CHAPPELL UNIVERSITY

Internet Society Interplanetary Chapter

IPNSIG

LAB 2

WASTC SPECIAL EVENT

TCP/IP IN SPACE

Configure and test ION-DTN on a Raspberry Pi

LAURA CHAPPELL
laura@chappellu.com

SCOTT SPICER
scott@chappellu.com

Contents

Presenters: Laura Chappell and Scott Spicer

Laura Chappell and Scott Spicer are leading voices at the intersection of advanced networking, education, and the future of space communications. As collaborators with the **InterPlanetary Networking Special Interest Group (IPNSIG)**, they champion the global expansion of Delay/Disruption-Tolerant Networking (DTN) and its transformative potential for both terrestrial and interplanetary connectivity.

Laura Chappell, founder of **Chappell University**, is recognized worldwide as a premier network analyst and educator. Known for her engaging and insightful teaching style, she has spent decades training IT professionals, engineers, and educators to understand, optimize, and secure networks across every scale. Laura's recent work with IPNSIG extends her lifelong mission of "connecting people through understanding networks" into the next frontier: space. She focuses on translating DTN's technical depth into practical, human-centered knowledge that inspires collaboration across scientific, academic, and engineering communities.

Scott Spicer is an emerging technology strategist and space networking advocate whose work bridges education on communication systems, distributed networks, and real-world applications of DTN. He contributes to the development and educational implementation of DTN lab environments, focusing on accessible demonstrations of interplanetary networking concepts. His work emphasizes the practical setup and operation of ION-DTN nodes, contact graph configuration, and the translation of complex networking principles into hands-on learning experiences.

Together, Scott and Laura combine technical insight with an educator's clarity and a visionary perspective on the future of communication-one where humanity stays connected, even across the vast distances of the solar system.

Welcome to the Raspberry Pi ION-DTN Lab 2

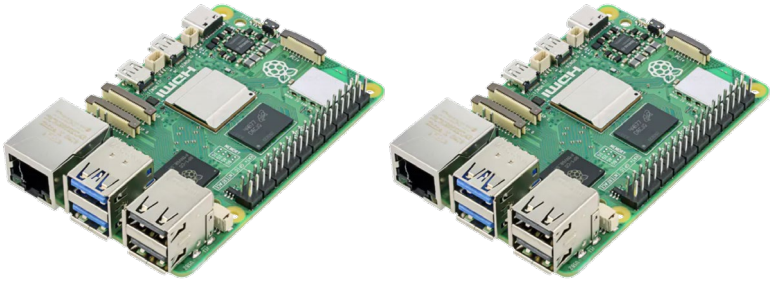
In this lab, we will examine show you how to set up a second Raspberry Pi 5 with ION to communicate between ION nodes over a network. We will examine the issues that may occur when communicating between ION nodes on separate networks, as well.

Prerequisites for Lab 2

Two Raspberry Pi-5s with at least 4 GB RAM, keyboard, mouse, monitor, power supply, internet access via wireless or LAN, and a minimum 64 GB microSD card.[1]

Both Raspberry Pi-5s should be configured as shown in the one-computer lab.

We will make adjustments to enable these two devices to talk to each other across the wireless network.



[1] This lab can be configured with a "headless" Raspberry Pi if you desire. Configuring the lab using a headless Raspberry Pi is beyond the scope of this lab.

Terminal Commands to Know

All terminal commands are case-sensitive.

`[up arrow]`	Displays last commands (nice to fix typos or see last command)
`ls -l`	Lists file and directories in the current location
`cd directory`	Change to a subdirectory
`cd ..`	Move up one directory
`cd ION [tab]`	Sample using tab to auto-complete a directory name
`cd ~`	Return to home directory
`clear`	Clear screen
`wget url`	Get a file from the web via command line
`unzip file`	Uncompress a file
`nano file`	Open a file with the nano command-line editor tool
`cat file`	Display contents of a file
`export variable`	Set an environment variable
`source script`	Executes a script in the current shell
`printenv`	Prints current environment variables for shell session
`sudo command`	Run programs with administrative (root) privileges
`make`	Compiles and builds software from source code
`hostname`	Display or set the current system hostname
`ifconfig`	View and configure network interfaces (legacy command)
`ip addr`	View IP addresses
`tail -f file`	Watch a file in real time (e.g., logs)
`mkdir name`	Create a new directory
`rm file`	Delete a file
`rm -r folder`	Delete a directory and its contents
`cp source target`	Copy a file
`mv source target`	Move or rename a file
`chmod +x file`	Make a file executable
`passwd`	Change your password
`sudo reboot`	Reboot the Pi (add `now` to immediately reboot)

Lab 2 List of Steps

The following steps define the configuration process for a two-computer ION-DTN test.

