

Cisco Switch Backup Script using Netmiko

This script connects to a Cisco switch using SSH or Telnet and backs up the running configuration.

Python Script:

```
from netmiko import ConnectHandler
from datetime import datetime

def backup_cisco_config(ip, username, password, secret='', use_telnet=False):
    device_type = 'cisco_ios_telnet' if use_telnet else 'cisco_ios'

    device = {
        'device_type': device_type,
        'ip': ip,
        'username': username,
        'password': password,
        'secret': secret, # optional, only needed if enable is required
    }

    try:
        print(f'Connecting to {ip} using {"Telnet" if use_telnet else "SSH"}...')
        net_connect = ConnectHandler(**device)

        # Enter enable mode if secret is provided
        if secret:
            net_connect.enable()

        # Get running configuration
        config = net_connect.send_command("show running-config")

        # Save to file
        timestamp = datetime.now().strftime('%Y%m%d-%H%M%S')
        filename = f"backup_{ip.replace('.', '_')}_{timestamp}.txt"
        with open(filename, 'w') as f:
            f.write(config)

        print(f'Backup from {ip} saved to {filename}')
        net_connect.disconnect()
    except Exception as e:
        print(f'Error connecting to {ip}: {e}')


# Test the function
if __name__ == "__main__":
    backup_cisco_config('192.168.1.1', 'admin', 'password', use_telnet=True)
```

```
except Exception as e:  
    print(f"Failed to backup {ip}: {e}")  
  
# Example usage:  
if __name__ == "__main__":  
    # === EDIT THESE ===  
    ip_address = "192.168.1.1"  
    username = "cisco"  
    password = "cisco"  
    enable_secret = "ciscoAuto2025" # Optional  
    use_telnet = False # Set to True to use Telnet  
  
    backup_cisco_config(ip_address, username, password, enable_secret, use_telnet)
```

Mustafa Ahmed Al-Ghurabi