Return FileWave Device Details as JSON

Get one desktop client's inventory details

The legacy inventory endpoint in this example returns the inventory shown under Client Info > Device Details as JSON. A help desk, inventory integration, or local Fileset script can use the response without scraping FileWave Central.

> This example requests the `DesktopClient` component.
> For another device component, confirm the supported component name and response schema in the Swagger documentation for your FileWave Server version.

> The endpoint requires a Device ID.
> An external integration may need one request to find Device IDs and a second request to retrieve details for the selected desktop client.

> A Fileset running on a client can receive that client's Device ID as a launch argument or environment variable, avoiding the lookup request.

Request the JSON response

Use either the HTTPS reverse-proxy path on port 443 or the direct command-line API path shown below. Set the server and Device ID variables, and supply the FileWave application token through the `FILEWAVE_TOKEN` environment variable before running either request:

```
server_dns="filewave-server.example"
device_id="replace-with-device-id"
```

> Keep the token out of the script and keep TLS verification enabled.
> Use a restricted application token supplied by a secure runtime environment. Do not add `-k` or `--insecure` to these commands.

> The `python3 -m json.tool` pipe formats the JSON for reading. Remove that pipe if Python 3 is unavailable or the calling system needs the compact response.

HTTPS reverse-proxy path on port 443 from macOS or Linux:

```
curl --fail --silent --show-error \
  --header "Authorization: ${FILEWAVE_TOKEN}" \
```

```
  --header "Accept: application/json" \
  "https://${server_dns}/api/inv/api/v1/client/details/${device_id}/DesktopClient" \
  | python3 -m json.tool
```

Direct command-line API path on port 20445 from macOS or Linux:

```
curl --fail --silent --show-error \
  --header "Authorization: ${FILEWAVE_TOKEN}" \
  --header "Accept: application/json" \
  "https://${server_dns}:20445/inv/api/v1/client/details/${device_id}/DesktopClient" \
  | python3 -m json.tool
```

> The reverse-proxy path uses HTTPS on port 443 and adds `/api` before `/inv/api`. The direct example uses port 20445; some environments use port 20443 instead.

A successful request returns a JSON object similar to the sanitized, abbreviated sample below.

> Returned fields vary by device component, FileWave version, inventory state, and configured Custom Fields. Treat this sample as illustrative rather than a fixed response schema.


```
{
    "CustomFields__ldap_username": {
        "status": 0,
        "type": "string",
        "updateTime": "2018-06-21T19:37:23.585851Z",
        "value": "example-user"
    },
    "CustomFields__local_ip_address": {
        "status": 0,
        "type": "string",
        "updateTime": "2018-06-21T19:49:51Z",
        "value": "192.0.2.10"
    },
    "CustomFields__malwarebytes_installed": {
        "status": 0,
        "type": "bool",
        "updateTime": "2018-06-21T19:49:51Z",
        "value": false
    },
    "CustomFields__po_number": {
        "status": 0,
        "type": "string",
        "updateTime": "2018-06-21T19:49:51Z",
        "value": "PO-EXAMPLE-001"
    },
    "CustomFields__property_tag": {
                "status": 0,
        "updateTime": "2018-06-21T19:49:51Z",
```

```
        "type": "string",
        "value": "Example organization"
    },
    "CustomFields__purchase_date": {
        "updateTime": null,
        "value": null
    },
    "CustomFields__school_name": {
        "status": 0,
        "type": "string",
        "updateTime": "2018-06-21T19:49:51Z",
        "value": "Example Elementary School"
    },
    "CustomFields__site_description": {
        "updateTime": null,
        "value": null
    },
    "CustomFields__textedit_version": {
        "status": 0,
        "type": "string",
        "updateTime": "2018-06-21T19:49:51Z",
        "value": "1.13"
    },
    "CustomFields__user_role": {
        "updateTime": null,
        "value": null
    },
    "archived": null,
    "auth_username": "example-user",
    "building": null,
    "cpu_count": 2,
    "cpu_speed": 2759000000,
    "cpu_type": "Intel(R) Core(TM) i5-3470S CPU @ 2.90GHz",
    "current_ip_address": "192.0.2.10",
    "deleted_from_admin": false,
    "department": null,
    "device_id": "0123456789abcdef0123456789abcdef01234567",
    "device_manufacturer": "VMware, Inc.",
    "device_name": "EXAMPLE-MAC-01",
    "device_product_name": "VMware7,1",
    "enroll_date": "2018-06-17T17:11:08.709785Z",
    "enrollment_state": 2,
    "filewave_client_locked": false,
    "filewave_client_name": "EXAMPLE-MAC-01",
    "filewave_client_version": "12.8.1",
    "filewave_id": 123,
    "filewave_model_number": 617,
    "free_disk_space": 56772587520,
    "is_system_integrity_protection_enabled": true,
    "is_tracking_enabled": false,
```

```
    "last_check_in": "2018-06-21T19:54:31.615710Z",
    "last_enterprise_app_validation_date": null,
    "last_ldap_username": null,
    "last_logged_in_username": "example-user",
    "last_state_change_date": "2018-06-21T19:50:09.339609Z",
    "location": null,
    "management_mode": 0,
    "monitor_id": null,
    "operating_system__build": "17B48",
    "operating_system__edition": "Desktop",
    "operating_system__name": "macOS 10.13 High Sierra",
    "operating_system__type": "OSX",
    "operating_system__version": "10.13.1",
    "operating_system__version_major": 10,
    "operating_system__version_minor": 13,
    "operating_system__version_patch": 1,
    "ram_size": 2147483648,
    "rom_bios_version": "VMW71.00V.0.B64.1706210604",
    "security__enrolled_via_dep": null,
    "security__fde_enabled": false,
    "security__firmware_password_change_pending": false,
    "security__firmware_password_exists": false,
    "security__firmware_password_rom_enabled": true,
    "security__hardware_encryption_caps": null,
    "security__passcode_is_compliant": null,
    "security__passcode_is_compliant_with_profiles": null,
    "security__passcode_lock_grace_period": null,
    "security__passcode_lock_grace_period_enforced": null,
    "security__passcode_present": null,
    "security__system_integrity_protection_enabled": true,
    "security__user_approved_enrollment": null,
    "serial_number": "EXAMPLE-SERIAL",
    "state": 0,
    "total_disk_space": 85689589760,
    "unenrolled": false
}
```

