FileWave Anywhere API (v2)

What

Hopefully it is now clear that FileWave Anywhere (formerly called WebAdmin) functions by making API calls to the FileWave server; meaning these same API calls can be used in scripts or by other 3rd party tools.

When/Why

Perhaps to make automated API calls from within Scripts ran on devices or 3rd party tools to GET or POST/PATCH/DELETE entries on the FileWave server.

How

In the main, the queries will look almost the same as the Command Line RESTful commands. Not only are the commands the same, but it is also necessary to use a base64 FileWave Administrator Token. Please view the Working with APIs KB page for more details if required.

Read a current Inventory Query

Imagine it was desirable to read an already existing Inventory Query with an ID of 191. The below shows the Command Line API call, followed by the FileWave Anywhere API call.

Command Line:

```
curl -s  -H "Authorization: $auth" https://$server_dns:20445/inv/api/v1/query_result/191
```

FileWave Anywhere:

```
curl -s  -H "Authorization: $auth" https://$server_dns:/api/inv/api/v1/query_result/191
```

As indicated in an earlier API KB document, FileWave Anywhere uses port 443 (default if not specified) and the URL has an additional '/api' at the beginning of the URL path.

In an example setup, both of these would report exactly the same output, which might look something like:

```
{
  "offset":0,
  "fields":
    [
      "Client_device_name"
  ],
  "values":
    [
      ["C02Z90WDLVDQ"],
      ["FW1063"],
      ["mac0001"],
      ["macadmin1"],
      ["mini"],
      ["ML1015VMNEWNAME"],
      ["ml1063"],
      ["Orac"],
      ["pro"]
  ],
  "filter_results":9,
  "total_results":9,
  "version":3
}
```

Send an MDM Command

Since FileWave Anywhere is a tool with a broader range of control, the API options extend beyond just basic queries. As just one example, the below shows a method to send an MDM command to a device, in this instance a command to restart an Android device, based upon the Device ID:

```
curl -X POST "https://${server}/api/android/restart_devices" \
  -H  "Content-Type: application/json" \
  -H "authorization: ${auth}" \
  -d '{"device_ids": ["3aa9cea3eb7ea992"]}'
```

Although FileWave Anywhere has much wider scope than the Command Line, there are some considerations with this API.

- Unlike the Command Line, it is not possible at this time, to run an Inventory Query with the FileWave Anywhere API, without actually creating an Inventory Query on the FileWave server.
- Access to Custom Fields values for devices at this time, is only accessible via an Internal API path and this method reports all associated Custom Field values for a device.
- Setting Custom Field values for a device using the FileWave Anywhere API does not allow the inclusion of a time stamp

As such, the Command Line API should be favoured for these above options.

Please see the FileWave Anywhere Documentation for a more in-depth look at the possible API options available.

> When referring to devices, the FileWave Anywhere API calls always use device_id. Depending upon the use case, it may also be desirable to consider using the Command Line RESTful API instead, if this value is not readily available.

> Where API calls are made through FileWave as scripts, any inventory value may be supplied as either a Launch Argument or Environment Variable. This could include device_id.

Revision #10
Created 4 July 2023 14:44:11 by Sean Holden
Updated 5 July 2023 19:02:46 by Sean Holden