

Cedar Rapids Area Homeschools Cyber Defense Club**Date** 2025-11-08**Presenter** Chris Leonard**Document** Teaching Notes

Using bash (the Bourne-Again Shell)



Bash Lesson Plan (90 minutes)

0. Setup (Before Class)

- Make sure each student has a terminal open (Linux, macOS, or WSL on Windows).
- Create a safe “sandbox” directory for them to experiment in:

```
mkdir ~/bash_playground && cd ~/bash_playground
```

- If needed, have them install and test **tldr**.

```
sudo apt update
sudo apt install tldr
tldr --update
tldr ls
```

1. Orientation (10 min)

- **Explain:** Bash is a command-line shell—like talking directly to the computer.
 - **Demo:** **pwd**, **ls**, **whoami**
 - **Quick challenge:** “Where are you right now?” (use **pwd**).
-

2. Filesystem Basics (25 min)

- **Teach:** **pwd**, **ls**, **cd**, **mkdir**, **touch**, **rm**, **rmdir**
 - **Mini-challenges:**
 - Create a folder called **school** and inside it make **class/notes**.
 - Make a file called **hello.txt** and delete it.
 - Race: who can build **school/class/notes** fastest?
-

3. Viewing Files (15 min)

Teach: **cat**, **less**, **head**, **tail**

Mini-challenges:

- Show first 5 lines of `/etc/passwd`.
- Show last line of `/etc/passwd`.
- Try to read `/etc/shadow` → discuss permissions.

BONUS!

- You'll probably want to edit files when you are using a `bash` shell.
- New users tend to prefer using `nano`.
- Create a new file in your `bash_sandbox` directory by using `touch`:

```
touch ~/bash_sandbox/test_file.txt
```

- Open the file in `nano`:

```
nano ~/bash_sandbox/test_file.txti
```

- Put into the file the names of two nursery rhymes you liked when you were little.
- Now you want to save the file. Look at the bottom of the screen. Nano tells you to type `^O` (`Ctrl-O`) to "write out" the file, which means saving the file. Do this to save your file.
- The other most important action in the `nano` menu is `[^X] Exit`. Type `Ctrl-X` to leave `Nano`.
- After you exit `nano`, use commands we discussed earlier to see what's in the file and then delete the file.

4. Permissions & Ownership (15 min)

- **Teach:** `ls -l`, `chmod`
- **Mini-challenges:**
 - Who owns `/etc/passwd`?
 - Make a file only you can read (`chmod 600 myfile`).
 - Remove all permissions (`chmod 000 myfile`) and see what happens.

5. Searching & Filtering (15 min)

- **Teach:** `grep`, `wc`, `sort`, `uniq`
- **Mini-challenges:**
 - Count users in `/etc/passwd` (`wc -l`).
 - Find all users with `/bin/bash`.
 - Sort users alphabetically.

6. Intro to Variables & Scripting (10 min, if time)

- **Teach:** variables, `if`, `while`

- **Demo only:**

```
name="Chris"
echo "Hello, $name"
```

```
if [ -f /etc/passwd ]; then
    echo "File exists!"
fi
```

```
count=1
while [ $count -le 5 ]; do
    echo "Welcome $count times"
    count=$((count+1))
done
```

✨ With this structure, you'll have a **tight rhythm of teach → challenge → discuss**, which keeps students engaged. The 12th graders can act as "coaches" during challenges, reinforcing their own knowledge while helping younger students.