This lab requires two Raspberry Pi-5s configured and tested with Steps 1 through 8 of the one-computer lab.

Close any terminal windows you may have open and RESTART your Raspberry Pi used in the one-computer lab.

To help identify which computer you should enter commands on, we will highlight rpi01 instruction areas yellow and rpi02 instructions bright blue.

We will go through the following nine steps in this lab.

Step 1: Download the configuration files

Step 2: Assign hostnames* and obtain IP addresses

Step 3: Edit the /etc/hosts files

Step 4: Edit the .bprc files*

Step 5: Comment out the export path statement

Step 6: Set up three terminal sessions on each Pi

Step 7: Start ION and log file viewing

Step 8: Run a connectivity test (*bping*)

Step 9: Stream bundles (*bpdriver/bpcounter*)

Step 10: Transfer a file (*bpcp/bpcpd*)

* Lab steps will depend upon your hostnames in use.

Step 1: Download the Configuration Files

Let's begin by obtaining the configuration files for each computer.

Location: rpi01 home directory	
☐	`wget -O comp1iontest.zip "https://bit.ly/c1f-iontest"` (capital letter "O" – not a zero; the 5th character of the file name is a one, not L)
☐	`unzip comp1iontest.zip`
Location: rpi02 home directory	
☐	`wget -O comp2iontest.zip "https://bit.ly/c2f-iontest"` (capital letter "O" – not a zero)
☐	`unzip comp2iontest.zip`

You should now have *comp1iontest/iontest* and *comp2iontest/iontest* directories on your Raspberry Pi-5s.

Step 2: Assign Hostnames and Obtain IP Addresses

We will assign hostname *rpi01* to rpi01 and *rpi02* to rpi02. You can define the hostnames during the Pi OS imaging process if you are configuring a new microSD card.

We will change the hostnames in terminal sessions on each computer.

These new hostnames will be temporary on your Raspberry Pi-5s.

Location: rpi01 home directory	
☐	`sudo hostname rpi01` (if using a different hostname on this Pi)
☐	`ip addr` (obtain the IP address of rpi01) **[Note the IP address: _____]**
Location: rpi02 home directory	
☐	`sudo hostname rpi02` (if using a different hostname on this Pi)
☐	`ip addr` (obtain the IP address of rpi02) **[Note the IP address: _____]**

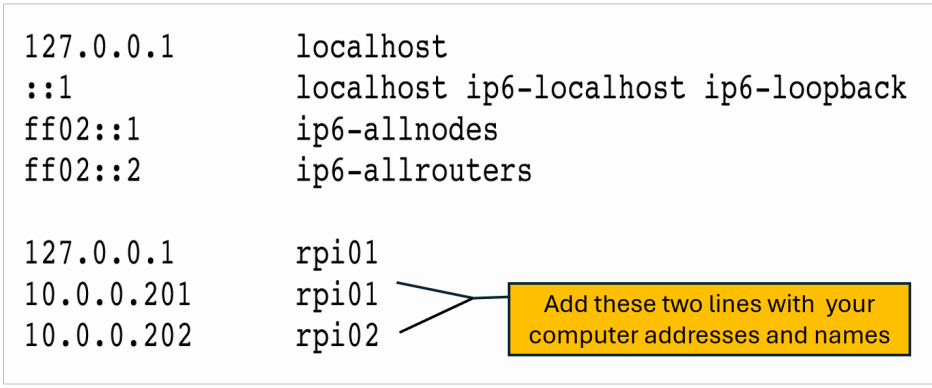
Now we will edit the */etc/hosts* file on each computer so we can use names instead of IP addresses in our test.

Step 3 provides a sample */etc/hosts* file when the IP address of *rpi01* is 10.0.0.201 and the IP address of *rpi02* is 10.0.0.202. Replace the IP addresses according to the IP addresses you discovered using `ip addr` in the steps above.