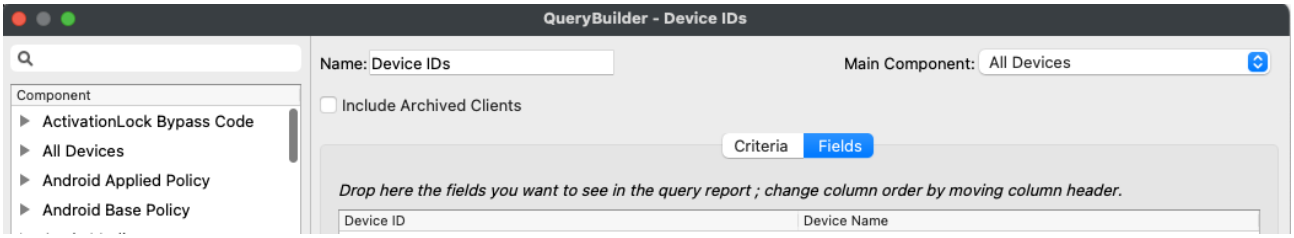

Redirect the formatted response to a file when another process or support workflow needs a saved snapshot:

```
umask 077
curl --fail --silent --show-error \
  --header "Authorization: ${FILEWAVE_TOKEN}" \
  --header "Accept: application/json" \
  "https://${server_dns}/api/inv/api/v1/client/details/${device_id}/DesktopClient" \
  | python3 -m json.tool > /my/path/device_info_${device_id}.json
```

`umask 077` limits a newly created file to the current user. The filename includes the same `$device_id` value; replace `/my/path` with a writable location appropriate for the device and handle the exported inventory according to your organization's data-protection rules.

Obtaining Device IDs

One way to retrieve Device IDs in bulk is to create a FileWave Report (formerly Query) that includes Device ID and any fields needed to identify the target devices. The example below includes Device ID and Device Name.



After saving the Report, use FileWave Anywhere API Documentation to locate its ID. The legacy `query_result` endpoint returns the Report results, including the Device ID column added above. Set `FILEWAVE_TOKEN` securely before running either example; both scripts stop before making a request when the variable is missing.

```
#!/bin/sh
set -e
server="filewave-server.example"
report_id="65"

if [ -z "$FILEWAVE_TOKEN" ]; then
  printf '%s\n' "Set FILEWAVE_TOKEN before running this script." >&2
  exit 1
fi

curl --fail --silent --show-error \
  --header "Authorization: ${FILEWAVE_TOKEN}" \
  --header "Accept: application/json" \
  "https://${server}/api/inv/api/v1/query_result/${report_id}"
```

```
# PowerShell for Windows
$ErrorActionPreference = "Stop"
$server = "filewave-server.example"
$reportId = "65"
if (-not $env:FILEWAVE_TOKEN) {
    throw "Set FILEWAVE_TOKEN before running this script."
}
$headers = @{
    Authorization = $env:FILEWAVE_TOKEN
    Accept = "application/json"
}
```

```
Invoke-RestMethod -Method Get `
    -Headers $headers `
    -Uri "https://$server/api/inv/api/v1/query_result/$reportId"
```

Related articles

- FileWave Anywhere API Documentation
- Command Line API (v1)
- FileWave Anywhere API (v2)
- How to write to a custom field using the FileWave API

Revision #14
★Created 2023-07-04 14:44:11 UTC by Sean Holden
Updated 2026-08-03 14:16:42 UTC by Josh Levitsky