README

# Directory/Files Details:

|  |  |
| --- | --- |
| **File/Folder name** | **Description** |
| SourceCode | This directory has source code and supporting files. |
| Design\_Document\_CS551\_P3\_  Group8 | This document has design details, program flow chart, exception handling. |
| Test\_Case\_Document\_CS551\_P  3\_Group8 | This Document contains Test case details and their output. |
| Vm file link | Download vm file and import into VMware virtual machine from the below link |

In the SourceCode directory

* kernel -> Directory, Has diff of changes in kernel and the corresponding files.
* damagetool -> Directory, Has user code and an executable to damage filesystem.
* recovertool -> Directory, Has user code and an executable to recover filesystem.
* BootImage -> Directory, Enhanced minix boot image.

**Minix Version**: Minix 3.2.0.

**Compiler**: Clang

# Setting up the new Minix:

1. If you want to use boot image then
2. Copy the files in 3.2.0 directory to /boot/minix\_latest/ directory in your minix machine.
3. Reboot the minix machine.

**(or)**

1. If you want to compile enhanced minix from scratch
2. Goto kernel/ directory
3. unistd.h 🡪 Added syscall declarations. Copy the file (or the changes) to /usr/include/ and /usr/src/include
4. callnr.h 🡪 Added syscall numbers. Copy the file (or the changes) to /usr/include/minix and /usr/src/include/minix
5. vfsif.h 🡪 vfs specific. Copy the file (or the changes) to /usr/include/minix and /usr/src/include/minix
6. vfs/proto.h 🡪 Copy the file (or the changes) to /usr/src/servers/vfs
7. vfs/table.c 🡪 Added syscalls in call vector in VFS. Copy the file (or the changes) to /usr/src/servers/vfs
8. vfs/request.c 🡪 For communication with MFS. Copy the file (or the changes) to /usr/src/servers/vfs
9. vfs/myvfslink.c 🡪 Added syscalls in call vector in VFS. Copy the file (or the changes) to /usr/src/servers/vfs
10. vfs/Makefile 🡪 Add the changes to /usr/src/servers/vfs
11. mfs/proto.h 🡪 Copy the file (or the changes) to /usr/src/servers/mfs
12. mfs/table.c 🡪 Added syscalls in call vector in MFS. Copy the file (or the changes) to /usr/src/servers/mfs
13. mfs/mymfslink.c 🡪 Added syscalls in call vector in MFS. Copy the file (or the changes) to /usr/src/servers/mfs
14. mfs/Makefile 🡪 Add the changes to /usr/src/servers/mfs

## Compilation:

1. Kernel
2. goto /usr/src/tools

make hdboot

reboot

1. User tools
2. Copy damagetool to some directory in minix machine.
   1. cd damagetool
   2. make
3. Copy recovertool to some directory in minix machine.
   1. cd recovertool
   2. make

## Steps to run the program:

**Damagetool**

1. goto damagetool directory
   1. ./damageFileSysteTool
2. When the program is executed you will see below options. Pick appropriate options.

# ./damageFileSystemTool

DAMAGE FILE SYSTEM TOOL – please select your choice

1. Delete the file without damaging the folder
2. Damage the inode bit Map by removing the file.
3. Currupt the Directory file completely.
4. Damage the inode time to damage the inode bit map.
5. Damage the zone bit map.
6. Damage the Directory file by corrupting its inode completely
7. Exit

Enter your choice ->

**Recovertool**

1. goto recovertool directory
   1. df

# df

1. Filesystem Size (kB) Free Used % Files% Mounted on
2. /dev/c0d0p0s0 65536 41572 23964 37% 3% /
3. /dev/c0d0p0s2 1699280 1207300 491980 29% 5% /usr
4. /dev/c0d0p0s1 331776 326524 5252 2% 1% /home
   1. ./recoverFileSystemTool /dev/c0d0p0s0
5. When the program is executed it recovers file system on the device provided.

**Functionality:**

Described in the Design Doc.

**Testing:**

Described in the Test cases Doc.