Step 3: Edit the */etc/hosts* Files

Our Lab 2 configuration files refer to the two Raspberry Pi-5s based on the names rpi01 and rpi02. We will need to edit the hosts files to add the IP addresses you learned in Step 2.

The following image shows the lines we have added based on the example in Step 2.



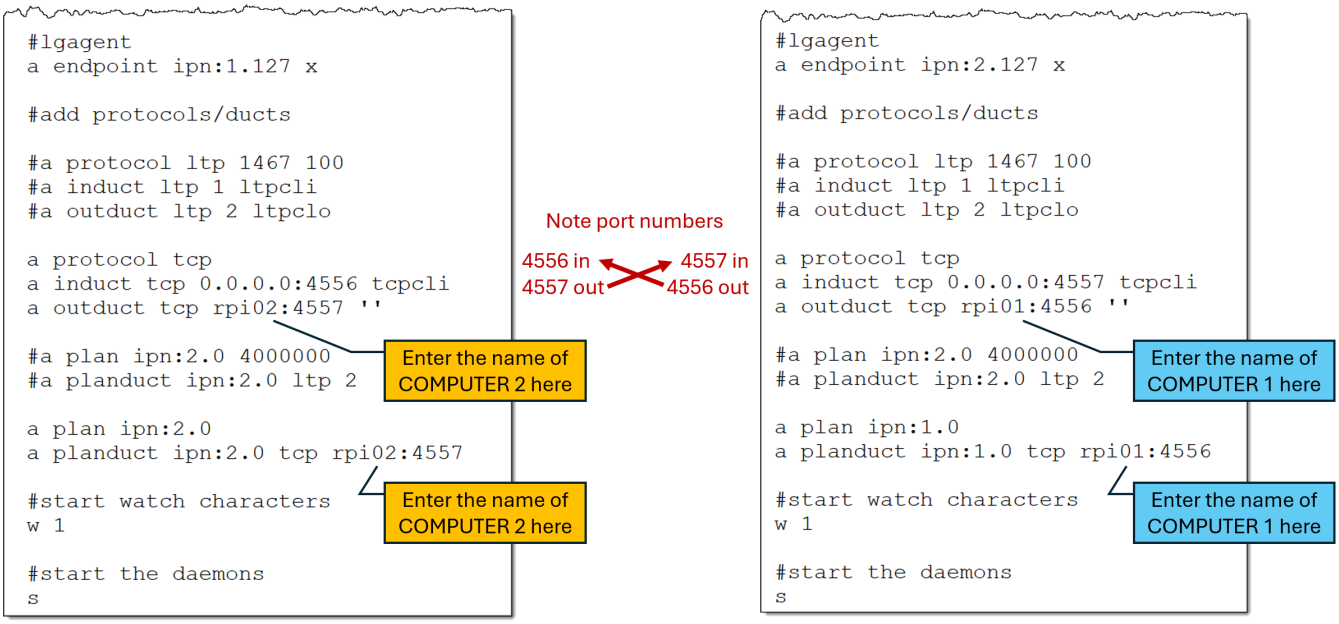
Location: rpi01 - any folder	
☐	`sudo nano /etc/hosts`
☐	Edit the *hosts* file to add the IP addresses of rpi01 and rpi02 (Make certain the 127.0.0.1 entry name is *rpi01*)
☐	Write out your changes and close *nano* (**Ctrl-o**, **return**, **Ctrl-x**)
Location: rpi02 - any folder	
☐	`sudo nano /etc/hosts`
☐	Edit the *hosts* file to add the IP addresses of rpi01 and rpi02 (Make certain the 127.0.0.1 entry name is *rpi02*)
☐	Write out your changes and close *nano* (**Ctrl-o**, **return**, **Ctrl-x**)

Step 4: Edit the *.bprc* Files[2]

In ION-DTN, *.bprc* files are Bundle Protocol Runtime Configuration files. These files are used to initialize and configure the Bundle Protocol (BP) engine on a node. They contain administrative commands that tell ION how to manage bundle endpoints, routing, and other BP-related features.

