites for sharing images and multimedia files (including Flickr, YouTube), and online forums and games. Moreover, their respondents made distinctions between two type of sharing which are sharing with oneself and sharing with others. Sharing files with oneself is very useful as it allows people to synchronise their activities regardless of their location, accessibility, or what devices are at hand. They found that USB drives and email are convenience and preferred methods for sharing with oneself. Analysing their results, they derived a set of design criteria for more effective file sharing system [24].

In contrast to previous studies which have focused on asking users themselves to report on how they share and protect files, Smetters and Good [25] conducted an automated survey of access control in a medium-sized corporation to collect behavioural data over time by analysing digital record of actual user behaviour as they believe that users' self-descriptions of their own behaviour can be incomplete or inaccurate. They used automated data mining to examine how users in a medium-sized corporation utilise two common access control features: the definition of access control groups, and the permissions settings, or ACLs, that users set on folders and documents.

They found that access control polices which are applied by users to their content are quite complex. Based on the results of their study, they derived a number of suggestions for the design of both access control systems themselves, and the interfaces used to manage them [25].

Mazurek et al. [26] conducted semi-structured interviews with 33 non-technical computer users in 15 households to examine the current access control attitudes, needs, and practices of home users when they share files inside and outside their homes. They found that people utilise a wide range of measures to restrict access to their files, some of them are standard access-control tools while others are ad-hoc tools. These tools are the same as those reported in [23] which are user accounts, password, encryption, limiting physical access to devices, and hide and delete sensitive files. They found that people have complex policies that ever-changing over time which are inadequately addressed in current file sharing and access control methods; a finding supported by Olson et al. [21, 22], Whalen et al. [23, 9], and Voida et al. [20]. Based on the results of their study, they have generated several guidelines for developers of access control systems aimed at home users.

Hart et al. [27] surveyed 23 blogging and social networking sites such as Blogger, Facebook, Flickr, YouTube, and MySpace to determine what access control and privacy features are currently available. They found that a lot of content-sharing sites provide primitive access control mechanisms which make a file entirely private or public while others allow more flexible control by offering private/friends/public access control model. The authors asserted that these models failed to support people's needs, and thus, proposed a method of access control for content-sharing sites that specify access control polices in terms of the content being mediated. Whalen et al. [28] pointed out that a potential solution for file sharing problems, such as exposing sensitive files accidentally, is to provide the user with clear information about file sharing settings and activities. Therefore, they explored existing research on awareness in collaborative environments, and used it to develop a framework for file sharing awareness. The authors used this awareness framework to develop a prototype for a file manager that facilitates file sharing by making sharing activity and settings more visible to the user.

Table 1 summarises the results of the above studies of file sharing with respect to answering the following fundamental questions: with whom the file is shared, what type of file is shared, how the file is shared and protected. The previous studies investigated these questions in details and provided valuable answers which could lead to better design of file sharing methods and access control models. However, the question of how the file is shared has been answered improperly. They merely answered the question of how people share their files by enumerating the methods of sharing files that people utilised. Such answers are applicable to the question of what methods people utilise to share their files rather than how the files are shared as we believe that the files can be shared in different ways using the same method.

3 CLASSIFYING the INSIDER PROBLEM

By surveying the previous work of insider security, we argue that the insider problem is significant and that no single definition can encompass the problem as a whole, which most researchers attempt to do. In the literature, insiders have always been defined and differentiated from outsiders by either being inside the network perimeter, trusted, authorised, or knowledgeable of the information system. Definitions based on these factors are either ambiguous or insufficient. To make progress and find a solution to the insider problem, we suggest that the problem should be classified into several categories which can be defined, studied and solved independently and which later can be combined to solve the problem as a whole. There are three factors which play an important role in classifying the insider problem which are: the type of activity that deals with an asset in an organisation; the type of asset that

Table 1: Summary of previous studies on file sharing

	With whom the file is shared	What type of file is shared	How the file is shared	How the file is protected
Olson et al. [21]	-The public, co-workers, managers and trusted co-workers, family and spouse.	-Email content, credit card number, transgression, work related documents, work email and desk phone number.	-	-
Voida et al. [20]	-Similar to Olson et. al.-With an average of 7 individuals or group	-34 different types of files e.g. business documents, paper drafts, music, ideas, schedules, and TV show	-Email (43%), shared network folders (16%) and posting content to a web site (11%)	-
Whalen et al. [9]	-Over 69% shared with two to four groups such as friends, family, research group, general public and colleagues. -25% shared with five to twenty groups.	-Only focused on sensitive files, such as email, personal financial or medical information, professional data or documents of an organisation, professional data or documents governed by law.	-Email (42%), shared network folders (14.7%), peer-to-peer program (10.3%) and file copy protocol (10.3%)	Various methods to control access to their sensitive files, some are technical (passwords, permissions) and others are socially-controlled such as hiding files.
Whalen et al. [23]	-	-	-Email (98%), shared network folder (55%), commercial content management systems (25%) and portable devices (25%)	Passwords; permissions/ access control lists; physical controls (e.g., safeguard in office or on person); encryption; obscurity (e.g., given files innocuous names, hidden directories); and deleting/relocating sensitive files.
Dalal et al. [24]	-With employees in professional sharing -With friends and family in personal sharing.	-In professional sharing: revolve around project work such as technical specifications, meeting minutes, and action items, proposals, reports.-In personal sharing: revolve around multimedia relational in nature such photograph and video.	Email (100%), - 80% used a wide variety of social software, such as wikis, blogs, social networking sites (including MySpace and Facebook) hosted services (such as Yahoo! Briefcase) public websites for sharing image and multimedia files (including Flickr and YouTube) and online forums and games.	-
Mazurek et al. [26]	-Family, friends, co-workers and strangers.	-Music, photo, video, private documents, school work, work files, and other personal documents.	-	-User accounts, password, encryption, limiting physical access to devices, and hide and delete sensitive files.

needs to be protected; and the type of attack that targeted the asset.

The activity. The activities are identified by the organisation for its partners, contractors, and employees to perform a particular job and might be different from one organisation to another. The activity will differentiate insiders from outsiders as an insider will be a person who is a legitimately given an activity by an organisation to perform a particular job. Therefore, the activity will lead to identifying who is the insider and what the insider is doing. The type of activity that insiders perform in an organisation are various and organisation-specific. Examples of activities that are given to insiders are file sharing, updating customer information, installing software to organisation's devices, setting up organisation's network, provisioning authorisation credentials to organisation's employees, etc.

The asset. The assets that need to be protected are identified by an organisation based on a clear description of activities in the organisation, such that each activity will involve one or more assets to deal with. For example, if an activity in an

organisation is employees sharing files with each other, the asset will be the file being shared which contains sensitive information. Another examples of activities and assets are an IT administrator who provisioning authorisation credentials to an organisation's employees where the asset here is the authorisation credential, a software developer who writes software scripts to an organisation computer where the asset can be the software itself or the computers that run the scripts, a network administrator who sets up the organisation's network and maintains it where the asset is the network.

Generally, the assets can be of three types which are the network which connects devices together, the devices which contains the data, or the data itself.

The attack. The attacks that targeted the asset can be generally of three types which are availability attacks, confidentiality attacks and integrity attacks, each of which can be performed in different ways which might require either physical security or IT security. Choosing which type of attacks to prevent is determined by the type of protection required for the chosen asset. For instance, if the asset is the network which needs to be available all the time, availability attacks should be prevented. On the other hand, if the asset is data that needs to be secret, confidentiality attacks should be prevented and so on. Therefore, the asset will determine which type of attacks should be prevented.

