File Descriptors and Redirections in Linux

What Are File Descriptors?

A **file descriptor (FD)** is a reference maintained by the Linux kernel for managing I/O operations on files, sockets, etc.

- Similar to a "ticket" used to identify a file/resource.
- Windows equivalent: **file handle**.

Default File Descriptors

FD	Stream	Description
0	STDIN	Standard Input
1	STDOUT	Standard Output
2	STDERR	Standard Error Output

Redirecting File Descriptors

1. Redirecting STDERR (FD 2) to Null

```
find /etc/ -name shadow 2>/dev/null
```

Suppresses permission errors.

2. Redirect STDOUT (FD 1) to a File

```
find /etc/ -name shadow 2>/dev/null > results.txt
```

3. Redirect STDOUT and STDERR Separately

```
find /etc/ -name shadow 1>stdout.txt 2>stderr.txt
```

4. Redirect STDIN (FD 0) from a File

```
cat < stdout.txt
```

5. Append STDOUT Instead of Overwriting

```
find /etc/ -name passwd >> stdout.txt 2>/dev/null
```

6. Redirect STDIN with Stream (Here Document)

```
cat << EOF > stream.txt
Hack The Box
EOF
```

Using Pipes (`|`) for Output Chaining

Example: Filter Output with `grep`

```
find /etc/ -name "*.conf" 2>/dev/null | grep systemd
```

Count Filtered Results

```
find /etc/ -name "*.conf" 2>/dev/null | grep systemd | wc -l
```

Summary

- File descriptors allow you to **control how data flows** in Linux.
- Redirection (`>`, `<`, `>>`, `2>`) helps isolate or suppress output.
- Pipes (`|`) let you pass output from one command into another.

Understanding these concepts helps streamline system tasks and improve efficiency.

Author: Aizakku04@htb
Topic: Linux File Descriptors, Redirection, and Pipes