

File, Directory, & Shared Resource Security

Module 11



Systems Security Management

Eller / MIS 
Copyright © 2015, Arizona Board of Regents

Module Objectives



- File Systems
 - Windows, Linux, & Mac OS X
- Ownership
- Access Control Lists
- Setting Permissions
- Sharing Folders
- Sharing Printers
- Groups
- Distributed File Systems
- Multilevel Security
- Zone of Control
- Housekeeping Procedures
- Next Module...

Systems Security Management

Eller/ MIS 
Copyright © 2015, Arizona Board of Regents

At the end of the module, you should be able to:

- Describe and understand the differences in the file systems in-use by various operating systems.
- Describe what file ownership means and how to take (or pass) ownership of a file.
- Describe what Access Control Lists (ACLs) are and how they are used to secure resources.
- How to set permissions on files and folders in each of the major operating systems.
- How to share folders within each operating system.
- How to use groups to simplify system administration.
- What distributed file systems are and how they are used in enterprise.
- What multilevel security is and how it is viewed by government and business.
- What zone of control means and how this concept applies to shared resource security.
- What housekeeping measures are necessary for system administrators to complete regularly.

File Systems



- What is a File System?
- Types of File Systems
 - File Allocation Table (FAT16/FAT32)
 - New Technology File System (NTFS)
 - Resilient File System (ReFS)
 - Linux File System (ext3/ext4)
 - Mac OS X File System (HFS+)

Systems Security Management

Eller/ MIS 
Copyright © 2015, Arizona Board of Regents

What is a File System?

Computer operating systems use file systems as a means of storing and organizing data. Each operating system has a different way to accomplish this. The major file systems used over the past 20 years include the following:

Types of File Systems

- File Allocation Table (FAT16/FAT32)
- New Technology File System (NTFS)
- Resilient File System (ReFS)
- Linux File System (ext3/ext4)
- Mac OS X File System (HFS+)

File Allocation Tables



- **FAT16**
 - 16-bit File System
 - MSDOS/PCDOS
 - 2 Gigabyte Limit
 - Short File Names (8.3)
- **FAT32**
 - 32-bit File System
 - Windows 95/98/ME
 - Also Supported By
 - All Versions of Windows
 - Linux
 - MacOS X
 - Long File Names (255 Characters)
 - 8 Terabyte Limit



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

What is FAT16?

FAT16 is a 16-bit version of the File Allocation Table file system. The FAT16 file system was used by multiple operating systems of the time, including the MS-DOS and PC-DOS operating systems in addition to the graphical user interface known as Windows 3.x. In addition, the IBM OS2 operating system made use of the FAT16 file system. The FAT16 system had multiple limitations that were (mostly) corrected in the next iteration of the file system (FAT32). These limitations included a maximum hard drive partition size of 2 gigabytes and could only use short file names (8 character name and a 3 character file extension - commonly referred to as 8.3).

What is FAT32?

The FAT32 file system succeeded the FAT16 file system and was designed to provide a number of significant improvements over FAT16. The FAT32 file system was introduced with Microsoft's Windows 95 operating system; however, it was also used by Windows 98, Windows ME, and was available as an option in Windows 2000 and Windows XP. At this time, all current operating systems support both the FAT16 and FAT32 file systems, although the FAT16 file system is quickly disappearing. The reason for continuing support for these file systems is due to the use of these file systems on portable memory devices (such as USB memory sticks, MP3 players, among others) in order to maintain compatibility between different OS options

Perhaps the most important reasons for upgrading to the FAT32 file system (from FAT16) included the ability to use long file names and a dramatic increase in the maximum hard drive partition size. The new ability for long file names now allowed for users to save files with easy to understand (and organize) file names utilizing a maximum of 255 characters. For partition sizes, FAT32 increased the previous 2 gigabyte maximum to an 8 terabyte maximum. This was absolutely unheard of at the time FAT32 was introduced, as 2 terabyte drives did not make their appearance until 2009 and as of 2014 are only in the 8 terabyte range with 10 terabyte drives filled with helium being sampled.

File Allocation Tables



- Attributes
 - Applied to Files & Folders
 - Types
 - Hidden
 - Archive
 - Read-Only
- Security
 - None



Systems Security Management


Eller/ MIS
Copyright © 2015, Arizona Board of Regents

What Attributes are Available in FAT?


Both versions of the FAT file system (FAT16 & FAT32) had three primary attributes one could apply to both files and folders. These attributes include: Hidden, Archive, and Read-Only. The Hidden attribute would actively “hide” files and folders from view when one would ask the file system for a list of the files and folders contained in a specific folder. This attribute would hide file(s) and folder(s) regardless of the method for listing (e.g. either through the graphical user interface or the command line). The Archive attribute was specifically designed for backup purposes. When a file was modified, the Archive attribute would be applied to the file. This would tell the backup software this file has changed and needed to be backed up. Once backed up, the backup software would then turn off the archive attribute on the file so it would not be backed up again unless the file changed again. Finally, the Read-Only attribute would prevent everyone from writing changes to the file unless this attribute was turned off.

Are there any Security Options in FAT?


There are no true security options available in either of the FAT file systems. While some might consider the use of the hidden file attribute a security feature (security by obscurity), this is not a means of keeping those who should not access the file(s) from accessing them. The Read-Only attribute could also be considered “security;” however, the primary failing of all of these attributes is that they could be turned off quickly and easily through either the command line or GUI.



NT File System



- NTFS
 - Designed to Replace FAT16/32
 - Supported By
 - Windows NT/2000+
 - Linux *
 - Mac OS X *
 - Multiple Versions
 - 256 Terabyte Limit




Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents


What is NTFS?