Based on these three factors, we can define the insider precisely as a person who is legitimately given an activity by an organisation to deal with the organisation's assets, and define the insider problem as particular types of attacks that performed by insiders on particular types of assets of an organisation during particular types of activities. Therefore, we can classify the insider problem into several categories based on these three factors such that each particular type of attack by insiders on a particular type of asset of an organisation during a particular type of an activity will result in a unique class of the insider problem which can be defined, studied and solved independently. For example, one class of the insider problem is preventing confidentiality attacks on sensitive files by employees when they share them with each other. Another class might be preventing availability attacks on an organisation's network by IT administrators when they maintain it, or preventing integrity attacks on customers information by employees when they update them etc.

Our concern in this paper is not to classify the insider problem thoroughly, rather we have provided an approach for such classification. However, we are interested in one class of the insider problem which is related to file sharing. The activity in our class is file sharing, the asset is the files being shared, and the attacks we are concerned with are confidentiality and integrity attacks. Thus, we define our class of the insider problem as preventing confidentiality and integrity attacks on sensitive files when employees share them with each other.

Since file sharing is not only an activity that is performed by an organisation's employees but also it is an activity that can be performed among friends, family members, or colleagues, we will look at this class of the insider problem from broader perspective to include any individuals performing such activity. In other words, the insider in our class will be the recipients whether those recipients are employees, friends, or family members.

4 PROTECTING SHARED FILES

Although we defined our class of the insider problem in the previous section, the attacks we are concerned with (i.e. confidentiality and integrity attacks) are still vague. These attacks can be performed in different ways, which in turn requires different types of protections. Claiming that a particular protection mechanism can protect the confidentiality of the files is not enough. Instead, one should claim that a particular protec-

tion mechanism can protect the confidentiality of the files under specific kinds of attacks. Therefore, In order to protect the confidentiality and the integrity of the shared files from the insiders (i.e. recipients), the different attacks and misuses that affect the confidentiality and the integrity of shared files must be identified.

Generally, protection of the shared files can be realised from two different angles: protecting the shared files while in transit, and protecting the shared files when they are received by the recipients. In this section we characterise the protections required by the shared files against different types attacks and misuses that are performed during the activity of file sharing.

4.1 Protecting the Shared Files in Transit

This type of protection prevents attacks on the file while it is transferred from the owner to the recipients. We divided these attacks into confidentiality attacks and integrity attacks as follows:

Confidentiality attacks. These attacks lead to the disclosure of the shared files to unauthorised users and can performed in two ways. First, someone eavesdrops or monitors the communication between the owner and the recipient to obtain knowledge about the files. We refer to such attacker *Interceptor*. Second, someone pretends to be the original recipient to deceive the owner and obtain the files. We refer to such attacker *Masquerader*. Therefore, there should be two types of protections to prevent unauthorised disclosure of the shared files in transit as follows. Protecting the confidentiality of files from interceptor and protecting the confidentiality of files from Masquerader.

Integrity attacks. These attacks lead to unauthorised modification to the shared files by unauthorised users. The attacker in such attacks pretends to be the original owner to deceive the recipient by sending them files as if they were originated by the original owner. These files can either be an entirely new files or modified version of the original files. We refer to such attacker *Masquerader*. Therefore, there should be one type of protection to prevent unauthorised modification of the shared files in transit which is protecting the integrity of files from Masquerader.

4.2 Protecting the Shared Files at the Recipient

This type of protection prevents misuses on the file after it has been received by legitimate recipients. These misuses can affect the confidentiality and integrity of the files. Such misuses can be committed by three different entities which are *Malicious* recipients, *Naive* recipients or *Masqueraders*. *Malicious* recipients are untrusted legitimate recipients who deliberately misuse the shared files. *Naive* recipients are trusted recipients who accidentally misuse the shared files. *Masqueraders* are unauthorised users who accidentally acquire a device of a trusted legitimate user which contains the shared files and misuse these files. Therefore, misuses can be deliberate which are committed by *Malicious* or accidental which are committed by *Naive* or *Masqueraders*.

Protection against *Malicious* and *Naive* recipients are different from protection against *Masquerader*. *Malicious* and *Naive* recipients are already allowed to view the files, therefore, confidentiality of the files is achieved by not allowing them to redistribute the files to unauthorised users. Also, they may or may not be allowed to modify the files, therefore, integrity of the files is achieved by not allowing them to modify it in an unauthorised manner. On the other hand, *Masqueraders* are unauthorised users, therefore, confidentiality is achieved by not allowing them to view or redistribute the files, and integrity is achieved by not allowing them to modify the files.

Moreover, protection against *Naive* recipients is different from protection against *Malicious* recipients. The former is trusted to not redistribute or

modify the files in an unauthorised manner, while the latter is untrusted and might strive to circumvent any protection to misuse the files. Therefore, we divided misuses which can be committed by the three entities broadly into confidentiality misuses and integrity misuses as follows:

Confidentiality misuses. Confidentiality misuses are those misuses which lead to the disclosure of the shared files to unauthorised users and which can be done in two ways. First, the shared file can be copied and sent to an unauthorised user through a file sharing method. Second, the device of a legitimate recipient which contains the shared file can be acquired by an unauthorised user, which we refer here to as a Masquerader, who discloses the shared files.

In the first case, the file can be redistributed in three ways. First, the file can be redistributed accidentally by a Naive legitimate recipient. Second, the file can be redistributed deliberately by a Malicious legitimate recipient. Third, the file can be redistributed accidentally by a Masquerader who found a device of a legitimate recipient unattended. In the second case, the file can be disclosed to Masqueraders in two ways. First, an unauthorised user steals the device of a Naive legitimate recipient. Second, a Malicious recipient lends his device to an authorised user.

Therefore, there should be five different types of protections to prevent unauthorised disclosure of the shared files at the recipients as follows. Protecting the confidentiality of files from accidental redistribution by Naive; protecting the confidentiality of files from accidental redistribution by Masquerader; protecting the confidentiality of files from deliberate redistributions by Malicious; protecting the confidentiality of files from accidental disclosure by Naive to Masqueraders; protecting the confidentiality of files from deliberate disclosure by Malicious to Masqueraders. Since the last two types of protection have similar impact which is disclosing the file to Masqueraders, we refer to them as protecting the confidentiality of files from accidental or deliberate disclosure to Masqueraders.

Integrity misuses. Integrity misuses are those misuses which lead to unauthorised modification to the shared files. Such unauthorised modification can be either modifying the shared files that do not allow any modification or modifying the shared file, that allowing partial modification, in an unauthorised manner. In both cases, the file can be modified in three ways. First, the file can be modified accidentally by a Naive legitimate recipient. Second, the file can be modified deliberately by a Malicious legitimate recipient. Third, the file can be modified accidentally by a Masquerader who found a device of a legitimate recipient unattended.

Therefore, there should be three different types of protections to prevent unauthorised modification of the shared files at the recipients as follows. Protecting the integrity of files from accidental modification by Naive; protecting the integrity of files from accidental modification by Masqueraders; protecting the integrity of files from deliberate modification by Malicious.

Below we classify the aforementioned protections into two types which are protection of files in transit and protection of the files at the recipients.

Protection of files in transit: this can be further divided into confidentiality protection and integrity protection.

- Confidentiality protection
  - Protecting the confidentiality of files in transit from *interceptor*
  - Protecting the confidentiality of files in transit from *Masquerader*
- Integrity protection
  - Protecting the integrity of files in transit from *Masquerader*

Protection of files at the recipients: this can be further divided into protection against accidental misuses when sharing with trusted recipient and protection against deliberate misuses when sharing with untrusted recipient.

Accidental misuse: this can be further divided into accidental misuse of confidentiality and accidental misuse of integrity.

