**Script Documentation:**

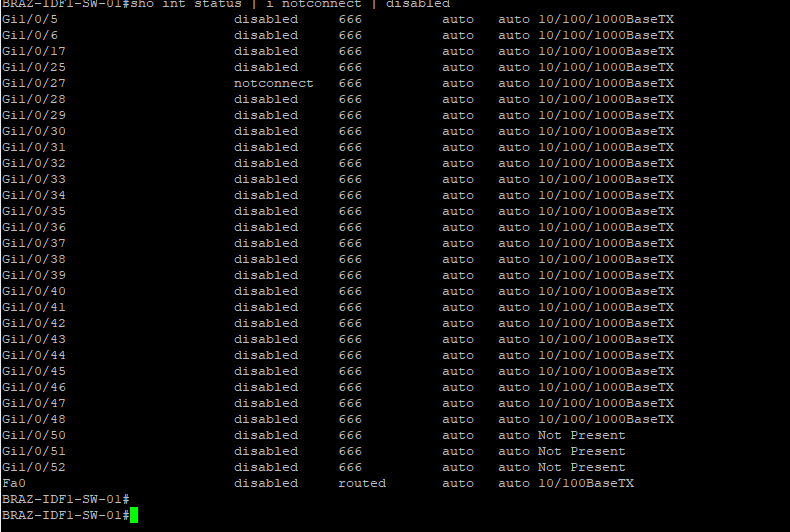
Cody Vollrath

This is a documenting the use case of this script and how to edit it if need be.

An example from the direct terminal is as follows:

This was not executed by the script, this was manually entered with the command “sho int status | i notconnect | disabled”.

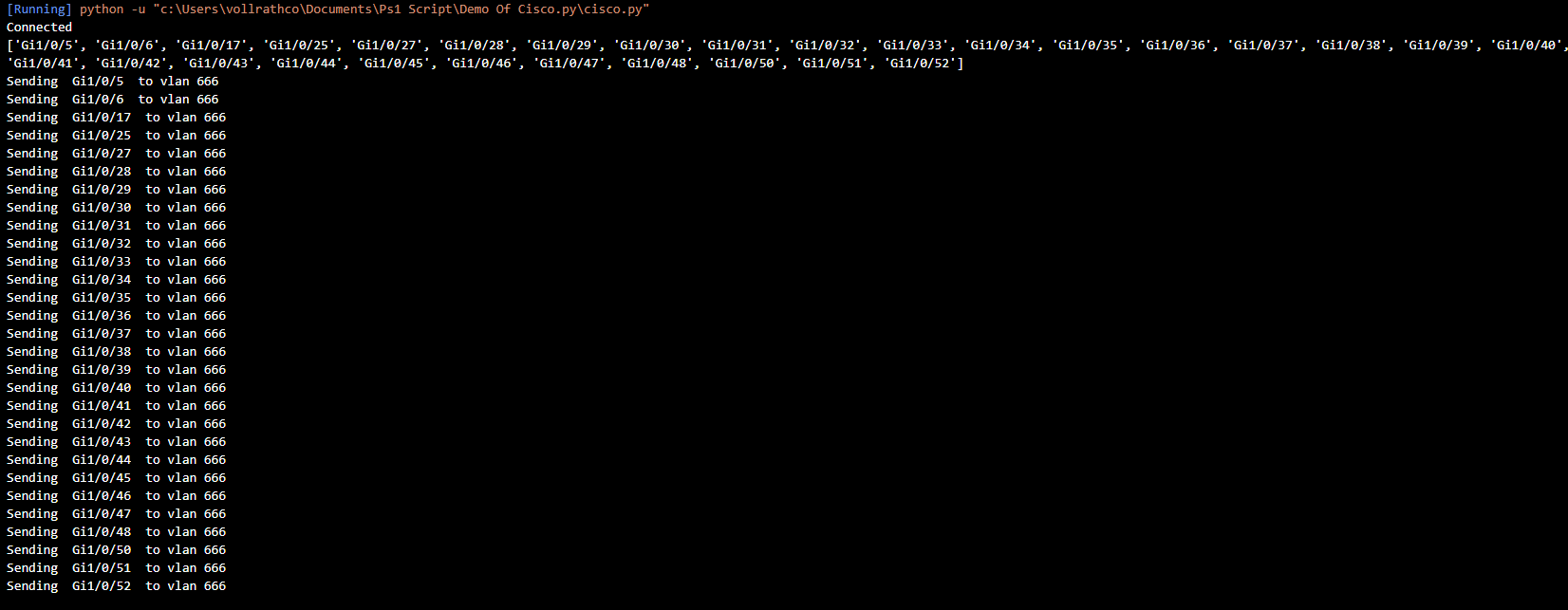
Notice the output shows not just the interface, but the status, as well as the node name below.



This has been remediated in the script, as we only need the interface name, and we only need certain ones too. I implemented a strainer to ensure only interface ids with the ‘/’ character within them would be allowed. As these are the only ID’s we need.

Below is the output from the script:

It is a little complicated looking, but it will display the status of the connection, after that, it will parse through the output from the SSH terminal and determine the interfaces we need based on the bias I implemented into the script. This will vet not only the undesired output, but also ensure the interfaces we want are valid.



**Methods/Functions in the script:**

1. mainMethod(host): This method takes 1 argument represented by a host IP address, and it is where all other methods will run. If you add anything to the script like a new method, please be sure to call that method in this method. The purpose of this method is to facilitate communication with the node and implement all other methods.
2. disable\_Paging(remoteConnect): This method takes 1 argument represented by an SSH connection object. Please see the Paramiko documentation to understand more on that.  
   <http://www.paramiko.org/>  
   The purpose of this method is to disable the paging on the node side, so the output does not display ‘—more--’ and displays the entire output. !important!
3. sho\_IP\_Brief(remoteConnect): This method takes 1 argument represented by an SSH connection object. The purpose of this method is to show only the disabled ports/interfaces and return output as a decoded string. This will allow it to be more readable.
4. getInterfaceNames(remoteConnect): This method takes 1 argument represented by an SSH connection object. The purpose of this method is to strain the output from sho\_IP\_Brief(remoteConnect).
5. convertInterfaceToVlan(listOfInterfaces, remoteConnect): This method takes 2 arguments represented by a list object for interfaces names, and an SSH object for the second one. The purpose of this method is the main point of the script, it will apply the changes we want to apply to the interfaces we want.