The New Technology File System, or NTFS, was created by Microsoft and introduced in the Windows NT operating system. This file system was designed to replace the FAT file systems, providing security options for the first time. The NTFS file system is still in use in the current iterations of the Windows operating systems (Windows 7 and Windows Server 2008 R2) and it will likely continue to be available as an option in the next versions of Windows as well. The NTFS file system is also supported by both the Linux and Mac OS X operating systems; however, neither OS is capable of writing to drives formatted with NTFS unless they are shared over a network or using 3rd party software.

Over the years since its introduction, NTFS has evolved, making more efficient use of space and allowing for the formatting of larger hard drives. Since Windows NT, each version of Windows has introduced a slightly improved version of NTFS. The current version of “NTFS provides performance, reliability, and advanced features not found in FAT. NTFS also provides support for volumes up to 256 terabytes in size, support for disk quotas, and support for mounted drives” (Microsoft Corporation, 2003).



NT File System



- Attributes
 - Same as FAT16/FAT32
 - Compression
 - Encryption (EFS)
- Security
 - File/Directory Permissions
 - Standard vs. Special
 - Ownership
- Copying vs. Moving

Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

What Attributes are used with NTFS?

NTFS offers the same three attributes available with both FAT file systems; however, it adds two significant attributes that dramatically increase the usefulness of the file system: compression and encryption. Compression is a useful feature to have, especially when disk space is a concern; however, it is important to remember that in order to execute compressed files, there must be enough disk space available to decompress and recompress the file. With encryption, the NTFS file system actually recodes the files using the Encrypting File System.

Encrypting File System (EFS)

EFS encryption is based on Public Key Infrastructure (PKI), which itself is based on the concept of key pairs. Each user is given a public key and a private key. A user's public key is available to anyone who asks for it; the private key is accessible only to the user who owns it.

When a user tells Windows he wants to encrypt a file, EFS generates a file encryption key. Windows uses this randomly generated key in conjunction with the Data Encryption Standard-X (DESX) algorithm to encrypt the file. (3DES encryption became an option in Windows XP SP1). The encrypted file is written to disk.

DESX is a strengthened version of the original DES algorithm and is a standard algorithm used by anyone who attempts to decrypt the file. The trick is that DESX doesn't work without a file encryption key. Since the file encryption key actually resides on the same system as the encrypted files, this key is vulnerable to compromise. To help protect the file encryption key, EFS encrypts it using the user's public key.

This means that the file is encrypted, but the key to the file is also encrypted using a different key. So two keys are needed to open the file. The first is the private key of the user who encrypted the file. Only the user's private key can decrypt something encrypted using the user's public key. Once the file encryption key has been decrypted, it can be used to decrypt the encrypted file.

When a user with permission to access the file opens the file, NTFS performs a decryption transparently in the background. But the file itself is not actually decrypted; that would leave it vulnerable to others while it is being accessed. Instead, the EFS creates a decrypted copy of the file and hands it to the application that opened the file. Any changes made to the decrypted copy of the file are automatically passed on to the encrypted copy. When the user closes the file, all that remains is the encrypted copy (Posey, 2006). In

other words, if you used EFS to encrypt a Word document, when you open the file to modify the document, EFS will make a copy of the file automatically and decrypt the copy. You can then modify the document and when you save it, the changes are actually saved back to the original encrypted version. When you close Word, the decrypted document is merged back into the encrypted version so all of your changes remain in the original encrypted file.

What Security Options are Available?

NTFS is the first Windows file system offering security features. Security is handled through Access Control Lists (ACLs), providing both standard and customizable security settings for each file and folder. The standard security options include: full control, modify, read & execute, read, and write.


- **Full Control** – gives the specified user or group complete control over a file or folder, including the ability to take ownership.
- **Modify** – gives the specified user or group all rights except the ability to change permissions and take ownership.
- **Read & Execute** – gives the specified user or group the ability to read files, list contents, execute files, and read permission settings.
- **Read** – gives the specified user or group the same rights as Read & Execute, without the ability to execute files.
- **Write** – gives the specified user or group the ability to create files and folders as well as write and apply attributes to files.

With Special Permissions, the system administrator can grant very specific rights to users or groups that may include some components of multiple standard options, but not all. For example, if you wanted to allow someone to have Modify rights without the ability to execute files or write attributes, this would be considered special permissions.

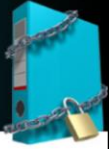
What Happens when you Copy or Move Files or Folders?

Copying and Moving files and folders when NTFS attributes or security have been applied to them provides an additional challenge. When you copy files from a secured folder to an unsecured folder on the same hard drive, the files will be unsecured in the second folder (the files inherit the security of the destination folder). When you move files from a secured folder to an unsecured folder on the same hard drive, the security and attributes will move with the files.


When copying or moving to a second hard drive or network share, in either case, the files or folders will inherit the security level of the destination. This is especially true if the destination is a drive formatted with a file system other than NTFS (e.g. such as a USB memory stick formatted with FAT32). There is an exception to this rule; however, as long as the destination uses NTFS, any files using EFS will remain encrypted. Something else to note, if an unencrypted file resides in an unencrypted folder and is moved into an encrypted folder (regardless of destination location – assuming it is NTFS) the resulting file will remain unencrypted; however, if the unencrypted source is copied into the encrypted destination, the resulting file WILL be encrypted (Eckel, 2002).



