Joshua Ferrara
25 June 2016
Revision 1.1

Dell PowerConnect 2724 Reverse Engineering Docs

Caution

I'd like to mention that this is in no way an official or complete documentation. Things could be wrong though I have documented it to the best of my ability. With the amount of bugs I found in testing, I'd wager there are definitely some quirks I will have missed.

Login and authentication

Obtain a Session ID (SID/SSID)

A session ID (SID/SSID) must first be obtained from the page `/login11.htm` with an HTTP `GET` request. The SID is stored in the attribute `value` of the element with an ID of `Session`. This SID will be used in future requests and keep us authenticated.

Make the login request

We must now make a `POST` request to the `/tgi/login.tgi` page. This post request will be `application/x-www-form-urlencoded` and contain these form values:

- `Username` – a string value containing a username to login with
- `Password` – MD5 encoded string containing `username + password + SID`
    - JavaScript Example: `hex_md5(username + password + SID);`
- `Session` – the SID mentioned above

Possible responses

- HTTP Status Codes
    - `300` – Login success
        - A cookie will be set. This cookie can be used to authenticate any new requests.
    - `200` – Login failed
        - The resulting HTML can be parsed to determine the cause of the failure
            - "Invalid password"
            - "No such user"

Reoccurring Issues

- After logging in many times, you eventually get "locked out" in a sense. The PowerConnect will not allow you to login even if the username/password is correct. Keep in mind that you do not want to keep repeatedly logging in and getting a new SID for each request, else you'll run in to this issue.

Messing around with VLANs

VLANs can be changed around with some simple HTTP `POST` requests to the `/tgi/vlan.tgi` endpoint. This `POST` request will be `application/x-www-form-urlencoded` and contain these form values:

- `op` – "select"
- `vlan` – VLAN to edit
- `ports` – a string with each character representing one port on the switch. 24 ports = 24 digits long. (See Fig. 1)
    - Type `3` is tagged egress packets

  - Type `1` is untagged egress packets
  - Type `0` denotes the port does not belong to the VLAN
- `trunks` – same as above except 6 digits long. (See Fig. 2)

Note: the form values must be submitted in the above order (ensure whatever HTTP library you use to make the requests respects this order). The PowerConnect 2724 does not like it in any other order. My assumption is that on the web server end, they're throwing away the form keys and pushing the form values into a 1D array. If form values aren't submitted in the above order, the server code will try to access the value submitted for a certain array position and will get another value in its place, causing a dropped TCP connection.

Fig 1
Ex: 000000310000011000000000

Port #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Type	0	0	0	0	0	0	3	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
Definition							T	U						T	T									

Key: T = Tagged Egress Packet, U = Untagged Egress Packet, Blank does not belong to VLAN

Fig 2
Ex: 030001

Lag #	1	2	3	4	5	6
Type	0	3	0	0	0	1
Definition		T				U

Key: T = Tagged Egress Packet, U = Untagged Egress Packet, Blank does not belong to VLAN

Possible responses for setting VLANs

- HTTP Status Codes
  - 302 – VLAN has been set
- TCP connection dropped?
  - Make sure you are submitting the POST request with the form values in the correct order.
  - If everything above is correct, sometimes this happens even with a correctly formatted request.