The lab .bprc configuration files are already set up with the hostnames of *rpi01* and *rpi02*. If you are using different hostnames, you must edit *.bprc* files on each computer.

The following images highlight the location of the changes required if you have named your computers differently.



Inducts use 0.0.0.0 to bind to everything using IPv4 and use port numbers to pass traffic up the stack to the application that is expecting the data. Host IP addresses can also be used.

Outducts use hostnames (based on the */etc/hosts* file) as a convenience. You may use IP addresses rather than hostnames if desired. If a host changes its IP address, you must change its entry in the */etc/hosts* file.

[2] This step is only required if you are **not** using the *rpi01* and *rpi02* hostnames.

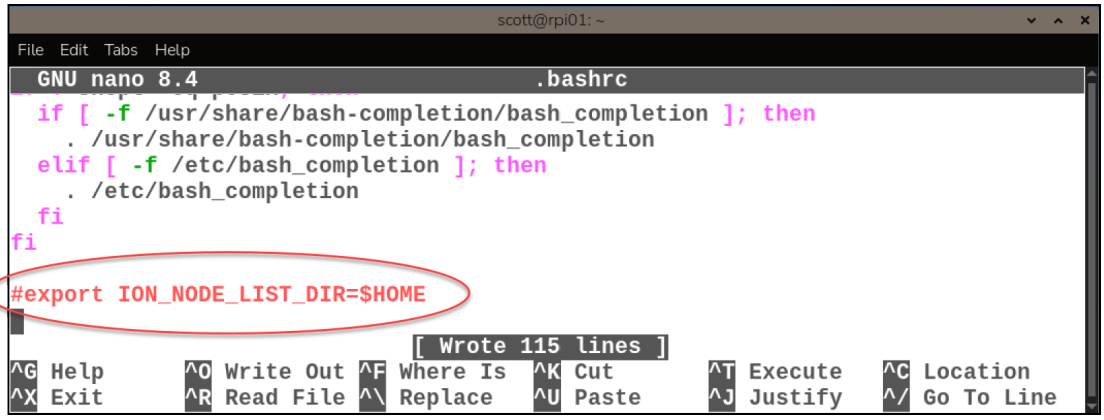
Again, you only need to change the hostnames if your hosts are **not** called *rpi01* and *rpi02*.

Location: rpi01 – home directory	
☐	`cd comp1iontest/iontest`
☐	`nano ipn1.bprc`
☐	Edit the *ipn1.bprc* file outduct and planduct lines to reflect the hostname of rpi02
☐	Write out your changes and close *nano* (**Ctrl-o**, **return**, **Ctrl-x**)
Location: rpi02 – home directory	
☐	`cd comp2iontest/iontest`
☐	`nano ipn2.bprc`
☐	Edit the *ipn2.bprc* file outduct and planduct lines to reflect the hostname of rpi01
☐	Write out your changes and close *nano* (**Ctrl-o**, **return**, **Ctrl-x**)

Step 5: Comment Out the Export Path Statement

If you are using a computer that was used for the one-computer lab, you must comment out the export line containing the path statement added in Lab 1.

Location: home directory of system with added environment variable	
☐	`cd ~` (to ensure you are in your home directory)
☐	`nano .bashrc`
☐	Go to the bottom of the *.bashrc* file and comment out this line (Add a `#` before the word `export`) `export ION_NODE_LIST_DIR=$HOME`
☐	Write out your changes and close *nano* (**Ctrl-o**, **return**, **Ctrl-x**)
☐	`source .bashrc`



Step 6: Set up Three Terminal Sessions on Each Pi

Set up three terminal sessions on rpi01 and three terminal sessions on rpi02. The following lists how we will use each terminal window.

