NTFS (New Technology File System) is the default file system of modern Windows, including Windows 10. It provides several security features (Microsoft, 2021a). Privileging, determining which users or groups can access or modify files or folders. Auditing, which logs access and changes to files or folders. Security Descriptors, consisting of user access identifiers for Access Control Lists, and control lists for accessing and auditing policies. Additionally, Security Descriptors allow file permissions to be defined. Files within the NTFS file system are also individually encrypted using the Encrypted File System, or EFS (Microsoft, 2023a).

Privileging is used in conjunction with the User Right’s found within Access Control Lists (Microsoft, 2023b). Privileges are held by a caller (Microsoft, 2023c). Privileges dictate system-level permissions of users and groups, such as the ability to modify any file even in cases where File Permissions may not permit such. Consequently, they are closely tied to the File System and provide an additional layer of security. Privileging was introduced alongside Access Control Lists in Windows NT 3.1 with NTFS in 1993. Earlier file systems, such as FAT (File Allocation Table) did not include privileging (Microsoft, 2008).

Auditing works in conjunction with security descriptors. If auditing is enabled for an object or event, then the security descriptor for the file will include an SACL (System Access Control List) (Microsoft, 2009a). This informs which actions should be audited. When an action that should be audited occurs, then a record is created in the Security-category of Window’s Event Viewer. This record includes information about the user, action preformed, and time occurred. Administrators may specify which objects are audited, including by whom and which user-class, as well recording of the permissions that led to that to that event being generated (Microsoft, 2022). Auditing was introduced apart of Windows NT 3.1 which released in 1993. The capabilities were enhanced in subsequent releases, with Windows NT 3.5 and Windows NT 4.0.

Security Descriptors are a critical component of security in the NTFS. They are a data structure that contains information about the security of an object (Microsoft, 2021b). They’re informed by Access Control Entries (ACEs) which Access Control Lists (ACLs) are made of. Security Descriptors are composed of three pieces of information (Microsoft, 2021c). First, a security identifier which determines the owner of an object, such as a user or group. Owners will always have access to an object. Second, a Discretionary Access Control List (DACLs) which defines a user or group’s access rights to an object, including level of access and basic file permissions; read, write, execute (Microsoft, 2021d). Third, a System Access Control List (SACLs) which specifies audit policy including which users or groups should be audited upon an attempt to access the object (Microsoft, 2021c). Security Descriptors also support inheritance. Child folders inherit permissions of the parent, with design in place to prevent security vulnerabilities from such (Microsoft, 2023d). Permissions will not be updated upon folder move for instance. Security Descriptors were implemented in Windows NT 3.1 with NTFS in 1993.

Encrypted File System (EFS) is a feature of NTFS that provides encryption of files or folder content. EFS uses a public-key system (Microsoft, 2023a). Files are encrypted with a randomly generated symmetric key (Microsoft, 2023e). This key is then encrypted with the user’s public-key. User’s must provide their matching private key to decrypt files to allow access. EFS uses AES-256 algorithm for encryption of a file’s data (Microsoft, 2023f). EFS was implemented into NTFS version 3.0 with the release of Windows 2000 (Microsoft, 2009b).

APFS (Apple File System) is the default file system of macOS. It provides several security features which are comparable to those found on Windows’ NTFS.

Encryption can be preformed on APFS at the file system level (Apple, 2020). Consequently, APFS encryption is integrated into the Operating System at a lower level than Windows’ EFS which is a system driver (Microsoft, 2023a). In terms of algorithms, APFS uses AES-XTS whereas EFS uses AES (Apple, 2020). Encryption keys on APFS are stored within the device on a component called the ‘Secure Enclave’ which maintains sensitive information separate from the CPU (Apple, 2021). In comparison, EFS’ encryption keys are stored within user data (Microsoft, 2023a). Unlike EFS’ public-private keys, APFS’ are solely symmetric (Apple, 2020). APFS also encrypts per-volume (per storage-drive) whilst EFS encrypts per-file.

APFS may alternatively be encrypted through FileVault, which encrypts the entire startup disk on a MacOS whereas APFS’ native encryption does so per-volume. Implementation differs such as how FileVault stores keys on the startup disk itself and uses AES-XTS-128 (Apple, n.d.-a; 2022a).

Access Rights, and therefore Privileging on APFS are handled by ACLs and BSD permissions (Apple, 2018). In contrast, NTFS uses only ACLs (Microsoft, 2021b). BSD permissions are based off POSIX/UNIX, and allow permissions to be set based off owner, a group, or other (Apple, 2011a). Permissions include execute, read, write or none and variant combinations of each with the other. They are set by the owner like with NTFS. This is more granular and simplistic compared to NTFS’ DACLs file permissions although not necessarily less secure.