- Accidental misuse of confidentiality:
  - Protecting the confidentiality of files at the recipients from accidental redistribution by *Naive*
  - Protecting the confidentiality of files at the recipients from accidental redistribution by *Masquerader*
  - Protecting the confidentiality of files at the recipients from accidental disclosure to *Masquerader*
- Accidental misuse of integrity:
  - Protecting the integrity of files at the recipients from accidental modification by *Naive*
  - Protecting the integrity of files at the recipients from accidental modification by *Masquerader*

Deliberate misuse: this can be further divided into deliberate misuse of confidentiality and deliberate misuse of integrity

- Deliberate misuse of confidentiality:
  - Protecting the confidentiality of files at the recipients from deliberate redistribution by *Malicious*
  - Protecting the confidentiality of files at the recipients from deliberate disclosure to *Masquerader*
- Deliberate misuse of integrity:
  - Protecting the integrity of files at the recipients from deliberate modification by *Malicious*

4.3 Summary

Figure 1 illustrates eleven types of protections that might be required to protect the files in transit and at the recipients. Protections of files in transit are concerned with preventing external attacks while protections of files at the recipients are concerned with preventing insider attacks.

The characterisation of the protections required by the shared files at the recipients, illustrates the different ways of how files can be misused by different types of insiders. This characterisation makes it clear which type of insider misuse needs to be prevented in a particular sharing scenario. For instance, misuses by Masqueraders need not to be prevented if the machine containing the file resides in a locked room where unauthorised users cannot access. Also, deliberate misuses by Malicious insiders need not to be prevented if the file is shared with trusted recipients. A major advantage of this characterisation is the avoidance of the chaos exists in the literature with respect to distinguishing insider attacks from external attacks, and between insiders attacks themselves. We listed different protections requirements to prevent different insiders misuses so that one can select the desired protection requirements for a particular sharing scenario and develop a mechanism to enforce it.

From our point of view, in order to protect a shared file against insiders attacks, it should only be shared with trusted insiders. Protecting the shared files from untrusted insiders who might strive to circumvent the protection is a dilemma for two reasons. First, each system has its own vulnerabilities and there is no system without vulnerabilities. Research efforts have proven that there is no system 100% secure against all deliberate attacks or misuses [29]. A brief look at the approach taken to protect commercial content, justifies this principle. Commercial content is protected by the use of Digital Rights Management systems that dictate how the content must be used by each individual. Although these systems are in place to protect commercial content, the content can still be obtained illegally in unprotected form. Second, the easiest way to circumvent any protection system used to protect confidential files is by exploiting the analog hole. All digital content must eventually be converted to human-perceptible form, known as the analog form, to be consumed by users. Once the digital content is converted to analog form, it will be in an unprotected form, and thus, it will be susceptible to unauthorised uses [30].

However, sharing a file with trusted insiders without any protection in place is risky. Even if insiders are trusted to not violate the content policy deliberately, there is a chance of accidental violation. According to a survey conducted by In-

fosecurity Europe and PwC on 1,402 UK companies, 36% of the worst security breaches in the year were caused by inadvertent human error [31]. Also, AngloSec conducted a survey on 197 network, security, and compliance professionals, and found that the greatest security concern is employees accidentally jeopardising security through data leaks or similar errors [32]. Therefore, there should be appropriate level of protection that prevents accidental misuses on the shared files by trusted insiders who do not misuse the files intentionally.

5 CHARACTERISING FILE SHARING

Although the different types of misuses on the shared files and the protection requirements are identified in the previous section, the activity of file sharing is still ambiguous. Some people conceive the activity of file sharing is to send an email attachment, while others conceive it as to make files available to others through peer-to-peer networks. Designing a mechanism that provides the various types of protection without taking into account how the activity of file sharing is performed, might be useless. This is pointed out by previous studies, where they showed that some people might avoid secure methods of file sharing and utilise insecure methods because it is more suitable for the task of sharing, although security is a concern for them. For instance, employees in organisations might be forced to utilise particular sharing methods because they are secure. However, since these methods have been built with only security in mind, they might not be suitable for the task of sharing that employees need to get their job done. Hence, employees usually tend to utilise other sharing methods that might be insecure to avoid obstacles found in secure methods, and hence, putting organisation's confidential files at risk. To avoid such problem, the different ways of performing the activity of file sharing should be considered when designing a protection mechanism, so that the protection mechanism will not only protect the shared files but also allow various task of sharing to be performed.

The activity of file sharing is performed by individuals for various purposes (e.g. professional or personal). The purpose of performing the activity of sharing makes it obvious to whom the files should be shared with (e.g. family, friends, colleagues, or anyone), which type of file to be shared (e.g. music, photo, video, business documents, etc.), and which method of sharing to be utilised that is possibly the most satisfied to the sharing purposes (e.g. secure, convenient, available to everyone etc.). These factors are discussed in literature and summarised in Table 1.

However, there two factors that are clearly affected by the purposes of sharing and which are overlooked by previous studies. These factors are file propagation and access which can be different based on the sharing purpose. To the best of our knowledge, we are not aware of any work that characterises the activity of file sharing based on these two factors. Therefore, In this section we characterise the activity of file sharing based on how files can be propagated and how files can be accessed after their propagation.

5.1 How files are propagated

5.1.1 Publish vs. Share:

Files can be propagated in two main ways depending on their sensitivity. Confidential files are only released to selected individuals while non-confidential files are released to everyone. Available file sharing methods can either allow people to share files with selected individuals (suitable for confidential files) or allow people to share files with everyone (suitable for non-confidential files). A few file sharing methods provide both options. We will use the term *share* to refer to a file that is released to selected individuals and the term *publish* to a file that is released to everyone. Publishing or sharing files can be performed in different scenarios. Therefore, we use the following terminology to characterise the different ways of how files are shared and published.

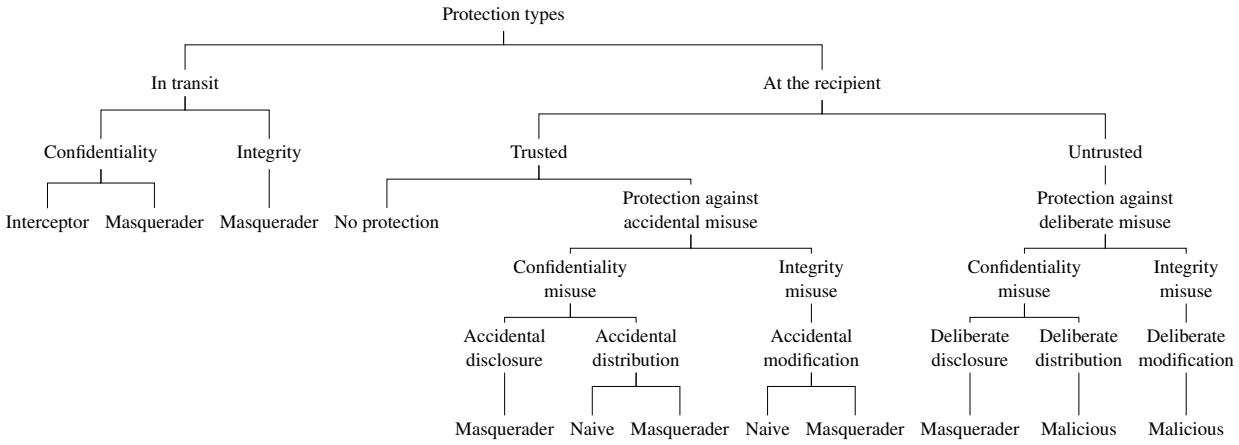

Figure 1: Types of protection of the shared files

Terminology:

- $\bar{O}$: a particular owner of files who might or might not be known in advance.
- $I\bar{n}G$: a set of owners of files whom their numbers and identities are known in advance and share their files with each other.
- $\bar{G}$: a set of owners of files whom their numbers and identities are known in advance and do not share their files with each other.
- $\bar{M}$: a set of owners of files whom their number and identities are not known in advance and do not share their files with each other.
- O: a particular recipient who is known in advance.
- G: a set of recipients whom their number and identities are known in advance and whom receive the same copies of the shared files.
- M: a set of recipients whom their numbers and identities are not known and whom receive the same copies of the shared files.