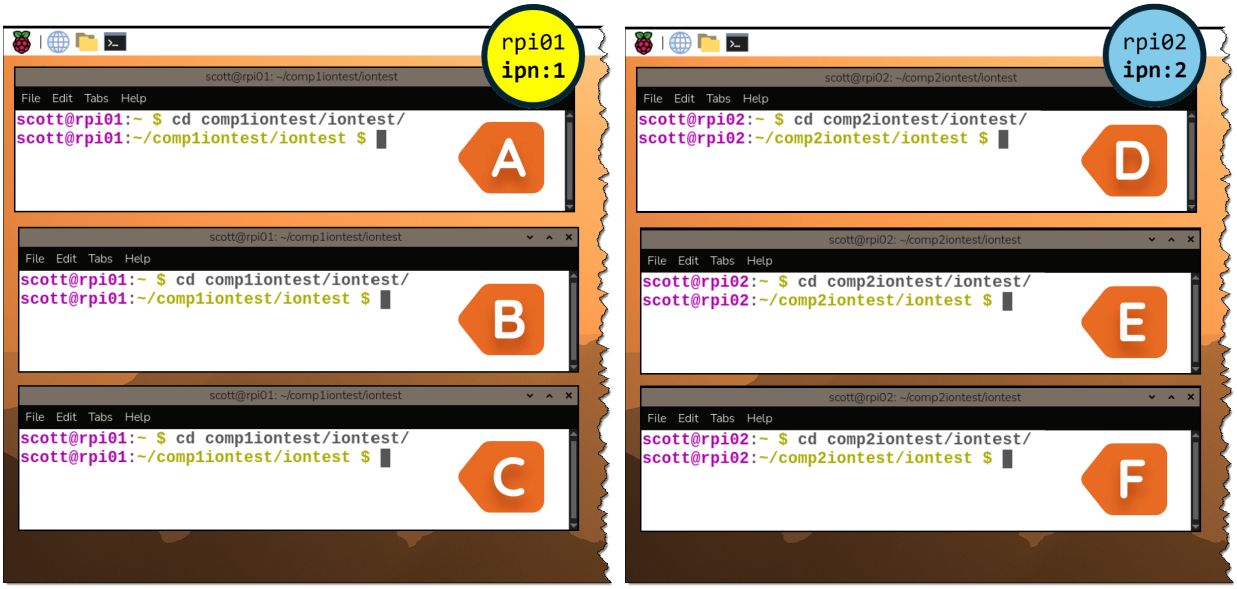
rpi01: Terminal Window A – launch ION, view watch characters
Terminal Window B – view ion.log entries
Terminal Window C – launch test commands

rpi02: Terminal Window D – launch ION, view watch characters
Terminal Window E – view ion.log entries
Terminal Window F – launch test commands

First, change into the appropriate ***iontest*** directories in each terminal window.

rpi01 Terminal Window A, B, and C comp1iontest/iontest Directory		**rpi02 Terminal Window D, E, and F comp2iontest/iontest Directory**	
☐	`cd comp1iontest/iontest`	☐	`cd comp2iontest/iontest`

Your Raspberry Pi-5s should look like the following images.



Step 7: Start ION and Log File Viewing

We will set up and launch *ionstart* in the top terminal windows of rpi01 and rpi02. We will also begin displaying the most recent ion.log entries in the second set of terminal windows.

rpi01 Terminal Window A comp1iontest/iontest Directory		**rpi02 Terminal Window D comp2iontest/iontest Directory**	
☐	`chmod +x ionstart`	☐	`chmod +x ionstart`
☐	`./ionstart`	☐	`./ionstart`

Next, we will set up the log file viewing terminals.

rpi01 Terminal Window B comp1iontest/iontest Directory		**rpi02 Terminal Window E comp2iontest/iontest Directory**	
☐	`tail -f ion.log`	☐	`tail -f ion.log`

Now you are ready to run ION-DTN tests between the two computers.

Step 8: Run a Connectivity Test (bping)

Now that our terminal windows are set up, we are ready to send DTN communications between the hosts. We will begin with *bping*.

We use *bping* initially to verify connectivity between hosts (just as we do with ping, which uses ICMP, on IP networks). Observe the watch characters in the top terminal windows.

Location: Terminal Window C comp1iontest/iontest Directory	
☐	`bping -c1 -q0 ipn:1.2 ipn:2.3`

This command sends one bping (`-c1`) from node 1.2 to node 2.3 (`ipn:1.2 ipn:2.3`) immediately (`-q0`). Note that *ionstart* automatically starts the *bpecho* service that *bping* relies on.

In ION-DTN, the **ipn:** addressing scheme is used for identifying endpoints in a Delay-Tolerant Network (DTN) using the IPN (Interplanetary Network) URI format standardized by the CCSDS. The format of IPN URIs is **`ipn:<node number>.<service number>`**. In this *bping* test, we are sending a *bping* from **node 1, service 2** to **node 2, service 3**.