ACLs were a later addition to MacOS and expanded permissions to be like NTFS (Apple, 2011b). For instance, ACLs allowed a permission group such as granting ability only to read file attributes to exist, which macOS’s BSD permissions could not allow. APFS’ ACLs with extended attributes are consequently the closest thing macOS has to Windows’ Security Descriptors.

APFS possesses an audit system largely through Apple’s Unified Logging System (Apple, n.d.-b). Whilst APFS possesses auditing like NTFS, it is much more limited in ability. APFS allows basic similar functionalities and viewing of data through binary form. NTFS, however, grants more control over audit policy which APFS does not. This is because NTFS through Security Descriptors can specify audit policy for different users and groups (Microsoft, 2021c).

Besides those, APFS has security features which NTFS lacks. One being System Integrity Protection (SIP) which aims to prevent file tampering from malicious software (Apple, 2022b). NTFS does not have anything comparable. SIP prevents modification of root files without special entitlement from Apple.

APFS also creates snapshots, being read-only copies of earlier file versions which can be restored (Apple, 2022c). NTFS does not have comparable. Snapshots are created automatically each hour and are stored for around a day. This improves security by allowing restoration to an undamaged system state.

The security features of NTFS may influence Application Developers in their work. For instance, with Privileging and Security Descriptors; an Application Developer will need to ensure that their application has correct file system permissions to function correctly. This is important as many applications often require access to other files. An example is an IDE needing permissions to read, write and potentially execute files.

Privileging may also involve data protection. Application Developers must pay attention to ensure that data cannot be accessed by the wrong people. Folders or files created must pay attention to this. An example could be with a Graphic Editor, such as Photoshop. Files utilised or created by such may contain metadata or other attributes that other users should not be able to see.

Similarly, privileging should also pay attention to security. They should not create vulnerabilities in the system. This could be done if a program does not sanitise user inputs and allows file executions allowing malicious actions or attacks to potentially occur.

Application Developers must also pay attention to Encryption. If application sends or receives encrypted data, then the application must have the necessary permissions and capability to handle such. Similarly, applications that handle encrypted data must handle them correctly; they should delete sensitive data when no longer necessary. They should also ensure that information cannot be accessed by unauthorised users.

Encryption may also impact application performance and load; may make it difficult to handle file sharing; may make back-up or restoration a consideration during development; and may have an impact on application compatibility between users or computers.

Auditing may help Application Developers in adhering to security compliance. Logs will show issues that an Application Developer can use to bugfix or make their application more secure. They may also reveal information which can be used in improving performance or curtailing to the user experience.

Overall, there are several ways NTFS may be improved. APFS has functionalities which NTFS is lacking. Snapshots would improve system resiliency, especially against specific attacks such as ransomware. Snapshots would allow ransomware attacks to be reversed and data to be restored. Snapshots would also be useful in case of errors or issues following software installation.

Integrity Protection would also benefit NTFS. It would prevent tampering of system critical infrastructure by users or malicious actors, ensuring protection against data corruption or accidental user errors.

NTFS’s security may also be improved through other methods. Better user education may protect against vulnerabilities; learning tools could be integrated through notifications or other Windows’s functionalities. Integration with additional security software may also be beneficial. These may include intrusion detection systems; anti-virus; threat intelligence tools.

# References

Apple, 2011a. *BSD Permissions and Ownership.* [Online]   
Available at: https://developer.apple.com/library/archive/documentation/MacOSX/Conceptual/BPFileSystem/Articles/BSDInfluences.html#//apple\_ref/doc/uid/20002286-SW1  
[Accessed March 2023].

Apple, 2011b. *Access Control Lists.* [Online]   
Available at: https://developer.apple.com/library/archive/documentation/MacOSX/Conceptual/BPFileSystem/Articles/ACLs.html  
[Accessed March 2023].

Apple, 2018. *File System Basics.* [Online]   
Available at: https://developer.apple.com/library/archive/documentation/FileManagement/Conceptual/FileSystemProgrammingGuide/FileSystemOverview/FileSystemOverview.html#//apple\_ref/doc/uid/TP40010672-CH2-SW20  
[Accessed March 2023].

Apple, 2020. *Encryption.* [Online]   
Available at: https://developer.apple.com/support/downloads/Apple-File-System-Reference.pdf  
[Accessed March 2023].

