

# inginius-push

- Github Workflow to synchronise an INGINious course to a GH repo.
- Bidirectional synchronisation **with no implicit overwriting.**
- Triggered on every main branch push.
- Opens PR with course changes on INGINious.
- Closing the PR resets the remote changes.

## Choose a workflow

Build, test, and deploy your code. Make code reviews, branch management, and issue triaging

Skip this and [set up a workflow yourself](#) →

### Categories

By UCLouvain - INGI

Deployment

Security

Continuous integration

Automation

Pages

Q Search workflows

Found 1 workflow

### Push to INGINious course

By UCLouvain - INGI

Github workflow to push repo changes to an INGINious course

Configure

Extremely useful for any course, use it if you don't.  
If you do already, update to latest version.

# network-trace problem

- Wireshark-like “*fill-in the blanks*” exercises.
- Task is configured with a packet trace and a set of fields to hide.
- Feedback can be provided on a per-field basis.
- The problem outputs all the HTML and JSON needed.

INGInious is flexible enough to be extended for very specific exercises.

There are some errors in your answer. Your score is 0.0%. [Submission #64352b1a607332efce35c2e7]

- **Source Port:** The Source Port field is defined in [Section 3.1 of RFC793](#). It is usually a unique port number that is chosen by the client stack among the unused port numbers. Port numbers that are lower than 1024 play a special role and can only be assigned by processes that have administrator privileges. When the server replies, the Source Port of the request is copied in the Destination Port of the response.

You have 1 wrong answer(s).

## TCP source and destination ports

The first TCP segment was sent by a client to a server to establish a connection. From the response returned by the server, can you infer the source and destination ports used in the first segment ?

#	Length	Summary	Status
0	24 bytes	Transmission Control Protocol, Src Port: ????, Dst Port: ????, Seq: 0, Len: 0	⚠
1	24 bytes	Transmission Control Protocol, Src Port: 1234, Dst Port: 53710, Seq: 0, Ack: 1, Len: 0	

0000 ?? ?? ?? ?? c9 61 3a ec 00 00 00 00 60 02 70 80  
0010 00 00 00 00 02 04 05 a0

0000 ? ? ? ? É a : 1 . . . . . p .  
0010 . . . . .

- Transmission Control Protocol, Src Port: ????, Dst Port: ????, Seq: 0, Len: 0
  - Source Port: ? 1234 Invalid
  - Destination Port: ? 1234 Valid
  - Sequence number: 0 (relative sequence number)
  - Acknowledgment number: 0
  - 0110 .... = Header Length: 24 bytes (6)