In general, files can be released either to O, G or M. However, the received files by the recipients who can be O, G and or M, might belong to $\bar{O}$, $I\bar{n}G$, $\bar{G}$ or $\bar{M}$. Therefore, including $I\bar{n}G$ as a category of sharing we have 11 different categories that can describe all the possible ways of how files are shared or published as described below.

- $\bar{O} \rightarrow O$ (OneToOne): This describes a situation when a particular owner of files wants to share his files with a particular recipient who is known in advance. For example, Alice wants to share her file only with Bob but no one else.
- $\bar{O} \rightarrow G$ (OneToGroup): This describes a situation when a particular owner of files wants to share his files with a set of recipients whom their numbers and identities are known in advance, and whom receive the same copies of the shared files. For example, Alice wants to share her file only with her colleagues Bob, Carol, and Dave but no one else.
- $\bar{O} \rightarrow M$ (OneToMany): This describes a situation when a particular owner of files wants to share his files with a set of recipients whom their numbers and identities are not known, and whom receive the same copies of the shared files. For example, Alice wants to share her file with everyone on the internet regardless of whom they are.
- $I\bar{n}G$ (InGroup): This describes a situation when owners of files whom their numbers and identities are known in advance want to share their files with each other. For example, Alice, Bob, and Carol want to share their files only with each other but no one else.
- $I\bar{n}G \rightarrow O$ (InGroupToOne): This describes a situation when a set of owners of files whom their numbers and identities are known in advance and share their files with each other, want to share their shared files with a particular recipient who is known in

advance. For example, Alice, Bob and Carol who are sharing their files with each other want to share these shared files only with their colleague Dave but no one else.

- $I\bar{n}G \rightarrow G$ (InGroupToGroup): This describes a situation when a set of owners of files whom their numbers and identities are known in advance and share their files with each other, want to share their shared files with a set of recipients whom their numbers and identities are known in advance, and whom receive the same copies of the shared files. For example, Alice, Bob and Carol who are sharing their files with each other want to share these shared files only with their colleagues in the same department but no one else.
- $I\bar{n}G \rightarrow M$ (InGroupToMany): This describes a situation when a set of owners of files whom their numbers and identities are known in advance and share their files with each other, want to share their shared files with a set of recipients whom their numbers and identities are not known and whom receive the same copies of the shared files. For example, Alice, Bob and Carol who are sharing their files with each other want to share these shared files with everyone on the internet regardless of whom they are.
- $\bar{G} \rightarrow O$ (GroupToOne): This describe a situation when a set of owners of files whom their numbers and identities are known in advance and whom do not share their files with each other, want to share their files with a particular recipient who is known in advance. For example, Alice, Bob and Carol who work in the same company want to share their files only with Dave who is their employer but not with each other or anyone else.
- $\bar{G} \rightarrow G$ (GroupToGroup): This describe a situation when a set of owners of files whom their numbers and identity are known in advance and whom do not share their files with each other, want to share their files with a set of recipients whom their numbers and identities are known in advance, and whom receive the same copies of the shared files. For example, Alice, Bob and Carol who work in the same company want to share their files only with employees of the HR department but not with each other or anyone else.
- $\bar{M} \rightarrow O$ (ManyToOne): This describes a situation when a set of owners of files whom their numbers and identities are not known in advance and whom do not share their files with each other, want to share their files with a particular recipient who is known in advance. For example, applicants to a particular job want to share their documents files only with Alice who is the employer but no one else.
- $\bar{M} \rightarrow G$ (ManyToGroup): This describes a situation when a set of owners of files whom their numbers and identities are not known in advance and whom do not share their files with each other, want to share their files with a set of recipients whom their numbers and identities are known in advance, and whom receive the same copies of the shared files. For example, applicants to a particular job want to share their documents files only with Alice, Bob and Carol, who are the employees responsible for recruiting new staff, but no one else.

Figure 2 illustrates these categories and classifies them to either publish or share. Note that we excluded situations that do not make sense such as $M' \rightarrow M$ and $G' \rightarrow M$, since any of the owners can be of the recipients and vise versa.

5.1.2 Static vs. Dynamic vs. Transfer mode

In any of the categories of files propagation described above, files can be moved from an owner to a recipient differently. For instance, the original file can be moved physically as an object in real world, leaving no copies behind, or a copy of the original file can be moved to the recipient. In the latter case, the moved copy can be either a dynamic or a static. Below we describe each one of them.

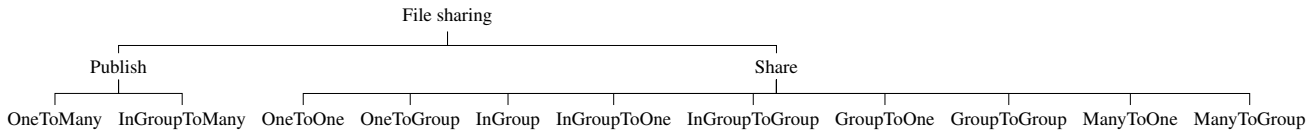

Figure 2: How files can be published and shared

Publishing or sharing in Static Mode: Publishing or sharing in a static mode describes a scenario where independent copies of the original file are moved from the owner to the recipients. Any changes made to the copies of the original file by the recipients or to the original file by the owner do not reflect on one another. It is useful when the owner of the file does not want to receive a new version of the published or shared files from the recipients or update the copies that the recipients have. An example of a method that allows sharing in a static mode is an email attachment where neither the owner nor the recipients can observe changes made on the shared files by others.

Publishing or sharing in Dynamic Mode: Publishing or sharing in a dynamic mode describes a scenario where copies of the original file that are linked to the original file are moved from the owner to the recipients. Therefore, any changes made to the copies of the original file by the recipients or to the original file by the owner do reflect on one another. It is adequate for a collaborative project where a group of members may work on a set of documents collectively. An example of a method that allows sharing in a dynamic mode is Dropbox where a file can be shared and updated by the owner or the recipients such that both can observe changes made to the shared files.

Publishing or sharing in Transfer Mode: Publishing or sharing in a transfer mode describes a scenario where the original file is moved, leaving no copies behind, from the owner to the recipients. The file is treated as a real world object that cannot exist at two places at the same time. Hence, in this mode, releasing a file to more than one recipient, requires the file to be held by one recipient at a time. We are not aware of any method meets this mode of publishing or sharing.

5.1.3 Distributed memory vs. Shared memory

Files can be moved from the owner to the recipients directly to their devices or indirectly to a location where recipients can access (e.g. server). We refer to the former as sharing or publishing in distributed memory (DM), and the latter as sharing or publishing in shared memory (SM). A file that is shared or published in DM, will be stored in each recipient device, allowing them to access the file when they are off-line. On the other hand, a file that is shared or published in SM, will be stored in a central location where recipients must access each time they need to access the file. Thus, unlike DM, a file in a SM requires the recipients to be online to get access to the file.

SM best suited for a file that is shared or published in a dynamic mode. Since the owner and the recipients have access to the same copy of a file, changes made to the file will be observed by others without the need to move copies of the file with the new changes to others. An example of a file that is published in a dynamic mode in SM is Wikipedia page. Also, SM can be suitable for transfer mode, such that a file in the central location can be accessed only by one recipient, who the file is transferred to, at a time. However, sharing or publishing in a static mode is not suitable for SM, particularly when the recipients are allowed to change the files. This because recipients of a file that is published or shared in a static mode are each intended to have a indepen-

dent copy of the original file that can be changed without affecting the copies that other recipients have. Hence, this cannot be achieved in SM, since all recipients access the same copy of the file simultaneously.

On the other hand, DM can be suitable for all sharing or publishing modes (i.e. static, dynamic, and transfer). In static mode, independent copies of the original files are moved to the recipients devices, while in transfer mode, the original file is moved to one recipient device at a time. In case of a dynamic mode, copies of the original files are also moved to the recipients devices, however, the moved copies are linked to the original file, so that any changes made on them will be communicated to other copies.