Apple, 2021. *Secure Enclave.* [Online]   
Available at: https://support.apple.com/en-gb/guide/security/sec59b0b31ff/web  
[Accessed March 2023].

Apple, 2022b. *About System Integrity Protection on your Mac.* [Online]   
Available at: https://support.apple.com/en-gb/HT204899  
[Accessed March 2023].

Apple, 2022c. *About Time Machine local snapshots.* [Online]   
Available at: https://support.apple.com/en-gb/HT204015  
[Accessed March 2023].

Apple, 2022. *Use FileVault to encrypt your Mac startup disk.* [Online]   
Available at: https://support.apple.com/en-gb/HT204837  
[Accessed March 2023].

Apple, n.d.-a. *Encrypt Mac data with FileVault.* [Online]   
Available at: https://support.apple.com/en-gb/guide/mac-help/mh11785/mac  
[Accessed March 2023].

Apple, n.d.-b. *Generating Log Messages from Your Code.* [Online]   
Available at: https://developer.apple.com/documentation/os/logging/generating\_log\_messages\_from\_your\_code  
[Accessed March 2023].

Microsoft, 2008. *NTFS Security Benefits.* [Online]   
Available at: https://learn.microsoft.com/en-us/previous-versions/windows/embedded/ms913208(v=winembedded.5)?redirectedfrom=MSDN  
[Accessed March 2023].

Microsoft, 2009a. *How Security Descriptors and Access Control Lists Work.* [Online]   
Available at: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc781716(v=ws.10)?redirectedfrom=MSDN  
[Accessed March 2023].

Microsoft, 2009b. *Overview.* [Online]   
Available at: https://learn.microsoft.com/en-us/previous-versions/tn-archive/dd277410(v=technet.10)  
[Accessed March 2023].

Microsoft, 2021a. *Security Features for File Systems.* [Online]   
Available at: https://learn.microsoft.com/en-us/windows-hardware/drivers/ifs/security-features-for-file-systems  
[Accessed March 2023].

Microsoft, 2021b. *Security Descriptors.* [Online]   
Available at: https://learn.microsoft.com/en-us/windows-hardware/drivers/kernel/security-descriptors  
[Accessed March 2023].

Microsoft, 2021c. *Security descriptors in file systems.* [Online]   
Available at: https://learn.microsoft.com/en-us/windows-hardware/drivers/ifs/security-descriptors  
[Accessed March 2023].

Microsoft, 2021d. *Access Rights.* [Online]   
Available at: https://learn.microsoft.com/en-us/windows-hardware/drivers/kernel/access-rights  
[Accessed March 2023].

Microsoft, 2022. *Basic security audit policies.* [Online]   
Available at: https://learn.microsoft.com/en-us/windows/security/threat-protection/auditing/basic-security-audit-policies  
[Accessed March 2023].

Microsoft, 2023a. *File Encryption.* [Online]   
Available at: https://learn.microsoft.com/en-us/windows/win32/fileio/file-encryption  
[Accessed March 2023].

Microsoft, 2023b. *Access Control Overview.* [Online]   
Available at: https://learn.microsoft.com/en-us/windows/security/identity-protection/access-control/access-control  
[Accessed March 2023].

Microsoft, 2023c. *Managing privileges in a file system.* [Online]   
Available at: https://learn.microsoft.com/en-us/windows-hardware/drivers/ifs/privileges  
[Accessed March 2023].

Microsoft, 2023d. *Inherited permissions are not automatically updated when you move folders.* [Online]   
Available at: https://learn.microsoft.com/en-us/troubleshoot/windows-server/windows-security/inherited-permissions-not-automatically-update  
[Accessed March 2023].

Microsoft, 2023e. *Back up the recovery agent Encrypting File System (EFS) private key in Windows.* [Online]   
Available at: https://learn.microsoft.com/en-us/troubleshoot/windows-server/identity/back-up-recovery-agent-efs-private-key  
[Accessed March 2023].

Microsoft, 2023f. *Files are corrupted after you encrypt them with ECC certificates by using the EFS feature in Windows 7 or in Windows Server 2008 R2.* [Online]   
Available at: https://support.microsoft.com/en-gb/topic/files-are-corrupted-after-you-encrypt-them-with-ecc-certificates-by-using-the-efs-feature-in-windows-7-or-in-windows-server-2008-r2-6d45e672-d466-b1ff-452a-b71850f66fe8  
[Accessed March 2023].

**GitHub:** https://github.com/8raLe/comp6015