Resilient File System



- Backwards Compatibility with NTFS
- Introduces Data Integrity Features
- Resiliency to Corruption
 - Data Striping/Scrubbing
 - Integrity/Availability
- Large Volume, File, and Directory Sizes
 - 1 Yottabyte (10^{24}), 16 Exabyte, 18.4×10^{18}
- Storage Pooling
 - Share Storage Across Systems



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

Resilient File System

Resilient File System or ReFS is a new local file system to replace NTFS that was introduced by Microsoft with Windows Server 2012. It is backwards compatible with NTFS and introduces data integrity and availability features for the file system. The way ReFS stores data protects it from many of the common causes of corruption and data loss, ensuring that data is not lost and, in event of a system error, is recovered and restored quickly, with no volume down time. ReFS is designed to stay online with built in data parity and mirroring, scrubbing, and, when configured with Storage Spaces, will automatically correct the corruption.

With increased storage capacities, from 16 Exabytes to 1 Yottabyte, filename sizes up to 32,768 characters, and 18 quintillion files allowed per volume, ReFS is designed to handle the data set sizes of today and tomorrow. This allows for a high level of scalability, and storage pooling is now built-in, with Storage Spaces also allowing storage to be shared across systems.



Linux File System



- **ext3 – Third Extended File System**
 - Used by Many Distributions
- **Max Drive Size**
 - Depends on Block Size
 - 1KB Blocks = 16GB Files = 2TB Drive
 - 2KB Blocks = 256GB Files = 4TB Drive
 - 4KB Blocks = 2TB Files = 8TB Drive
 - 8KB Blocks = 2TB Files = 16TB Drive *
 - Disadvantages



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

What is the Third Extended File System (ext3)?

“The ext3 or third extended file system is a journaling file system that is commonly used by the Linux kernel. It is the default file system for many popular Linux distributions. Ext3 has a significant advantage [over other Linux file systems] in that it allows in-place upgrades from the ext2 file system without having to back up and restore data. Ext3 also uses less CPU power and is considered safer than other Linux file systems due to its relative simplicity and wider testing base” (ext3, 2010).

It is important to note that “a journaling file system is a file system that keeps track of the changes it intends to make in a journal before committing them to the main file system. In the event of a system crash or power failure, such file systems are quicker to bring back online and less likely to become corrupted” (Journaling File System, 2010).

What is the Maximum File & Drive Size Available in ext3?

Linux file systems store information in blocks, and depending on the size of a block will determine the maximum drive and file sizes.

- 1KB Blocks = 16GB Files = 2TB Drive
- 2KB Blocks = 256GB Files = 4TB Drive
- 4KB Blocks = 2TB Files = 8TB Drive
- 8KB Blocks = 2TB Files = 16TB Drive *

It is important to note that the option for 8KB blocks is not available in all Linux distributions.

What are the Disadvantages of ext3?

Despite improvements over the previous version (ext2), there are a number of disadvantages to the file system. There are limits to the file system's functionality due to the necessity of maintaining backwards compatibility with ext2. The file system does not support defragmentation, so portions of files can be split across a drive, making file access a little slower. There is also no support of file recovery built into ext3, and while commercial tools exist, none guarantee file recovery, and if a drive is formatted, there is no hope for any recovery at all. Finally, in ext3, there is no default support for compression techniques (ext3, 2010).



Linux File Systems



- ext4 – Fourth Extended File System
 - Merged into Linux Source v. 2.6.28
 - October 11, 2008
- Backwards Compatible
- Max Drive Size
 - 16 Terabyte File
 - 1 Exabyte Drive
- Max Subdirectory Size Increased
 - 64,000 from 32,000 (ext3)



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

What is the Forth Extended File System (ext4)?

Ext4 was developed out of a need to improve upon the ext3 file system. Work was originally begun as an extension of ext3, meaning the final version was meant to be an updated version of ext3 instead of a completely new file system. However, much of the Linux community was opposed to upgrading the existing version of ext3 for compatibility reasons and because some feared the updated version might cause significant problems for existing system configurations. Instead it was proposed to rename the file system upgrade to ext4 and allow it to be an option for install instead of a mandatory upgrade to ext3. This plan was approved and the final version of ext4 was merged into the Linux source code (specifically version 2.6.28 and later) for most distributions on October 11, 2008.

What is the Maximum File & Drive Size Available in ext4?

Since ext4 is backwards compatible with ext3 and ext2, the same maximums exist as with those file systems; however, ext4 added the ability for maximum file sizes of up to 16 terabytes with support for hard drives and volume sizes of up to 1 exabyte. These maximums are mostly to support the use of very large databases. In addition, ext4 improved upon ext3 by increasing the maximum number of files and folders contained within a single folder to 64,000 from 32,000.



Mac OS X File System



- Mac OS Extended Format
 - HFS+ - Hierarchical File System Plus
 - 32-bit File System
 - Supports 255 Character Filenames
- Max Drive Size
 - 2 Terabytes (10.0 – 10.1.5)
 - 8 Terabytes (10.2 – 10.2.8)
 - 16 Terabytes (10.3 – 10.3.9)
 - 8 Exabytes (10.4+)
- Supports FAT16/FAT32 & NTFS (Read Only)



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

What File System is Used by the Mac OS X?

HFS+ is the preferred file system on Mac OS X. It supports journaling, quotas, byte-range locking, Finder information in metadata, multiple encodings, hard and symbolic links, aliases, support for hiding file extensions on a per-file basis, etc. HFS+ is a 32-bit file system which supports 255 character filenames for easier organization and search.

