EECS Tutorial: cslab Linux Environment Web Access

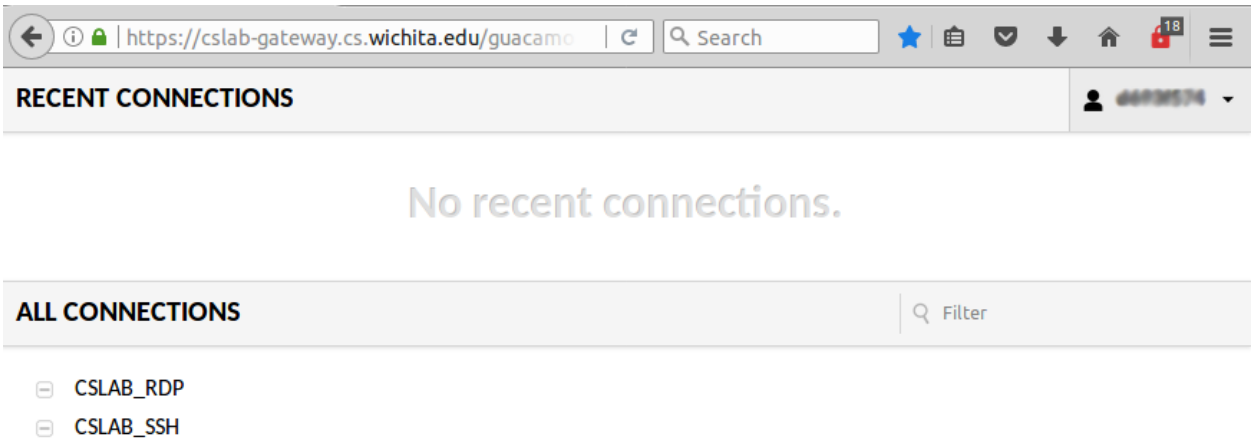
FAQ for web access into cslab Linux environment

For direct SSH access into cslab see eecs_tutorial_cslab_ssh_access

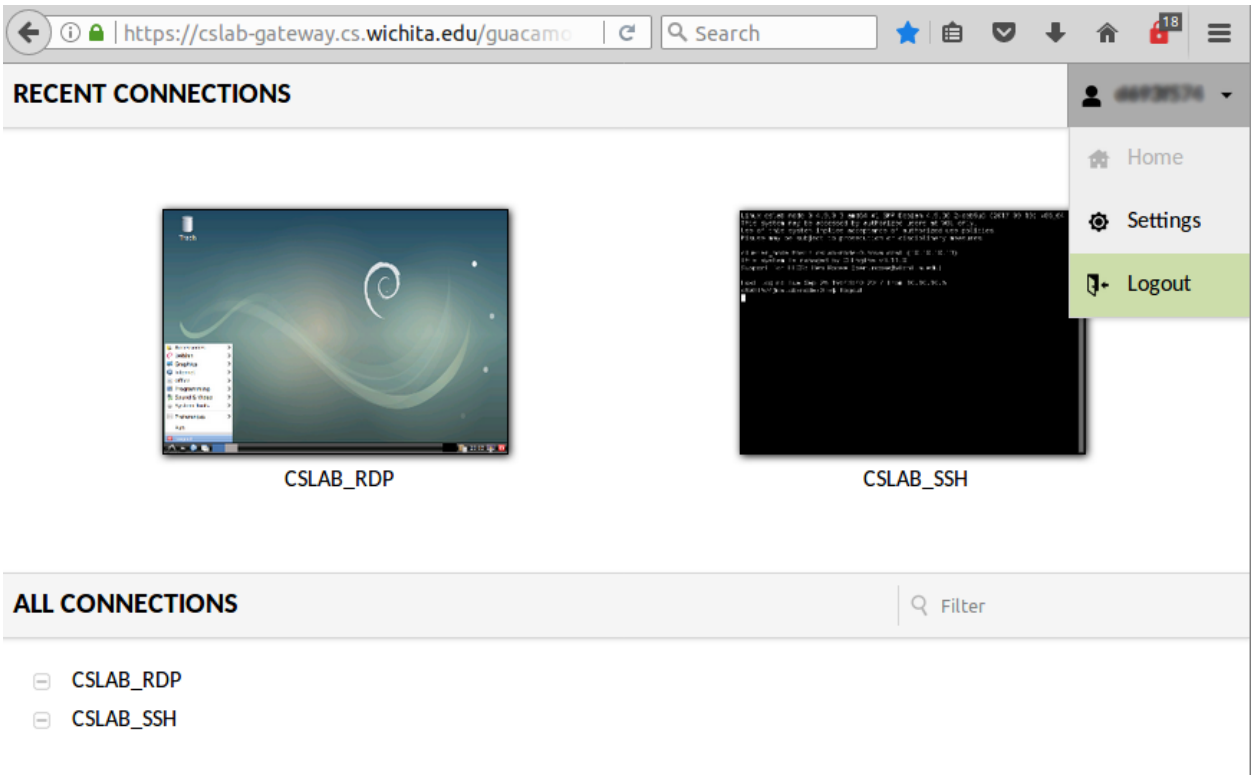
How do I access the cslab Linux environment via a web-browser?