Table 2 illustrates 33 types of files propagation. Each cell in the table marked with letter T indicates a way of propagating a file. For instance, OneToOne sharing can be performed in static (DM), dynamic (DM or SM) or transfer (DM or SM) mode. In other words, an independent copy of the original file can moved to one particular recipient device (static DM), a linked copy to the original file can be moved to one particular recipient device (dynamic DM) or a copy of the original file is moved to a location where one particular recipient can access (dynamic SM), or the original file is moved to one particular recipient device rather than a copy (transfer DM), or moved to a location where one particular recipient can access (transfer SM).

Table 2: Types of propagation

Types of propagation	Static (DM)	Dynamic (DM or SM)	Transfer (DM or SM)
OneToOne	T	T	T
OneToGroup	T	T	T
OneToMany	T	T	T
InGroup	T	T	T
InGroupToOne	T	T	T
InGroupToGroup	T	T	T
InGroupToMany	T	T	T
GroupToOne	T	T	T
GroupToGroup	T	T	T
ManyToOne	T	T	T
ManyToGroup	T	T	T

5.2 How files are accessed:

Once files are propagated, recipients need to access them. There are only two types of access that the recipients might need which are read and write access. The former allows them to read the file while the latter allows them to append or remove content from that file. However, an owner of a file might want to restrict these types of access based on the sharing or publishing purpose. For instance, the owner might want the recipients to: *a)* read but not edit the file. *b)* not read but edit the file by appending new content only. *c)* read and edit the file by appending or removing content. *d)* not read and not edit the file but just hold it (e.g. cloud storage providers). We refer to these access types as ReadOnly, WriteOnly, ReadWrite, and NoReadOrWrite, respectively.

Additionally, an owner of a file might find these types of access not restrictive enough for some sharing or publishing purposes. For instance, the owner might know that the recipients need only to read or edit the file *a)* for a limited number of times (e.g. only once). *b)* for a limited period of time (e.g. for three days starting from 1/9/2014). *c)* on a specific time (e.g. only Monday from 9am - 3pm). *d)* at a specific location (e.g. only in London). Therefore, these can be used as restrictions to further control the different access types mentioned above.

Table 3: Types of access

Types of access	Ln [1]	Lp [2]	St [3]	Sl [4]
ReadOnly	T	T	T	T
WriteOnly	T	T	T	T
ReadWrite	T	T	T	T
NoReadOrWrite	F	T	F	T

Table 3 illustrate 14 types of access the recipients might have. Each cell marked with letter T indi-

[1]Limited number of times
[2]Limited period of time
[3]Specific time
[4]Specific location

Table 4: Types of files propagation and access

How Files are propagated and accessed		ReadOnly				WriteOnly				ReadWrite				NoReadOrWrite			
		Ln	Lp	St	Sl	Ln	Lp	St	Sl	Ln	Lp	St	Sl	Ln	Lp	St	Sl
Share	Static	T	T	T	T	F	F	F	F	T	T	T	T	F	T	F	T
	Dynamic	T	T	T	T	T	T	T	T	T	T	T	T	F	T	F	T
	Ttransfer	T	T	T	T	F	F	F	F	T	T	T	T	F	T	F	T
Publish	Static	T	T	T	T	F	F	F	F	T	T	T	T	F	F	F	F
	Dynamic	T	T	T	T	T	T	T	T	T	T	T	T	F	F	F	F
	Transfer	T	T	T	T	F	F	F	F	T	T	T	T	F	F	F	F

cates a useful type of access that owners of files might want the recipients to have. Table 2 shows two unuseful types of access which are (NoReadOrWrite,Ln) and (NoReadOrWrite,St). This because restricting NoReadOrWrite access for a limited number of time or for a specific time does not make sense. However, the other two restrictions on NoReadOrWrite access (i.e. Lp and Sl) might be useful for some scenarios of sharing. For instance, an owner might want to share files with a cloud storage providers provided that the files are kept in the provider servers that are located at a particular geographical area. Another owner might want the files to be kept at the provider servers until a particular point of time after which the provider will not be authorised to keep the files in the servers.

It should be noted that recipients can only have one type of access, however, various restrictions can be used to restrict that type of access. For instance, an owner might want the recipients to have the following type of access: (ReadOnly, Lp, Sl) which allows the recipients to read the file for a limited period of time and at specific location. These two restrictions should be satisfied in order for the recipients to read the file. Also, there is a difference between having no type of access at all and having NoReadOrWrite type of access. The former disallows holding the file, while the latter allows holding the file but not reading or editing it.

Table 4 combines the different types of files propagation and access and identifies the useful combinations of these types. Each letter T in the table identifies a useful a way of propagating and accessing a file by the recipients. The term *Share* and *Publish* can be replaced with any of the categories of files propagation depicted in Figure 1. As shown in the table, not all types of access are suitable for all types of files propagation (i.e. not all combinations of files propagations types and access types are applicable). For instance, it is not sensible for the recipients to have WriteOnly type of access for a file that is shared or published in a static or transfer mode. Although the recipients will be able to add content to the file, no one can observe this content. Also, it is not useful to publish a file with NoReadOrWrite type of access, since there is no need to release the file to everyone and not allowing them to read it or edit it.

6 TAXONOMY BASED on the CHARACTERISATION of FILE SHARING

Based on the characterisation of the activity of file sharing discussed in the previous section, we define a framework that can be used to classify the activity of file sharing in a systematic way. This framework is shown in Figure 3, will help to classify the activity of file sharing by distinguishing how files are propagated to and accessed by the recipients. Below is a brief description of the proposed framework.

The framework has a tree-based structure, where each level represents either a way of files propagation or files access. Paths of the tree are numbered. Therefore, specifying the path number for each level of the tree starting from the root down-

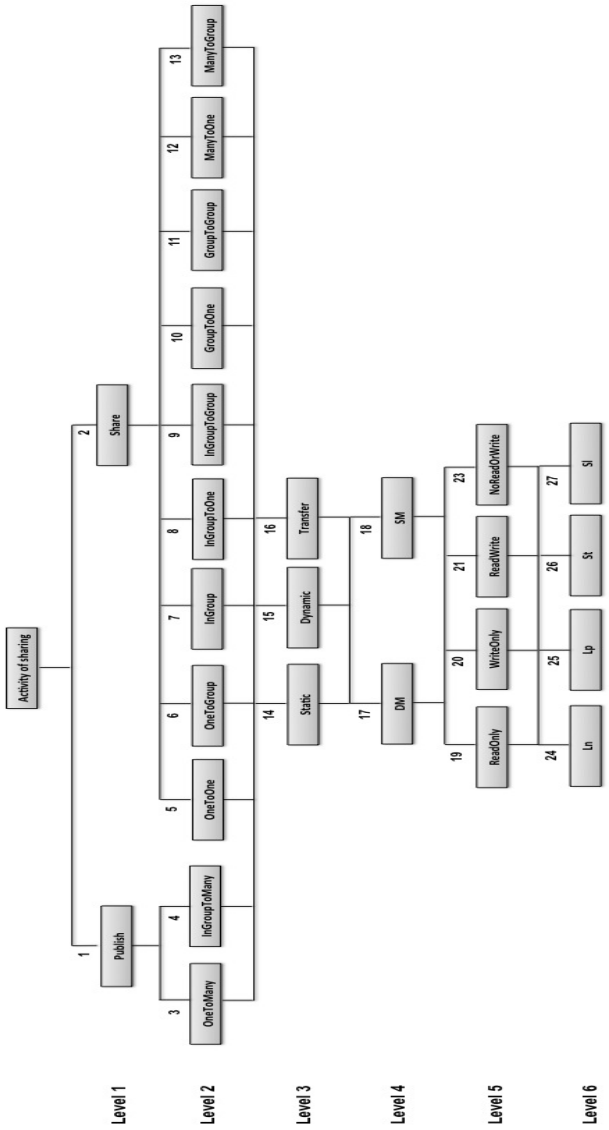

Figure 3: Framework for classifying the activity of file sharing

wards, will result in a unique class of the activity of file sharing. The first four levels after the root (i.e. paths from 1-18) represents types of files propagation, while the last two levels (i.e. paths 19-27) represents types of files access. At each level of the framework a unique choice has to be made. In this way every class of file sharing will form a single path in the tree. However, there is one exception: namely level six,"restriction over access types". Any class of file sharing can utilise none or multiple restrictions (e.g. Ln and Lp at the same time) over one type of access (e.g. ReadOnly) as described in the previous section.