What is the Maximum Drive Size Available?

| | |
|--|-------------------|
| Maximum number of volumes (all Mac OS X versions) | no limit |
| Maximum number of files (or files and folders) in a folder (all Mac OS X versions) | up to 2.1 billion |
| Maximum volume size and file size (Mac OS X 10.0 - 10.1.5) | 2 TB |
| Maximum volume size and file size (Mac OS X 10.2 - 10.2.8) | 8 TB |
| Maximum volume size and file size (Mac OS X 10.3 - 10.3.9) | 16 TB |
| Maximum volume size and file size (Mac OS X 10.4 or later) | close to 8 EB |

Information from Apple.com (Mac OS X, 2008).

Other File System Support

The Mac OS X operating system also supports the FAT16/FAT32 file system (to use memory sticks among other options) and the NTFS file system. IT is important to note that the NTFS file system is supported by Mac OS X as read only, meaning by default you cannot write any information to an NTFS formatted hard drive from the Mac, with two exceptions:

Exception #1: The NTFS formatted drive is shared on a network, which is how the Mac accesses the drive.

Exception #2: There are third party applications available for the Mac, which allow the OS to write files to a local NTFS formatted hard drive.

Ownership



- What is Ownership?
 - All Files & Folders Have Owners
 - Creator/Owner Permission
 - Full Control
- Taking Ownership
 - Restricted Action
 - Granting Ownership
 - Why?



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

What is Ownership?

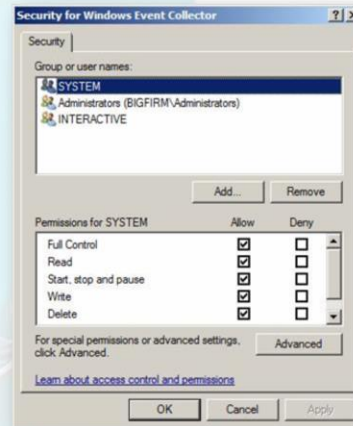
Whenever you create a file, whether the file is a Microsoft Word document, a downloaded file from the Internet, or an E-Mail attachment, you will automatically become the owner of the file. What this means is as the owner, you have the right to do anything you want (or need) to do with the file. You can edit it, save it, copy it, move it, and delete it, among other tasks. Every file and folder on your computer has an associated owner. In general with Windows, the default file owner is the Administrator account. Once Windows has been installed and setup for an individual to use, every file created from that point will have the specific user as the file or folder owner. This is accomplished through a special permission group in Windows called Creator/Owner. By default this special group is granted Full Control rights to files and folders.

Can you take Ownership?

As a system administrator it is possible to take ownership of a file or folder. This action is a restricted action within the Windows OS, meaning only user accounts with specific abilities are able to take ownership from another user account. For example, if the user account "jdoe" owns a specific file, the Administrator account can take ownership of the file away from "jdoe." What this means now is "jdoe" now cannot take ownership back from Administrator, nor can "jdoe" change the file permissions using an access control list (which we will discuss in a moment). In addition, a file owner can grant ownership to files and folders. This task is very similar to taking ownership, but instead of taking ownership from another user account, the owner can change ownership to another user. This task is usually used by system administrators who need to maintain proper ownership records, usually in order to maintain disk quotas. The reason for this is all about maintaining the security of files and folders on computers and the network.

Access Control Lists

- What is an Access Control List (ACL)?
- What Uses ACLs?
 - Files
 - Folders
 - Objects
 - Shared Folders
 - Printers



Systems Security Management


Eller / MIS
Copyright © 2015, Arizona Board of Regents

What is an Access Control List?


An Access Control List (ACL) allows a system administrator to create a list of users, groups, or computers which can access a specific resource.

What Uses ACLs?


Access control lists are used by a number of different operating systems for many different purposes. This includes securing files, folders, objects (such as user accounts in Active Directory), and shared resources (such as network folders and printers). Each of these examples can use an ACL to limit access to specific resources as deemed necessary by the system administrator or management.



Setting Permissions



- Windows
 - Requires NTFS
 - Right Click File/Folder
 - Properties
 - Security Tab
 - Add/Remove Users/Groups
 - Set Permission Level
 - Click OK
 - Take/Grant Ownership in Advanced



Systems Security Management

Eller / MIS
Copyright © 2015, Arizona Board of Regents

Setting Permissions in Windows

Setting file permissions requires the use of the NTFS file system. As such you cannot set permissions as shown on a hard drive formatted with the FAT file systems (USB memory sticks for example) without reformatting the drive as NTFS.

How to Set Permissions on a File or Folder in Windows

- Right Click File/Folder
- Properties
- Security Tab
- Add/Remove Users/Groups
- Set Permission Level (Full Control, Modify, Read & Execute, List Folder Contents, Read, Write, or Deny)
- Click OK

In addition, in the Windows environment, by clicking on the Advanced button shown here you will find options for taking ownership and assigning more specific permissions than the default options.



Setting Permissions



- Linux
 - Open Terminal (or Connect via SSH)
 - Locate File or Folder
 - Examine Existing Permissions
 - Permissions Formatted “rwxrwxrwx”

| | |
|---|---------|
| r | Read |
| w | Write |
| x | Execute |
 - 1st “rwx” Refers to the File Owner
 - 2nd “rwx” Refers to the Group
 - 3rd “rwx” Refers to Others
 - E.g. Everyone Else
 - Ex: “chmod 755 filename” = rwxr-xr-x



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

How to Set Permissions From the Command Line in Linux

In a Linux environment, you can change file permissions in two ways: through a command prompt or through the graphical user interface (GUI). The GUI (pronounced “goeey”) is done in a manner similar to Windows. Here we will discuss setting permissions through the command prompt.