1. Open your favorite HTML5 compatible web-browser. To test the compatibility of your browser go to html5test.com
2. In your browser go to the cslab *guacamole* interface at cslab-gateway.cs.wichita.edu
3. At the login screen enter your myWSU ID and password and you will be presented with the *Apache Guacamole* home screen.

   Enter your myWSU ID with only lowercase letters for the login username.



4. To connect into an LXDE graphical RDP desktop session for working on programming assignments click on `CSLAB_RDP`.
5. To connect into a command-line SSH terminal session for working on programming assignments click on `CSLAB_SSH`.
6. When you next log into cslab-gateway.cs.wichita.edu the *Apache Guacamole* home screen will show clickable thumbnails of your recent connections.
7. To log out of *Guacamole* from the home screen, click on your `myWSU_ID` at the top right. This drop-down list also enables you to open the `Settings` menu which includes displaying an on-screen keyboard for mobile devices.
8. During your initial connection into an RDP desktop session, occasionally you may see a policykit error message pop-up window. This is a known software bug which is being worked on. Clicking the OK button will close the error and it should not affect the rest of your login session.
9. To open/close the *Guacamole* menu sidebar while in the cslab environment press the key combination **Ctrl+Alt+Shift**. The *Guacamole* menu sidebar enables you to log out, disconnect, change settings, upload/download files, and use a remote clipboard.



10. **When you finish your work session, please make sure to disconnect/logout from your connection within the cslab environment:**
    - **by using the *Guacamole* menu sidebar and clicking on your [username].**
    - **by using the [logout] taskbar button on bottom right or [logout] menu item on bottom left within the RDP desktop session.**
11. **NOTE: All cslab-nodes automatically reboot every night between 2am and 3am. Any programs/processes left running will be killed during the reboot cycle.**
12. For further help on using the *Guacamole* interface go to Using Guacamole Guide

What am I allowed to do within the cslab Linux environment?

- You are allowed to use all software installed on the cslab Linux environment for their intended purpose of educatio and training in your programming and computer science classes at Wichita State University.
- When you log into the cslab environment you are accepting responsibility and accountabily for the commands, programs, and proccesses run by your Linux user account and the data stored within your user account. This is a position of trust: the university trusts that you will use cslab responsibly.

- If it is deemed by your instructor or the EECS systems administrator that commands, programs, or processes have been used by your user account on the cslab-nodes and associated servers for malicious purposes or outside of their intended purposes, then your access into cslab may be temporarily or permanently removed and there may be additional academic consequences to your actions.

Never share your myWSU password with other users.
Never attempt to use root or sudo privileged commands within cslab environment!

What software is available within the cslab Linux environment?

- cslab environment gives you both graphical and command-line Linux tools for writing, compiling, and debugging your CS programming class assignments. The Linux operating system running in cslab is Debian 9 (stretch) with a default LXDE desktop and Bash shell. Software tools/packages installed on the cslab-nodes include:
  - Text and code editors: leafpad, nano, vim, and emacs.
  - Integrated development environments (IDE): atom, geany, and eclipse.
  - Compiling tools: GNU C compiler (gcc), g++, make, java, perl, python, python-virtualenv, haskell-compiler, prolog, flex, and bison.
  - Debugging tools: GNU debugger (gdb) and data display debugger (ddd).
  - Latex tools: pdflatex, texlive, and texmaker.
  - Version control tools: git and subversion.
  - IRC clients: hexchat, weechat, and irssi.
  - GUI terminal emulators: terminator, lxterminal, and xterm.
  - CLI terminal multiplexers: screen and tmux.
- To check if a specific software package or version is installed within the cslab-nodes, connect into the SSH terminal session or open a terminal emulator in the RDP desktop session and type:

```
apt list --installed specify_package_name_here
```

- If a Linux software package is not installed within the cslab Linux environment which you require to complete your class assignment, then please ask your instructor whether this package can be installed for you by the EECS systems administrator.

Do not attempt to install any software packages yourself!

How do I copy text within the cslab Linux environment?

Copying text from your local computer to the remote environment:

1. Copy the required text to the clipboard within your local computer application using your preferred method of copying, i.e. **Ctrl+C**.
2. Within the cslab environment in your browser, open the *Guacamole* menu sidebar by pressing the key combination **Ctrl+Alt+Shift**.
3. Paste the copied text to the remote *Guacamole* [Clipboard] field using your preferred method, i.e. **Ctrl+V**.
4. Close the *Guacamole* menu sidebar by pressing the key combination **Ctrl+Alt+Shift**.
5. Within the RDP desktop session, any text shown in the *Guacamole* [Clipboard] can be pasted into a remote cslab application by normal methods, i.e. **Ctrl+V**.
6. Within the SSH terminal session, text in the *Guacamole* [Clipboard] can be pasted into the terminal by right-clicking on the browser window with your mouse or by pressing the key combination **Ctrl+Shift+V**.