Due to space limitations and to avoid redundant branches, not the entire tree is drawn. For instance, level two has eleven types to choose from, two types belong to path one, and nine types belong to path two. Each of these types has the same three possibilities for level three (i.e. Static, Dynamic and Transfer). Hence, at level three there are eleven identical groups of the three possible values. Therefore, to avoid using redundant branches, the types of level three are written once and can be used by all types of level two. On the other hand, each type of level one might have different possibilities of level five, and also from level three to six, each level might have different possibilities of its subsequent level. However, for simplicity and to save space, the maximum possibilities that each level should have is specified while table1, 2 and 3 can be utilised to help exclude those unuseful classes of sharing. For example, as shown in table 3, NoReadOrWrite access is not suitable for publishing. Hence, if the path 1 is chosen, regardless of the paths chosen for level 2, 3, and 4, path 23 in level 5 should be excluded. Otherwise, this class will be unuseful. Another example is when path 23 is chosen in level, paths 24 and 26 should be excluded as restricting the number of time and specific time on NoReadOrWrite access is unuseful.

6.1 Utilising the taxonomy of file sharing

The framework, depicted in Figure 3, can be utilised in two ways. First, the framework can be applied to classify the activity of file sharing, by showing different classes of how owners might want their files to be propagated and accessed for different sharing scenarios. Second, it can be applied to classify available file sharing methods, by showing which method provides which class of the sharing activity. Below two examples are discussed to illustrate how the framework can be utilised.

Example 1: classifying the activities of file sharing in an organisation. Alice has a start-up company consists of several departments

which are Human Resource, Marketing, Production, Finance. Each department contains several employees. Employees within the same department and between different departments need to share files with each other to get their job done. Therefore, Alice wants to define how the activity of file sharing should be performed among employees.

Alice knows the Marketing department is responsible for dealing with customers. The Marketing department should send surveys to customers, however, Alice wants the surveys that should be sent to customers to be approved by the Manager of the department who is then responsible to move copies of the surveys to customers devices, so that customers can access them on their devices to read and edit them and move these copies back to the department if they are willing to do so. Hence, Alice specified the following class of file sharing for this department: 1-3-14-17-21.

Also, Alice knows that employees of the Production department, each should write a report and share it with other employees in the same department, so that each will be aware of others work and able to modify other reports in case mistakes are found. Alice wants employees to view and edit others reports when they are in their offices and during working hours. Hence, Alice specified the following class of file sharing for this department: 2-7-15-18-21-(26 + 27).

With respect to the Finance department, Alice knows that employees of this department write reports that should be viewed by employees of the Human Resource department in order for them to make a decision for recruiting new employees. However, Alice wants these reports to be approved by the Manager of the department who then is responsible to move copies of the reports to the company's server where employees of the Human Resource department can access. This will allow these reports to be updated by the Manager of the Finance department while employees of the Human Resource department will be able to view up to date reports. In addition, Alice wants employees of the Human Resource department to view these reports for a limited period of time and during working hours. Hence, Alice specified the following class of file sharing for this department: 2-6-15-18-19(25 + 26).

Finally, Alice who owns the company, needs to view a monthly reports written by each department manager. Alice does not want any manager to view reports written by other managers. Therefore, Alice specified the following class of file sharing between the managers and herself as follows: 2-10-14-17-19.

Example 2: Classifying file sharing methods There are various methods of file sharing exist today. Some of them have been designed merely for sharing files such as File Hosting Services, FTP, and Peer-to-peer file sharing, while others include file sharing as an added feature to their main purposes such as Emails and Social Networking Sites. Table 5 classifies some of the most popular file sharing methods based on the taxonomy described in the previous section. Each cell in the table shows which path the sharing method can take in each level of the framework. Below is a brief discussion of the classified methods in the table.

Email email is considered the most commonly-used method for sharing files. Although there is a few drawbacks of sharing files via an email such as limitation on file size, it is still popular method for sharing files at the present due to certain features. These features are the ease of use, the widespread availability and the suitability to various tasks. Almost anyone uses a computer owns an email account, and knows how to use it. Therefore, by using an email to share files, the user will avoid all the difficulties associated with other methods of file sharing such as ensuring that all recipients have the same method to be able to share the files or ensuring that all recipients know how to use the method of sharing especially if the method is quite complex and difficult to learn. Examples of emails are Hotmail,Yahoo, and Gmail.

- Level 1: since email requires an owner of a

Table 5: Classifying file sharing methods

File sharing methods	Emails	Peer-to-peer file sharing	Anonymous FTP	None-anonymous FTP	Cloud-storage services
Level 1	2	1	1	2	1,2
Level 2	5, 6, 7, 8, 9, 10, 11, 12, 13	3	3, 4	5, 6, 7, 8, 9	3, 4, 5, 6, 7, 8, 9, 10, 11
Level 3	14	14	14	14	14, 15
Level 4	17	17	17	17	17, 17
Level 5	21	21	21	21	19, 21
Level 6	-	-	-	-	-

file to enter the emails addresses of the recipients, which means that recipients should be known in advance, it is only suitable for sharing rather than publishing (i.e. path 2 in Figure 3).

- Level 2: email allows an owner of a file to share the file with a particular person or with a group of people, hence, files can be shared as OneToOne or OneToGroup. A group of owners can share their files with each other by email, as well as sharing their shared files with another person or group. Therefore, files can also be shared as InGroup, InGroupToOne and InGroupToGroup. Also, email allows a group of owners who do not share their files with each other to share their files with a particular person or a group of people. This group might or might not be known in advance to the recipients. Therefore, files can be shared as GroupToOne, GroupToGroup, ManyToOne, ManyToGroup. Hence, all paths (i.e. 5,6,7,8,9,10,11,12,13) of level two are applicable for sharing files by emails.
- Level3: email allows an owner of a file to only send a copy of the file to the recipient rather than the file itself. The copy received by the recipient is not linked to the original, therefore, any changes to the copy by the recipient will not be reflected on the original file. Hence, email allows sharing files in static mode only (i.e. path 14 in Figure 3).
- Level 4: since the copy that is sent to the recipient must be stored at the recipient device in order to be accessed, email allows sharing files in distributed memory rather than shared memory (i.e. path 17 in Figure 3).
- Level 5: email provides only one type of access to the recipients which is ReadWrite which allows the recipients to read and edit the received files (i.e. path 21 in Figure 3).
- Level 6: email provides no restrictions on the type of access the recipients have.

Peer-to-peer file sharing Peer-to-peer (P2P) file sharing applications have gained much attention in recent years. As its name suggests, P2P file sharing applications utilise P2P network. Unlike client-server network, P2P network consists of multiple computers (nodes) that are able to act as client and server in the same time. For instance, a node in a P2P network can send request to another node in the network while responding to requests from other nodes. Therefore, in P2P file sharing applications, files are not uploaded to a central server, instead, they are scattered across users devices which each of which can act as a client and server simultaneously. Examples of P2P file sharing applications are Napster, LimeWire, Shareaza, Kazaa,and BitTorrent.