To begin you will need to open a terminal window (if using the GUI) or connect to the Linux system remotely via SSH (Secure Shell) software. You will need to locate the file or folder you need to set permissions on. This is done using the commands “ls” (for listing the contents of a directory) and “cd” (for changing to a new directory). Once you have located the file or folder, you will need to do another directory listing in order to see the current permission level. To see the permissions you would type “ls -al” and hit enter. You should see something similar to the following:

```
drwxr-xr-x 2 root root 4096 Jan 1 2010 usr
```

Looking at the above, there are a few things you can tell that have nothing to do with permission level. The letter “d” at the beginning denotes this is a directory and “usr” is the name of the directory. You will also see the date the folder was created (Jan 1, 2010). The number 4096 refers to the size of the folder in bytes (note not the size of the folder’s contents, just the folder size itself). The two listings for “root” show the user account and group which owns the folder.

Finally we get to the rwxr-xr-x section. This is where the folder’s permissions are defined. In general, possible permissions are formatted in the command line as rwxrwxrwx. The “r” refers to “read” permissions, the “w” refers to “write” permissions, and the “x” refers to “execute” permissions. When reading the line “rwxrwxrwx,” without any Linux experience the first question becomes why does it repeat three times. The answer to that is as follows:


The first “rwx” refers specifically to the folder’s owner.

The second “rwx” refers specifically to groups.


The third “rwx” refers to all others who might access the folder.

In order to change the folder’s permissions you will use the “chmod” (Change Mode) command. This command uses numbers to designate the permission level for each “rwx” section. Depending on the number combination you use will determine the permission setting. For example, for a permission setting of “rwxrwxrwx” the number you would enter is “777.” In order to get “rwxr-xr-x” as shown in the above example, you would use the number “755.” What you are stating with “755” is the folder owner has full permissions, while groups and others cannot write to the folder (but can read and execute from the folder).

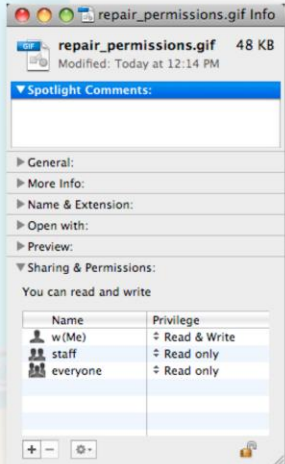
More details about the “chmod” command can be easily found on the Internet.



Setting Permissions




- Mac OS X
 - Command Line
 - Similar to Linux
 - Get Info (File/Folder)
 - Permission Settings
 - Read & Write
 - Read Only
 - Write Only (Drop Box)
 - No Access
 - Set for Owner, Group, Others



| Name | Privilege |
|----------|--------------|
| w(Me) | Read & Write |
| staff | Read only |
| everyone | Read only |

Systems Security Management


Eller / MIS 
 Copyright © 2015, Arizona Board of Regents

How to Set Permissions from the Command Line in the Mac OS X

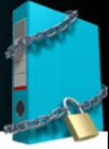
Since the Mac OS X is based on a Linux kernel, setting permissions from the command line is done in the exact same manner as in Linux.

How to Set Permissions from the GUI in the Mac OS X


Setting permissions from the Mac GUI is done by selecting a folder and choosing Get Info from the File menu. On the Get Info window you will see a section called Sharing & Permissions where you can specify users or groups and set permissions from a pre-determined list. This list includes Read & Write, Read Only, Write Only (Dropbox), and No Access. These permissions can be set on the owner, group, or others, exactly the same options as in Linux.




Sharing Folders



- Windows
 - Right Click on a Folder
 - Choose Properties
 - Sharing Tab
 - Give Share a Name
 - Set Access Permissions (ACL)
 - Click OK to Start Share
 - Minor Differences Depending on OS Version
 - Non-Domain Members
 - Additional Steps Necessary



Systems Security Management


Eller/ MIS 
 Copyright © 2015, Arizona Board of Regents

How to Share a Folder in Windows



In order to share a folder on a Windows system, begin by locating the folder you want to share. Now do the following:

- Right Click on the Folder
- Choose Properties
- Click on the Sharing Tab
- Give the Share a Name
- Set Access Permissions (ACL) by clicking Permissions
- After setting the appropriate permissions click OK
- Click OK to create the shared folder

Now, there are minor differences to how this is accomplished depending on the version of the operating system you are using. For example a Windows Home version will share in a different manner than a Windows Professional or Server OS will. This is to make the process of sharing files in a home network environment easier. Also, if your computer is not a member of a Windows domain, additional steps will be necessary.



Sharing Folders



- Linux (Command Line)
 - Compile & Install Samba
 - Edit /etc/samba/smb.conf
 - [share]
comment = share
path = /path/to/your/shared/folder
public = yes
browsable = yes
writeable = yes
 - Set Permissions on Folder
 - Using “chmod” Command

Systems Security Management


Eller/ MIS
Copyright © 2015, Arizona Board of Regents

How to Share a Folder in Linux Using the Command Line


In order to share a folder on a Linux system using the command line you will need to download the Samba program. Once downloaded, the program needs to be compiled and installed. After install, navigate to the /etc/samba folder and edit the smb.conf file. Inside that file you will find a section with the title [share]. You will need to modify the file with the following information:

```
[share]
comment = share
path = /path/to/your/shared/folder
public = yes
browsable = yes
writeable = yes
```


After you save the file and exit the editor you will need to set permissions on the folder itself. You can do this using the chmod command as outlined on slide number 15.