Copying text from the remote environment to your local computer:

1. Within the RDP desktop session, text can be cut or copied from any cslab application by normal cut/copy methods, i.e. **Ctrl+C**.
2. Within the SSH terminal session, text to be copied is ”highlighted” using the mouse.
3. Open the *Guacamole* menu sidebar by pressing the key combination **Ctrl+Alt+Shift**.
4. The copied or ”highlighted” text will appear in the remote *Guacamole* [Clipboard] and can then be selected and copied to your local computer clipboard, i.e. **Ctrl+C**.
5. Close the *Guacamole* menu sidebar by pressing the key combination **Ctrl+Alt+Shift**.

How do I download/upload files within the cslab Linux environment?

Using guacctl/guacget to download a file:

- Within the *guacamole* SSH terminal session, you can use the *Guacamole terminal session control utility (guacctl)* for downloading files. To download a file from the SSH terminal session to your local computer via the web-browser type:

  ```
  guacget file_to_be_downloaded
  ```

- `guacget` is an alias for the command `guacctl --download`. To see all the option flags available for this command type `guacctl`.

- **NOTE: `guacget` only works in SSH terminal sessions. `guacget` does not work in RDP desktop sessions.**

Drag-and-drop to upload a file:

- You can drag-and-drop a file from your local computer onto the cslab web-browser window. This can be used in both RDP desktop and SSH terminal sessions.

- By default the file is uploaded into your user home directory on the remote cslab-node.

- You can set a custom destination directory for future uploaded files when using drag-and-drop by typing:

  ```
  guacctl -s custom_upload_directory
  ```

Guacamole file browser to upload or download a file:

1. Open the *Guacamole* menu sidebar by pressing the key combination **Ctrl+Alt+Shift**.

2. Click on the disk drive icon under [Devices] to open a file browser of the remote cslab-node.

3. Browse to your user home directory on the remote server. You can then browse to subdirectories within your user home. Your home directory full path on the remote cslab-node will look like the following:

   ```
   stu##/your_myWSU_id/
   ```

4. If you are unsure where your home directory is located on the remote cslab-node, in a terminal or in the SSH terminal session type `pwd` to show you the full path of your present working directory.

5. Downloads are initiated by double-clicking on any file shown, while uploads are initiated by clicking the [Upload Files] button. Clicking [Upload Files] will open a file browsing dialog where you can choose one or more files from your local computer, ultimately uploading the selected files to the directory currently displayed within the remote cslab-node file browser.

6. Close the *Guacamole* menu sidebar by pressing the key combination **Ctrl+Alt+Shift**.

Can I open multiple web-browser tabs/windows into the cslab Linux environment?

- cslab *guacamole* interface restricts RDP desktop sessions to a single connection per user. You can view the remote RDP desktop session from only one web-browser tab/window at a time. If you attempt to open more than one web-browser tab/window and connect to a second RDP desktop session, then the gateway will present an error message in the second tab.

- cslab textitguacamole interface restricts SSH terminal sessions to a maximum of four concurrent connections per user. You can open up to four web-browser tabs/windows and connect into remote SSH terminal sessions at the same time. Each open SSH terminal session tab/window will run a separate command-line shell instance on the same remote cslab-node.

- You can use the `tmux` or `screen` terminal multiplexer commands to run multiple command-line shells concurrently in the same SSH session tab/window.

- You can open one *guacamole* web-browser tab/window to connect to a graphical RDP desktop session and open additional tabs/windows to connect to SSH terminal sessions concurrently. However, the *Apache Guacamole* system cannot guarantee that your RDP desktop session and your SSH terminal session(s) will connect to the same remote cslab-node.

What are the currently known bugs/glitches within the cslab Linux environment?

Due to the cslab environment being so new, there are a few software bugs and glitches which you may see at times. Most of these bugs occur during operations that involve the new cslab environment interacting with the older CS servers and are not critical issues. Known bugs include:

- A policykit error message pops up when user first logs into a *Guacamole* RDP desktop session if other users are also logged in. This is a minor bug within policykit and can be easily mitigated by clicking *Okay* in the pop-up window.
- When the *handin* command is used in the cslab environment for programming assignment submission it may produce a `segmentation fault` error. This is caused by the older 32-bit *handin* program running on the cslab 64-bit processor architecture. The error is minor and does not affect submission of student assignments for grading.

If you experience a bug or error when using the cslab Linux environment which stops you from completing your programming assignments, please inform your instructor at your earliest convenience.