Interested in the command parameters for *bping*?
Type `man bping` to bring up the user guide.

Step 9: Stream Bundles (bpdriver/bpcounter)

We use *bpdriver* to show that streaming bundles work. The *bpdriver* tool sends bundles to another node with number, size, and rate selectable (see the man page). The *bpcounter* testing program receives the bundles and calculates the throughput.

We will begin by configuring rpi02 to receive bundles. This step must be completed before running *bpdriver* from the sending node.

rpi02 Terminal Window F **comp2iontest/iontest Directory**	
☐	`bpcounter ipn:2.4 1000`

Node2 (service 4) is now configured to receive 1000 bundles.

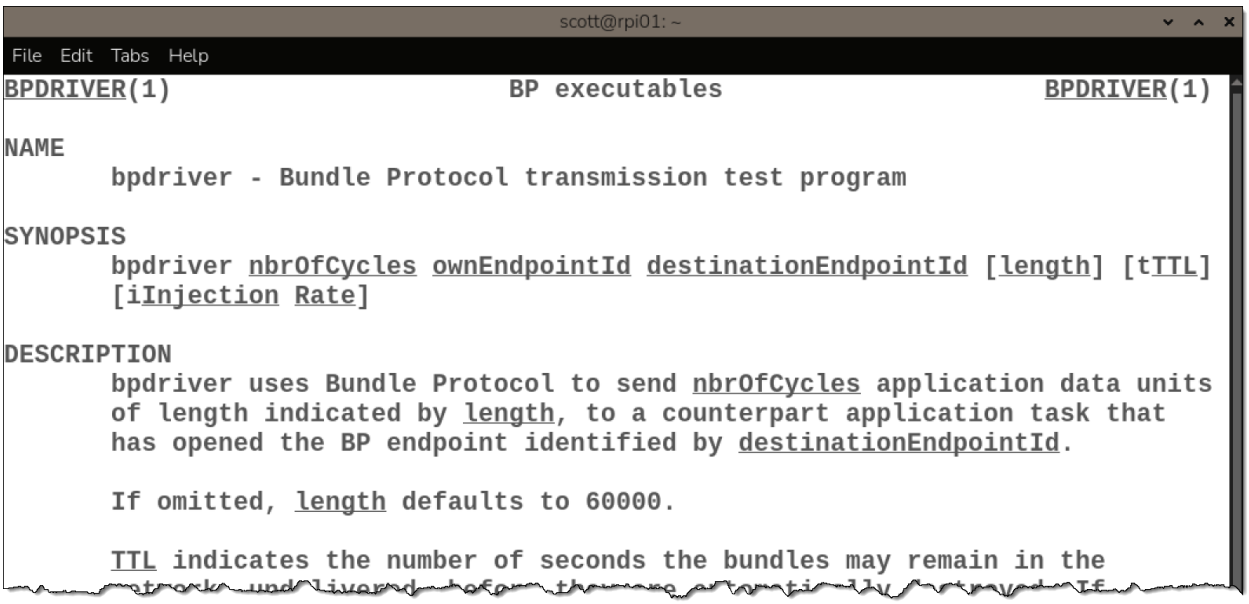
Now let's move to the node1 terminal window to transmit the bundles.

Location: Terminal Window C **comp1iontest/iontest Directory**	
☐	`bpdriver 1000 ipn:1.4 ipn:2.4 -10000`

This test sends 1,000 bundles that are 10,000 bytes each.

We will review the results demonstrated in the lab.

Interested in the command parameters for *bpdriver*?
Type `man bpdriver` to bring up the user guide.



Step 10: Transfer a File (bpcp/bpcpd)

We use *bpcpd* (Bundle Protocol Copy Daemon) on a receiving node and *bpcp* (Bundle Protocol Copy) on a sending node. These tools use *cfdp* (CCSDS File Delivery Protocol) to transfer files.