Shared Folders



- Linux (GUI)
 - Right Click on the Folder
 - Choose Folder Sharing
 - Check the Share this Folder Box
 - Choose a Name
 - Click Create Share
 - Set Access Permissions
 - This will be accessible via Samba
 - Also accessible via Windows and Mac Clients



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

How to Share a Folder in Linux Using the GUI

Sharing a folder from a Linux GUI is a much easier process. First, locate the folder you want to share on the network and do the following:

- Right Click on the Folder
- Choose Folder Sharing
- Check the Share this Folder Box
- Choose a Name
- Click Create Share
- Set Access Permissions

Once you create the shared folder it will immediately be available to any Linux system using Samba as well as Windows and Mac OS X clients.



Sharing Folders



- Mac OS X
 - Apple Menu
 - System Preferences
 - View Menu
 - Sharing
 - Select the Folder
 - Select User/Group Permissions
 - Alternative Method
 - Select Get Info on Folder
 - Check Box - Shared Folder
 - Set Access Rights in Sharing and Permissions




Systems Security Management

Eller / MIS
Copyright © 2015, Arizona Board of Regents


How to Share a Folder in Mac OS X Using the GUI

Sharing a folder in the Mac OS X is also a simple matter. One method for sharing folders starts by clicking on the Apple menu and choosing System Preferences. On the View menu, select Sharing. Now choose the folder you want to share and select the users and groups that need access along with the appropriate permissions for each.

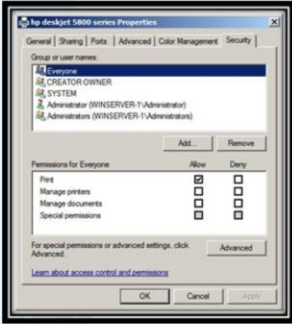
Alternatively, you can share a folder by choosing the Get Info option on a specific folder. In the Get Info window, check the box named Shared Folder and set the appropriate access rights under Sharing and Permissions.




Sharing Printers



- Windows
 - Can be Shared During Install
 - Remember to Set Security!
 - When not Shared During Install
 - Right Click on Printer
 - Choose Properties
 - Sharing Tab
 - Provide Shared Name
 - Click OK to Share the Printer
 - Set Security Permissions
 - Security Tab



Systems Security Management

Eller/ MIS 
 Copyright © 2015, Arizona Board of Regents

How to Share a Printer in Windows

Sharing a printer on a Windows system is an easy prospect. There are two ways to accomplish this. The easiest way is when you install the printer itself on the system. During the install you will be asked if you want to share the printer. You can assign the printer a name and when the install completes the printer will be available on the network. One caveat to this is you **MUST** go and reset security permission after install. This will be discussed in a moment.

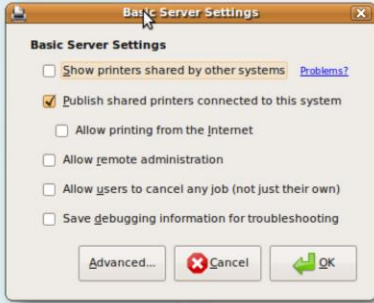
When you elect to not share the printer during the install process, you can still share the printer after the fact. This is done in a manner similar to sharing a folder. Just do the following:

- Right Click on Printer
- Choose Properties
- Sharing Tab
- Provide Shared Name
- Click OK to Share the Printer

Once you have shared the printer (regardless of how you shared it) you must change the security permissions on the printer itself. By default, Windows allows Everyone to print. In this case, Everyone MEANS Everyone. Regardless of whether or not you have a network user account and people can print to the server even if they are not physically connected to the network. Make sure you remove Everyone from the list and add only those users or groups who need to access the shared printer.

Sharing Printers

- Linux (GUI)
 - Click System Menu
 - Choose Administration
 - Choose Printing
 - Setup the Printer
 - Select
 - Publish Shared Printers Connected to this System
 - Under Server Settings



Systems Security Management


Eller / MIS
Copyright © 2015, Arizona Board of Regents

How to Share a Printer in Linux


Sharing a printer on a Linux system is also easy to do. Start with the System Menu and do the following:

- Choose Administration
- Choose Printing
- Setup the Printer
- Select
 - Publish Shared Printers Connected to this System
 - Which is Located Under Basic Server Settings

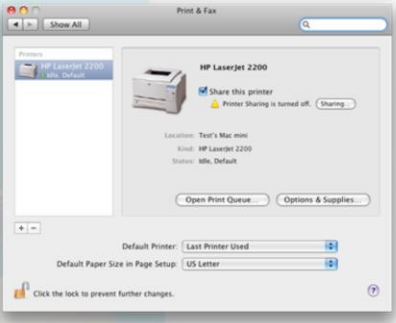
NOTE: These instructions may slightly differ depending on the distribution of Linux being used.



Sharing Printers



- Mac OS X
 - Click Apple Menu
 - Click System Preferences
 - Click Print & Fax
 - Select the Printer
 - Select “Share this Printer”



Systems Security Management

Eller / MIS
Copyright © 2015, Arizona Board of Regents

How to Share a Printer in Mac OS X

Sharing a printer on the Mac is also easy to do. Just do the following:

- Click Apple Menu
- Click System Preferences
- Click Print & Fax
- Select the Printer
- Select “Share this Printer”

Groups



- Created for User Membership
- Simplifies Administration
- Assign User Accounts to Groups
 - Assign Groups to Shared Resources
 - Prevents Need for Changing Permissions on a Shared Resource
 - Just Change Group Membership
- Different Types of Groups Available
 - Security, Local, Domain, Global, Universal...