- Level 1: P2P file sharing requires an owner of a file to use a P2P client to register the file to P2P network. Once the file is registered to the network, other users who use clients that connect them to the same network will be able to search and download that file. Therefore, it is suitable for publishing rather than sharing.
- Level 2: P2P file sharing allows an owner of a file to share the file with everyone on the network, therefore, files can be published only as OneToMany.
- Level 3: P2P file sharing allows an owner

of a file to publish an independent copy of the file to the recipients. Hence, it allows publishing in static mode.
- Level 4: Since the sent copies to the recipients will be stored at their devices in order to be accessed, P2P file sharing allows publishing files in distributed memory rather than shared memory.
- Level 5: P2P file sharing provides only one type of access to the recipients which is ReadWrite which allows the recipients to read and edit the received files.
- Level 6: P2P file sharing provides no restrictions on the type of access the recipients have.

Anonymous FTP Anonymous FTP allows anonymous access to the uploaded files on the FTP sever to anyone with an FTP client or even through a web browser. Most anonymous FTP servers allow anonymous users to download files from the server but no one can update the directory except the owner of the directory.

- Level 1: Since the uploaded files to the server are publicly available to be accessed by anyone with an FTP client, anonymous FTP is suitable for publishing rather than sharing.
- Level 2: Anonymous FTP allows a particular owner of files to publish the files to everyone, or a group of owners of files, who share their files with each other, to publish their files with everyone. Therefore, files can be published only as OneToMany or InGroupToMany.
- Level 3: Anonymous FTP allows an owner of a file to only publish an independent copy of the file to the recipients. Hence, it allows publishing in static mode.
- Level 4: Since the sent copies to the recipients must be stored at their devices in order to be accessed, Anonymous FTP allows publishing files in distributed memory rather than shared memory.
- Level 5: Anonymous FTP provides only one type of access to the recipients which is ReadWrite which allows the recipients to read and edit the received files.
- Level 6: Anonymous FTP provides no restrictions on the type of access the recipients have.

None-anonymous FTP Unlike anonymous FTP, non-anonymous FTP does not allow anonymous access to the uploaded files on the server. Users accessing non-anonymous FTP server will be prompted for a unique username and password which will be used as a basis for making decision whether to allow or deny the user access to the files. As a result, the owner of the directory can specify different operations that can be performed by each user such as view, update, delete, and execute a file in the directory. The differences between anonymous FTP and non-anonymous FTP is only in level 1 and 2.

- Level 1: Unlike anonymous FTP, since the users in non-anonymous FTP are prompted for a unique username and password, which means that not anyone can access the files, non-anonymous FTP is suitable for sharing rather than publishing.
- Level 2: Non-anonymous FTP allows an owner of a file to share the file with a particular person or a group. Also, it allows a group of owners of files to share their files with each other as well as sharing their shared files with a particular person or group. Therefore, files in non-anonymous FTP can be shared as OneToOne, OneToGroup, InGroup, InGroupToOne, InGroupToGroup.

Cloud-storage services Cloud-storage services allow the users to create storage accounts to store their files. Users are able to perform several operations on their storage accounts such as upload, download, delete, and share files. These operations can be performed by the users in two ways. First, through a web browser from any device. Second, through a proprietary software client that installed into their devices.

Cloud-storage services offer a synchronisation service, which means operations on a storage account made through a browser will be reflected in the installed client of that account and vice versa. Also, users who own several devices (e.g., laptop, tablet, smartphone) can install a client into each device to synchronise the files stored in their storage accounts across their devices. Examples of cloud-storage services are Dropbox, Google Drive, Microsoft's Skydrive.

- Level 1: Cloud-storage services allow users to share their files either with users subscribed to the same service or with users form the outside. Sharing files with other users subscribed to the same service requires an owner of a file to select a person or a group of people from the same service to share the file with and specify the operations that they can perform on the shared file (e.g., read and write). Since the file will be released only to users from the same services and to whom the owner has selected, it is suitable for sharing. On the other hand, sharing files with other users that are not subscribed to the same service, requires the owner of the file to generate URL for that file and distribute the URL to others. The URL can be distributed to everyone (e.g. posted in a public forum) or to a person or a group (e.g. via email). Therefore, Cloud-storage services are suitable for publishing and sharing.
- Level 2: Cloud storage services allow an owner of a file or group of owners of files who are sharing their files with each other to publish their file to every one. Also, it allows an owner of a file to share his file with a particular person or group and a group of owners to share their files with each other. Also, it allows a group of people who do not share files with each other to share their files with a particular person or group. Therefore, files in cloud storage providers can be published or shared as OneToMany, InGroupToMany, OneToOne, OneToGroup, InGorup, InGorupToOne, InGroupToGroup, GroupToOne, and GroupToGorup.
- Level 3: Cloud storage services allow an owner of a file to publish or share an independent or linked copies of the file to the recipients. Hence, it allows publishing and sharing and static and dynamic mode.
- Level 4: The published or shared copies of the files must be stored at the recipients devices to be accessed. Therefore, Cloud-storage services allow publishing and sharing files in distributed memory.
- Level 5: Cloud-storage services allow the recipients to have two types of access which are ReadOnly and ReadWrite.
- Level 6: Cloud-storage services provide no restrictions on the type of access the recipients have.

6.2 Summary

We used the characterisation of the activity of file sharing in Section 5 to define a framework that can classify all possible ways of performing the activity of file sharing. The framework depicted in Figure 3, can be used to specify different policies for different sharing scenarios to meet the protection requirements of the shared files discussed in Section 4.2. For example, levels 5 and 6 of the framework are concerned with Read and Write operations and is useful to specify policies to protect the file against accidental modification by Naive or Masquerader and accidental disclosure to Masquerader. To protect the file from accidental modification by Naive, the file should be shared with ReadOnly or NoReadOrWrite type of access, so that Write operation cannot be performed on the file. To protect the file from accidental disclosure to and accidental modification by a Masquerader, ReadOnly, WriteOnly, and ReadWrite types of access should be more restricted. For example, they can be restricted to be exercised only on a specific time (e.g. working hours) or location (e.g. office building), so that whether the device of a legitimate user is stolen from home or found unattended outside working hours, Read and Write operations cannot be performed.

One the other hand, levels 1 to 4 of the framework are concerned with Send operation and is useful to specify policies to protect the file against accidental redistribution by Naive or Masquerader. For example, Send operation on a class of sharing such as share-OneToOne-Static-DM, can only be performed successfully if the file is sent to only one user who is either the owner of the file or an authorised recipient, and the copy to be sent is not linked to the original file, and must be moved to the recipient device. Therefore, any attempt to send the file to a group of people or to one user who is neither the owner nor the authorised recipient will fail.

Since the framework can classify all possible ways of sharing files, having a mechanism that enforces all classes of the framework will ensure that files will not only be protected against insider misuses but also files will be shared as their owners desired.

7 CONCLUSION and FUTURE WORK

This paper has studied one category of the insider threat problem that is concerned with file sharing. In particular, protecting the shared files against insider misuses. In this paper we investigated three fundamental questions to the design of a protection mechanism against insiders misuses. Since the insider problem is not well-defined in the literature and the insider is not clearly identified, we have proposed a classification to the insider threat problem and defined the insider and the insider threat problem precisely. Having defined the insider problem and identified the insider precisely, is the first step towards protecting the shared files against insiders. More importantly is identifying misuses that insiders might perform on the shared files. We have looked at the different insiders misuses on the shared files and characterised the protection requirements of the shared files against them. However, in order to provide a useful protection mechanism that will not interfere with the activity of file sharing, we need to consider how the activity of file sharing can be performed. Otherwise, such protection mechanism might be abandoned because it is not suitable for various task of sharing. Therefore, we have characterised the activity file sharing, motivated from the development of extensive use-case scenarios. From the characterisation we have defined a framework that can classify all possible ways of how the activity of file sharing can be performed. Such framework can be used to specify different policies for different sharing scenario to meet the protection requirements of the shared files.