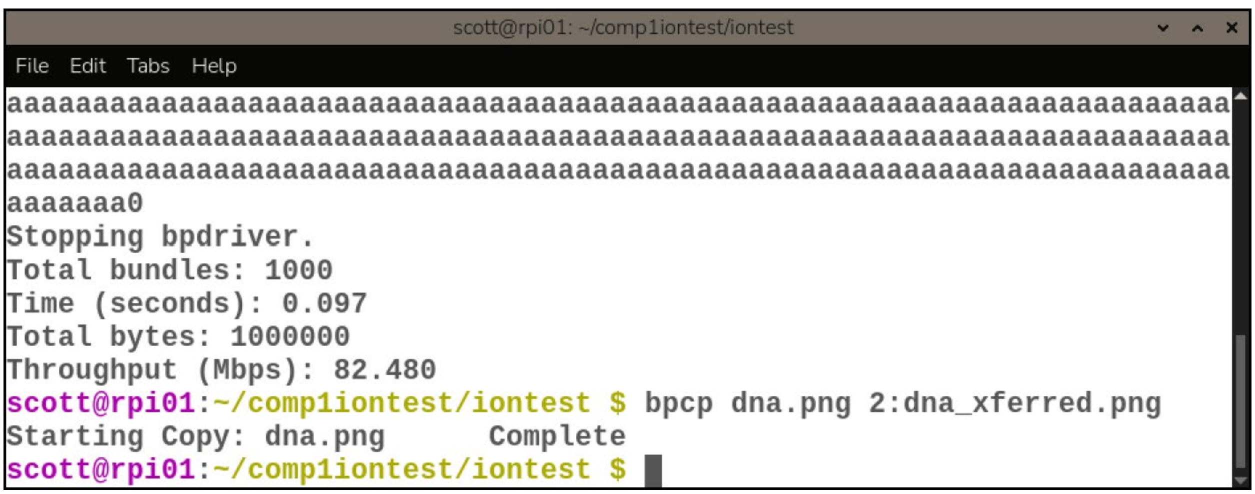
We will begin by configuring rpi02 to receive files.

rpi02 Terminal Window F **comp2iontest/iontest Directory**	
☐	`bpcpd`

Node2 (rpi02) is now configured to receive a file on service 4.

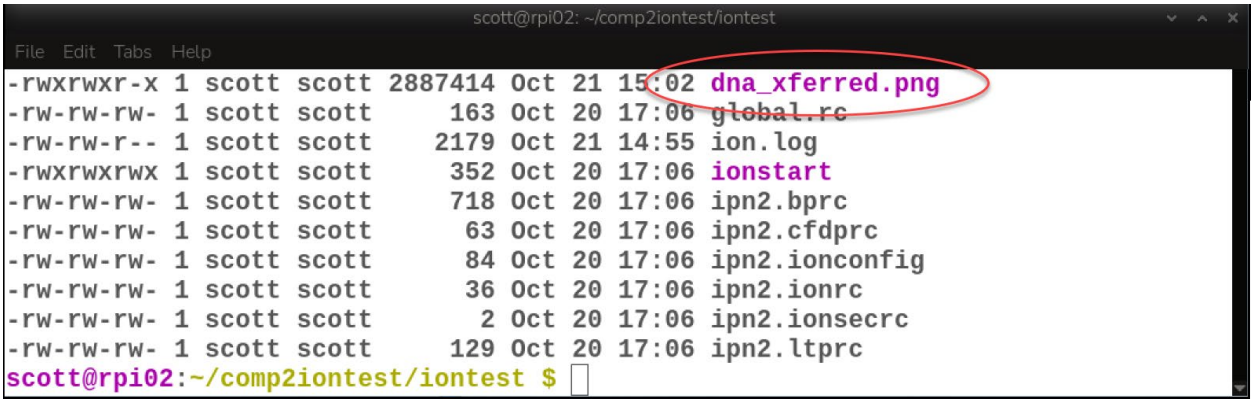
Now let's move to the rpi01 command terminal window to transmit the file.

Location: Terminal Window C **comp1iontest/iontest Directory**	
☐	`bpcp dna.png 2:filename` (replace `filename` with a new name for the file at the target)



Now let's verify the receipt of the file on the target system.

rpi02 Terminal Window F **comp2iontest/iontest Directory**	
	`Ctrl+c` to terminate the *bpcpd* process
☐	`ls -l` Your new file should be located in the *comp2iontest/iontest* directory.



We will review the results demonstrated in the lab.

