

Inventory Device Name Ipping

Endpoints Summary

Method	Path	Swagger
GET	/inventory_device_name_ipping/{name}/	Swagger ↗

“ The inventory device name ipping endpoint provides functionality to retrieve device information by name, specifically focused on IP ping capabilities and network connectivity status for inventory management systems.

Base URL: <https://control.zequenze.com/api/v1>

Authentication: All endpoints require a Bearer token:

Authorization: Bearer <your-api-token>

Overview

The inventory device name ipping API category is designed for network administrators and IT teams who need to retrieve device information based on device names, with a specific focus on IP ping capabilities and network connectivity monitoring. This endpoint is particularly useful for inventory management systems that need to verify device connectivity status, perform network diagnostics, or retrieve device information for monitoring dashboards.

Common scenarios include automated network health checks, device discovery processes, and integration with monitoring systems that need to correlate device names with their ping status and connectivity information. The endpoint provides essential device data that can be used to determine if a device is reachable on the network and retrieve its current operational status.

This API is typically used in conjunction with other inventory management endpoints to provide a complete picture of device status and network topology within an organization's infrastructure.

Endpoints

GET /inventory_device_name_ipping/{name}/

Description: Retrieves device information and ping status for a specific device identified by its name. This endpoint is essential for network monitoring applications that need to check device connectivity and retrieve basic device information for inventory management purposes.

Use Cases:

- Network health monitoring systems checking device connectivity
- Automated inventory audits verifying device accessibility
- Troubleshooting tools that need to correlate device names with network status
- Dashboard applications displaying real-time device connectivity status

Full URL Example:

```
https://control.zequenze.com/api/v1/inventory_device_name_ipping/server-01/
```

Parameters:

Parameter	Type	In	Required	Description
name	string	path	Yes	The unique name identifier of the device to retrieve ping information for

cURL Example:

```
curl -X GET "https://control.zequenze.com/api/v1/inventory_device_name_ipping/server-01/" \  
-H "Authorization: Bearer YOUR_API_TOKEN" \  
-H "Content-Type: application/json"
```

Example Response:

```
{  
  "id": 1247,  
  "name": "server-01",  
  "ip_address": "192.168.1.100",  
  "ping_status": "active",  
  "last_ping_time": "2024-01-15T14:30:25Z",  
}
```

```
"response_time_ms": 12,  
"device_type": "server",  
"location": "Data Center A",  
"status": "online",  
"uptime_percentage": 99.8  
}
```

Response Codes:

Status	Description
200	Success - Returns device information and ping status
401	Unauthorized - Invalid or missing API token
404	Not Found - Device with specified name does not exist
500	Internal Server Error - Server error occurred while retrieving device data

Common Use Cases

Use Case 1: Network Health Dashboard

Monitor the connectivity status of critical infrastructure devices by querying their ping status and response times. Use this endpoint to populate real-time network health dashboards that display device availability and performance metrics.

Use Case 2: Automated Network Diagnostics

Implement automated scripts that check device connectivity as part of routine network maintenance. The endpoint can be used to identify devices that may be experiencing network issues or are unreachable.

Use Case 3: Inventory Verification

Validate that devices listed in your inventory system are actually accessible on the network. This is particularly useful during asset audits or when onboarding new devices to ensure proper network configuration.

Use Case 4: Troubleshooting Integration

Integrate with helpdesk or ticketing systems to automatically check device connectivity when support tickets are created, providing technicians with immediate network status information.

Best Practices

- **Implement proper error handling** for 404 responses when device names don't exist, as this is common during device lifecycle changes
 - **Cache results appropriately** since ping status information can be used frequently but doesn't need real-time updates for most use cases
 - **Use descriptive device names** that follow your organization's naming conventions to make API calls more intuitive and maintainable
 - **Monitor rate limits** if polling multiple devices frequently, and consider batching requests when possible
 - **Validate device names** before making API calls to ensure they match your naming conventions and reduce unnecessary 404 responses
 - **Log API responses** for network connectivity trends and historical analysis of device availability patterns
-
-

Revision #4

Created 2026-02-04 05:06:15 UTC by ipena@zequenze.com

Updated 2026-02-11 02:55:55 UTC by ipena@zequenze.com