We are currently working to develop a mechanism to enforce the different protections types against accidental misuses on the shared files (See Figure 2). Since software is the major cause of many breaches in security, a promising approach to create a secure software is to write it in a typesafe programming language. Therefore, we take a type-based approach to enforce security policies which is a language-based technique to provide security in programs. Generally, system security requirements can be divided into two concerns which are access control and information flow control. The former places restrictions on the release of the resources, while the latter on its propagation. Access control requirement can be specified by our characterisation of how files can be accessed while information flow control can be specified by our characterisation of how files can propagated. Next, we formalise our approach using a type system to formally analyse access control and information flow whereby our characterisation of file sharing are represented as security type annotations and access control and information flow polices are enforced through type checking to prevent accidental misuses.

REFERENCES

[1] S. Christensen. Introduction to file sharing services: An it-forensic examination of p2p clients, Accessed on [30/11/2014]. URL http://www.yumpu.com/en/document/view/15379746/introduction-to-file-sharing-services-cacpdf.

[2] IBM. The floppy disk, Accessed on [30/11/2014]. URL http://www-03.ibm.com/ibm/history/ibm100/us/en/icons/floppy/transform/.

[3] M. S. Smith. The history of file sharing: Where did it begin?, Accessed on [30/11/2014]. URL http://www.brighthub.com/computing/smb-security/articles/67395.aspx.

[4] C. Nistor. File sharing - history, Accessed on [30/11/2014]. URL http://www.pctips3000.com/file-sharing-history/.

[5] Wikipedia. Timeline of file sharing, Accessed on [[30/11/2014]. URL http://en.wikipedia.org/wiki/Timeline_of_file_sharing.

[6] Software Engineering Institute. 2011 CyberSecurity Watch Survey. Software Engineering Institute, Carnegie Mellon University, 2011.

[7] Matt Bishop and Carrie Gates. Defining the insider threat. In *In Proceedings of the 2008 Cyber Security and Information Infrastructure Research Workshop*, 2008.

[8] J. Hunker. Taking Stock and Looking Forward - An Outsider's Perspective on the Insider Threat. In S. J. Stolfo, S. M. Bellovin, A. Keromytis, S. Hershkop, S. W. Smith, and S. Sinclair, editors, *Insider Attack and Cyber Security - Beyond the Hacker*, Advances in Information Security. Springer, 2008.

[9] T. Whalen, E. Toms, and J. Blustein. File sharing and group information management. Workshop on Personal Information Management (PIM 2008), 2008.

[10] R. Anderson and R. Brackney. Understanding the insider threat. In *Proceedings of a March 2004 Workshop. Prepared for the Advanced Research and Development Activity (ARDA). http://www.rand.org/publications/CF/CF196*, 2004.

[11] M. Bishop. Position: "insider" is relative. In *Proceedings of the 2005 workshop on New security paradigms*, NSPW '05, pages 77–78, 2005.

[12] R. Chinchani, A. Iyer, H. Q. Ngo, and S. Upadhyaya. Towards a theory of insider threat assessment. In *Proceedings of the 2005 International Conference on Dependable Systems and Networks*, DSN '05, pages 108–117. IEEE Computer Society, Washington, DC, USA, 2005.

[13] K. Michelle and E. Kowalski. Insider Threat Study: Computer System Sabotage in Critical Infrastructure Sectors. May.

[14] J. Predd, S. L. Pfleeger, J. Hunker, and C. Bulford. Insiders behaving badly. *IEEE Security & Privacy*, 6 (4):66–70, 2008.

[15] C. W. Probst, J. Hunker, M. Bishop, and D. Gollmann. 08302 summary – countering insider threats. In Matt Bishop, Dieter Gollmann, Jeffrey Hunke, and Christian W. Probst, editors, *Countering Insider Threats*, number 08302 in Dagstuhl Seminar Proceedings. Dagstuhl, Germany, 2008. ISSN 1862-4405.

[16] J. Hunker and C. W. Probst. Insiders and insider threats: An overview of definitions and mitigation techniques. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, 2(1):4–27, 2011.

[17] CERT. The CERT Insider Threat Center @ONLINE, April 2013. URL http://www.cert.org.

[18] S. M. Bellovin. The Insider Attack Problem Nature and Scope. In S. J. Stolfo, S. M. Bellovin, A. Keromytis, S. Hershkop, S. W. Smith, and S. Sinclair, editors, *Insider Attack and Cyber Security - Beyond the Hacker*, Advances in Information Security. Springer, 2008.

[19] G. Silowash, D. Cappelli, A. Moore, R. Trzeciak, T. J. Shimeall, and L. Flynn. Common Sense Guide to Mitigating Insider Threats 4th Edition, December 2012.

[20] Stephen Voida, W. Keith Edwards, Mark W. Newman, Rebecca E. Grinter, and Nicolas Ducheneaut. Share and share alike: exploring the user interface affordances of file sharing. In Rebecca E. Grinter, Tom Rodden, Paul M. Aoki, Edward Cutrell, Robin Jeffries, and Gary M. Olson, editors, *CHI*, pages 221–230. ACM, 2006. ISBN 1-59593-372-7.

[21] J. S. Olson, J. Grudin, and E. Horvitz. A study of preferences for sharing and privacy. In *Proceedings of CHI 05*, pages 1985–1988. ACM Press, 2005.

[22] J. S. Olson, J. Grudin, and E. Horvitz. Toward understanding preferences for sharing and privacy. MSR Technical Report 2004–138, 2004.

[23] T. Whalen, D. Smetters, and E. F. Churchill. User experiences with sharing and access control. In *In CHI 06: CHI 06 extended abstracts on Human factors in computing systems*, pages 1517–1522. ACM Press, 2006.

[24] B. Dalal, L. Nelson, D. Smetters, N. Good, and A Elliot. Ad-hoc guesting: when exceptions are the rule. In *Proceedings of the 1st Conference on Usability, Psychology, and Security*, UPSEC'08, pages 9:1–9:5, 2008.

[25] D. K. Smetters and N. Good. How users use access control. SOUPS '09. ACM.

[26] M. L. Mazurek, J. P. Arsenault, J. Bresee, N. Gupta, I. Ion, C. Johns, D. Lee, Y. Liang, J. Olsen, B. Salmon, R. Shay, K. Vaniea, L. Bauer, L. F. Cranor, G. R.

Ganger, and M. K. Reiter. Access control for home data sharing: Attitudes, needs and practices. In *CHI 2010: Conference on Human Factors in Computing Systems*, CHI '10, pages 645–654. ACM, New York, NY, USA, 2010.

[27] M. Hart, R. Johnson, and A. Stent. More content-less control: Access control in the web 2.0. *Control*, pages 1–3, 2006.

[28] T. Whalen, E. G. Toms, and J. Blustein. Information displays for managing shared files. In *Proceedings of the 2nd ACM Symposium on Computer Human Interaction for Management of Information Technology*, CHiMiT '08, pages 5:1–5:10. ACM, New York, NY, USA, 2008.

[29] K. Scarfone and P. Mell. The common configuration scoring system (ccss): Metrics for software security configuration vulnerabilities. Technical Report 7502, National Institute of Standards and Technology, December 2010.

[30] S. Haber, B. Horne, J. Pato, T. Sander, and R. E. Tarjan. If piracy is the problem, is drm the answer? In E. Becker, W. Buhse, D. Gnnewig, and N. Rump, editors, *Digital Rights Management*, volume 2770 of *LNCS*, pages 224–233. Springer, 2003. ISBN 3-540-40465-1.

[31] Infosecurity Europe and PwC. 2013 information security breaches survey. Technical report, April 2013.

[32] AlgoSec. The state of network security 2013: Attitudes and opinions, 2013.