Systems Security Management

Eller / MIS
Copyright © 2015, Arizona Board of Regents

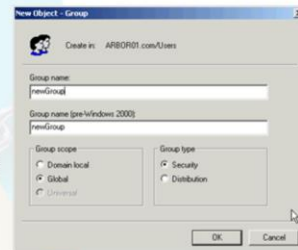
Groups

We have touched on groups briefly in Module 9; however, groups become even more important when securing files, folders, and shared resources. To recap, groups are created primarily for user membership. The use of groups helps to simplify administrative tasks on servers. A SysAdmin can assign user accounts to specific groups and then assign the appropriate groups to shared resources. By assigning groups to shared resources instead of specific user accounts, a SysAdmin will prevent the need for changing permissions on a shared resource. When the SysAdmin needs to remove access for a specific person, he or she can simply remove the specific user from the group and that user will no longer be able to access the shared resource.

On servers there are several group choices available, including security, local, domain local, global, and universal.

Groups

- Security Groups
 - General Group Used for Security Purposes
 - Windows, Linux, Mac OS X
- Local Groups
 - Manage Resources on a Local Computer/Server
 - Windows
- Domain Local Groups
 - Used when AD is Deployed
 - Used to Manage Domain Resources
 - Windows



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

Groups (continued)

Security Groups

Security groups are a general group used solely for security purposes. Users can be assigned to these groups and the groups can be assigned to files, folders, and shared resources to grant or deny access. This type of group is available in Windows, Linux, and Mac OS X systems.

Local Groups

Local groups allow a SysAdmin to manage resource access on a local computer or stand-alone server. This type of group is available only on a Windows system and is not typically used in a network environment.

Domain Local Groups

Domain local groups are available and used only when a Microsoft Active Directory Domain is deployed on a network. These groups are used to manage resources across a domain. Since Active Directory is required, this type of group is only available on Windows servers.

Groups



- Global Groups
 - Security Groups within a Single Domain
 - Can Contain Both Domain Local Groups & User Accounts in the Same Domain
 - Windows
- Universal Groups
 - In AD, Used to Group Objects in Multiple Domains, Child Domains, and Forests
 - Windows
- Group Identification Number (GID)
 - Used to Uniquely Identify a Group



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

Groups (continued)

Global Groups

Global groups are security groups that are available within a single Windows Active Directory Domain. These groups can contain both user accounts in the domain and can actually have domain local groups as members as well.

Universal Groups

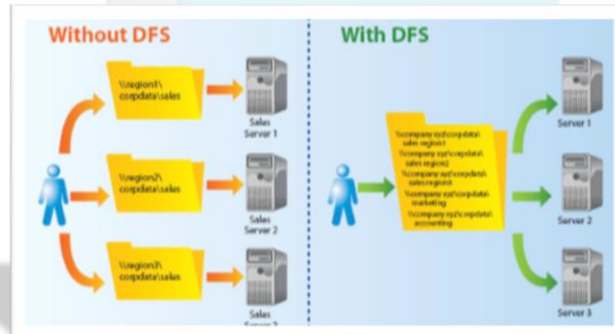
Universal groups are also available only in a Windows Active Directory domain and can be used to group objects from multiple domains and child domains within the same forest. These objects can include user accounts, domain local groups, and global groups. Universal groups can also add members from domains in other forests assuming a trust relationship exists.

Group Identification Numbers (GIDs)

Group ID numbers are used in all operating systems to uniquely identify a group. Every group will be assigned a different GID and the GID will be referenced when access permissions are set on shared resources.

Distributed File Systems

- Network Protocol
- Multiple Servers
- Appearance to Users
 - Consolidated to Single Location
- Microsoft Offers Built-in DFS
- Other Third-Party DFS Solutions



Systems Security Management

Eller/ MIS
Copyright © 2013, Arizona Board of Regents

Distributed File Systems

A distributed file system is a network protocol that allows data stored on multiple servers to appear as if the data is located in a single network location. In a normal network environment you would expect to see several servers which all contain data specific to departments, teams, or projects. A distributed file system, or DFS, would allow a SysAdmin to leave the data on each of the different servers while creating a single centralized location where users can access the data from each server they have access to. This makes the process of finding data on the network a little easier for users since they will always know the data is available from the single location.

Microsoft offers a built-in DFS solution for free with Windows Server Standard, Enterprise, and Datacenter versions. Should an organization decide not to adopt the Microsoft solution, there are a variety of third-party DFS solutions available for purchase.

Multilevel Security



- What is Multilevel Security?

- Protecting Data

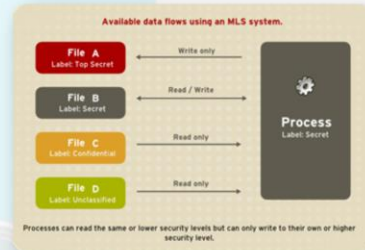
- Allows & Denies
 - Managed by the System

- Government/Military

- Secrets Remain Secret

- Corporate

- Can Tolerate Some Leakage
 - Business Software Example
 - Microsoft SharePoint



Systems Security Management

Eller / MIS
Copyright © 2015, Arizona Board of Regents

Multilevel Security

Multilevel security is a method by which a system can automatically protect the data stored there using multiple methods. Protecting data using multilevel security means the server will allow and deny depending on the user's authorization. So, for example, you may use NTFS permissions to make a specific file read-only to a specific user, then share the folder that file resides in on the network using share-level permissions. The share-level permissions might grant the user read and write access. Finally, the server may be setup to be a stand-alone server, meaning it is not a member of a Windows domain, so a centralized user account would not be allowed. This would then require the user to enter a username and password in order to access the resource. This is an example of multilevel security.