We have also included a file, *beyond.png*, in your ipn:2 *iontest* folder. Try sending this file to ipn:1. Verify the file transfer worked by examining the folder contents on ipn:1.

Congratulations! You've finished the two-computer Raspberry Pi ION-DTN Lab!

Wrapping Up

Congratulations on completing the first Raspberry Pi ION-DTN Lab!

You've now set up a two-computer ION-DTN configuration, tested connectivity, and sent a bundle streams and files between nodes.

This hands-on lab was designed to provide a foundational understanding of ION-DTN in a dual-node simulation environment. However, your learning journey doesn't stop here. We strongly encourage you to continue exploring, experimenting, and testing with ION-DTN.

Remember to visit the course portal page to review the videos and download additional lab supplements. The course portal page will remain accessible for at least three months after this class.

https://www.chappell-university.com/dtn-wastc1025

Take a moment and review the Appendix for tips on increasing the RPi-5 clock speed, runnind a headless Raspberry Pi, configuring SSH timeouts and retries, and joining/supporting IPNSIG.

For additional information on ION, refer to
https://ion-dtn.readthedocs.io/en/ion-open-source-4.1.3s/ION-Quick-Start-Guide
This lab used "Build ION 4.1.3 (and earlier versions) without actual cipher suite".

Consider joining the ISOC Interplanetary Chapter (which automatically adds you to the IPNSIG group). Your participation supports the mission of IPNSIG to expand interplanetary networking and ensure the Internet truly becomes universal, even beyond Earth.

Questions? Comments? Corrections? Email us at the addresses below.

In addition, please reach out to us if you are interested in bringing ION-DTN labs to your school.

Laura Chappell
Chappell University
laura@chappellu.com

Scott Spicer
Chappell University
scott@chappellu.com

Appendix

Increase the Clock Speed of the RPi-5

The RPi-5 normally runs from 1.5 GHz to 2.4 GHz. To run at max speed (ensure you have a case with a fan and decent heatsink), use the *cpufrequtils* program.

Location: home directory	
☐	`sudo apt install cpufrequtils`
☐	`cpufreq-info` (view information on CPU frequency scaling)
☐	`sudo cpufreq-set -g performance` (force the CPU to run at maximum frequency at all times)
☐	`cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor` (confirm performance is now active on CPU core 0 - should display "performance")
☐	(displays actual frequency in kHz) `cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_cur_freq` (displays actual frequency in kHz - this will show "2400000" as the maximum CPU clock speed)

"Headless" Raspberry Pi

If you wish to do a "headless" installation (without needing a monitor, keyboard, or mouse directly attached to it) on your own, directions may be found at https://www.raspberrypi.com/documentation/computers/getting-started.html). In addition, we have provided step-by-step instructions on your course portal page.

Configure SSH Timeout and Retries

Consider changing the timeout and retry count settings to make SSH connections last longer if they are inactive for a while.

Location: home directory	
☐	`sudo nano /etc/ssh/sshd_config`
☐	Use the down arrow on your keyboard to locate the **Change ClientAliveInterval** and **ClientAliveCountMax** settings.
☐	Uncomment both settings.
☐	Change **ClientAliveInterval** value to 1000000
☐	Write out your changes and close *nano* (**Ctrl-o**, **return**, **Ctrl-x**)
☐	`sudo systemctl reload ssh`

Join IPNSIG (ISOC Interplanetary Chapter)

IPNSIG (Interplanetary Networking Special Interest Group) was founded in 1998 by Vint Cerf and researchers within academia and NASA/JPL. We are now a full Chapter within the Internet Society, known as the Interplanetary Chapter.

You can join us through the ISOC member portal (https://www.internetsociety.org/).

After creating your ISOC membership and logging in, click **Join a Chapter/SIG**, enter *Interplanetary Chapter*, and click **Search**.

Donate to IPNSIG

IPNSIG works to extend terrestrial networking into solar system space, which is consistent with the Internet Society's objectives to grow the Internet to unpopulated areas and connect the unconnected domain—and to ensure that even in space, "The Internet is for Everyone".

IPNSIG, Inc. is an exempt organization as described in Section 501(c)(3) of the US Internal Revenue Code; EIN 87-4782452.

Please consider donating (https://www.ipnsig.org/donate) to help us continue our quest.