|  |  |  |  |
| --- | --- | --- | --- |
| **BASIC PERMISSION** | **EFFECT ON FILES** | **EFFECT ON DIRECTORIES** | **PRACTICAL EFFECT ON DIRECTORIES** |
| **r (read)**  **mode = 4** | 1. Files can be **read**.  2. If only this permission is granted to files then files can only be read. | 1. Contents of the directory (only filenames without any other details) can be listed.  2. Only **filenames** from the filesystem metadata can be queried. | 1. If only this permission is granted then only filenames can be listed without any other details like permissions and timestamps.  2. If only this permission is granted **files cannot be accessed**.  3. It is the permission to **list directory contents**.  4. Can be used to just get only the names of files and subdirectories in parent directory by listing contents.  5. Tab completion will work as it is permission to list filenames.  6. In short it just allows access to **filenames not inodes** that is the reason we are unable to get permissions, timestamps and other data. |
| **w (write)**  **mode = 2** | 1. Files can be **changed**.  2. If only this permission is granted to files then content can only be written or appended to the files. | 1. Any file in the parent directory can be created or deleted as contents of a directory are filenames not the actual file contents.  2. This does not mean that you can delete subdirectories that you do not own as files residing in those subdirectories are contents of those subdirectories not the main parent directory. | 1. Files can be **created or deleted or renamed** by any user having write permission on the parent directory provided that execute bit is set on the parent directory.  2. Files can be deleted by the user regardless of the ownership or permissions of the files.  3. It is the permission to **modify directory contents**.  4. This permission **does not allow a user to modify file contents** it only allows creating or renaming or removing a file if a user has write permission on parent directory.  5. If only this permission is granted then the user cannot create or delete files in a parent directory because for this permission to work **execute permission is necessary** as files are actually stored on inodes.  6. This property can be **limited by setting sticky bit**. |
| **x (execute)**  **mode = 1** | 1. Files can be **executed** as commands.  2. If only this permission is granted to **script files** then they cannot be executed as for executing a script file first reading it is necessary.  3. If only this permission is granted to **binary files** then they can be executed. | 1. Contents of the directory can be accessed depending on the permissions of the files and subdirectories in the directory.  2. Allows querying of **inodes and their contents** from filesystem. | 1. File names cannot be listed.  2. Files can be accessed if the exact file name is known.  3. Files can be accessed only if the user has permission to read or write the files.  4. It is the permission to **access directory contents**.  5. If only this permission is granted then the user cannot list the files contained in a directory but can **access a file** if the user knows the filename and also owns it.  6. Tab completion will not work as it is not the permission to list filenames.  7. In short it is the **permission that really allows the user to access any file content if the file permissions allow it**.  8. In short it just allows access to **filenames and inodes** that is the reason we are able to get permissions, timestamps and other data and this is the reason why execute permission is necessary for the write permission to work. |

**LINUX BASIC FILE PERMISSIONS**