Governments and the Military use multilevel security as a means to allow secrets to remain secret. Only the users with the correct permission levels throughout the network system will be able to access certain information. Corporations on the other hand can tolerate some information leakage, meaning there is less of a need for multilevel security. Having said that, the Microsoft SharePoint server platform was designed to not only allow users to more easily collaborate, it also provides multiple means of securing data throughout the system so only those who need access to a file will be granted access.

Zone of Control



- Who Has Control?
 - Systems Administrators?
 - Users?
 - Departments?
- It Depends!
- SysAdmins
 - Set
 - Verify
 - Enforce



Systems Security Management

Eller/ MIS
Copyright © 2015, Arizona Board of Regents

Zone of Control

Now that we have learned all about file systems and how to store, retrieve, and share data on servers, the question becomes who has control of the data stored on a server? Do SysAdmins have control? How about the users? Departments or teams? The answer is it depends! In general, the SysAdmin will have physical control of the data and will be responsible for setting access permissions, verifying only those who are authorized can access the data, and enforcing IT policies to maintain file system security. Depending on the need for data sharing, will determine who has logical control of the data, so this could be specific users, departments or teams.

Housekeeping Procedures



- Why is Housekeeping Necessary?
 - Archival/Removal of Unnecessary Files
 - Recover Disk Space
 - Reduce Necessary Backups
 - Ensure Permissions Remain Correct
 - Need to Know Change?
 - Defragmentation
 - Speed Up File Access



Systems Security Management

Eller / MIS
Copyright © 2015, Arizona Board of Regents

Housekeeping Procedures

Why is Housekeeping Necessary?

Housekeeping is an important part of System Administration, and is absolutely necessary when it comes to file servers. SysAdmins need to regularly archive or remove unnecessary files from network systems. The reason for this is to allow the system to reclaim disk space for use by others as well as to reduce the amount of data that needs to be regularly backed up.

SysAdmins also need to perform regular housekeeping maintenance in order to ensure access permissions remain correct. In some cases it may become necessary to remove someone's access even though they are still employed with the organization. Finally, SysAdmins should schedule regular defragmentation jobs on the file servers. Fragmented files are caused by applications needing to save a data file and there is not enough contiguous free space available in one chunk so it splits the file into two or more pieces in order to save the file. Defragmentation recombines each piece into a single file and rearranges the data on the hard drives so a few files as possible remain fragmented. This helps to speed up access to files when they are needed.

Next Module...



- Physical Security
- Server Security
- Workstation Security
- Facilities Considerations (HVAC, Power, Fire)
- Applicable Policies
- Media Considerations
- Media Reuse/Destroy
- Fax Security Considerations
- Zone of Control
- Designing a Network Topology for Security
- Network Topologies (Bus, Ring, Star, Bus-Star)
- Communications Media (Coax, Twisted Pair, Fiber, Wireless)
- Accepted Guidelines for Cable Installation
- Deploying Structured Wiring Design
- Implementing Structured Network Design
- Vertical Wiring Principles
- Centralized Management
- Virtual LANs
- Network Redundancy
- Aggregation
- OSI Model Implications

Systems Security Management

Eller/ MIS 
Copyright © 2013, Arizona Board of Regents

In the next module we will be discussing Physical and Network Security, including:

- Physical, Workstation, and Server Security
- Facilities Considerations (HVAC, Power, Fire)
- Media Considerations (Reuse/Destroy)
- Fax Security Considerations
- Network Topologies (Bus, Ring, Star, Bus-Star)
- Communications Media (Coax, Twisted Pair, Fiber, Wireless)
- Accepted Guidelines for Cable Installation
- Deploying Structured Wiring Design
- Implementing Structured Network Design
- Vertical Wiring Principles
- Centralized Management
- Virtual LANs
- Network Redundancy
- OSI Model Implications

Resources



- Eckel, E. (2002, November 15). Memorize file copying and moving traits for Win2K Server exam. *TechRepublic*. Retrieved from http://articles.techrepublic.com.com/5100-10878_11-1061203.html.
- Ext3. (2010, May 13). Wikipedia, the Free Encyclopedia. Retrieved from <http://en.wikipedia.org/wiki/Ext3>.
- Ext4. (2010, May 22). Wikipedia, the Free Encyclopedia. Retrieved from <http://en.wikipedia.org/wiki/Ext4>.
- Journaling File Systems. (2010, April 30). Wikipedia, the Free Encyclopedia. Retrieved from http://en.wikipedia.org/wiki/Journaling_file_system.
- Mac OS X: Mac OS Extended format (HFS Plus) volume and file limits. (2008, July 29). *Apple, Inc.* Retrieved from <http://support.apple.com/kb/HT2422>.
- Microsoft Corporation. (2003, March 28). NTFS Technical Reference. Retrieved from [http://technet.microsoft.com/en-us/library/cc758691\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc758691(WS.10).aspx).
- Posey, B. M. (2006, October 17). Recovering encrypted files from an NTFS partition. *SearchWindowsServer.com*. Retrieved from http://searchwindowsserver.techtarget.com/tip/0,289483,sid68_gci1224652_mem1,00.html?ShortReg=1&mboxCov=searchWindowsServer_RegActivate_Submit&.
- Rizzo, T. (2004, March 17). WinFS 101: Introducing the new Windows file system. *Microsoft Corporation*. Retrieved from <http://msdn.microsoft.com/en-us/library/aa480687.aspx>.
- Singh, A. (2003, December). Mac OS X file systems. *Mac OS X Internals*. Retrieved from http://osxbook.com/book/bonus/ancient/whatismacosx/arch_fs.html.
- Smith, R. (2008, January 21). Multilevel security. *Cryptosmith*. Retrieved from <http://www.cryptosmith.com/multilevel/>.