

Asset Integration Guide

Overview

The Asset API allows you to interact with the Argus Asset Management system.



The current Asset API is V1, which is documented on this page.

The [V2 API](#) is currently under development, and should NOT be used for production integrations.

- [Overview](#)
- [Detailed API documentation](#)
- [Host Assets](#)
 - [Fetching a host](#)
 - [Searching for hosts](#)
 - [Creating a host](#)
 - [Updating a host](#)
- [Services](#)
 - [Creating a service](#)
 - [Assign a host to a service](#)
- [Vulnerabilities](#)
 - [Search for vulnerabilities](#)
 - [Mark a vulnerability as resolved](#)
 - [Unresolve a vulnerability](#)

Detailed API documentation



The [Swagger API documentation](#) is always up-to-date and lets you try out any query with your user session or an API-key.

Host Assets

Fetching a host

To fetch a host asset by host ID:

```
curl -X GET -H "Argus-API-Key: my/api/key" -H "Content-Type: application/json" https://api.mnemonic.no/assets/v1/host/3fa85f64-5717-4562-b3fc-2c963f66afa6
```

Searching for hosts

To search for host assets, use the search endpoint. Please see [Swagger](#) for details on all possible search parameters.

The example below searches for all hosts belonging to customerID 1, which have an IP address within the 10.0.5.0/24 CIDR range, and which are member of the specified service (by service ID).

```
curl -X POST -H "Argus-API-Key: my/api/key" -H "Content-Type: application/json" https://api.mnemonic.no/assets/v1/host/search -d '{
  "serviceID": [
    "3fa85f64-5717-4562-b3fc-2c963f66afa6"
  ],
  "ipRange": "10.0.5.0/24",
  "customerID": [ 1 ]
}'
```

Creating a host

To create a new host asset, you need to specify customerID, name (hostname), and description. For server assets, you should also specify any static IP-addresses assigned to the host.

Please see [Swagger](#) for details on all possible fields which may be set on create.

```
curl -X POST -H "Argus-API-Key: my/api/key" -H "Content-Type: application
/json" https://api.mnemonic.no/assets/v1/host -d '{
  "customerID": 1,
  "name": "MyHostName",
  "description": "This is my host",
  "type": "SERVER",
  "ipAddresses": [
    "94.127.56.1"
  ],
  "aliases": [
    "external.host.name"
  ]
}'
```

Updating a host

To update an existing hosts, use the host ID.

Please see [Swagger](#) for details on all possible fields which may be updated.

```
curl -X PUT -H "Argus-API-Key: my/api/key" -H "Content-Type: application
/json" https://api.mnemonic.no/assets/v1/host/3fa85f64-5717-4562-b3fc-
2c963f66afa6 -d '{
  "operatingSystemCPE": "cpe:2.3:o:linux:linux_kernel:2.6.0:*:*:*:*:*:*:*",
  "addIpAddresses": [
    "94.127.56.2"
  ],
  "deleteIpAddresses": [
    "94.127.56.1"
  ]
}'
```

Services

Creating a service

To define a new service, you need to specify customerID, name, and description.

```
curl -X POST -H "Argus-API-Key: my/api/key" -H "Content-Type: application
/json" https://api.mnemonic.no/assets/v1/service -d '{
  "customerID": 1,
  "name": "My Service",
  "description": "This is my important service"
}'
```

Assign a host to a service

To assign a host to a service, use the attachhosts endpoint:

```
curl -X PUT -H "Argus-API-Key: my/api/key" -H "Content-Type: application
/json" https://api.mnemonic.no/assets/v1/service/<serviceid>/attachhosts -
d '{
  "hostAssetIDs": [
    "3fa85f64-5717-4562-b3fc-2c963f66afa6"
  ]
}'
```

Vulnerabilities

Search for vulnerabilities



Searching for vulnerabilities may be slow. To speed up results, please limit the time period, custom

To search for vulnerabilities, use the vulnerability search API. Please see [Swagger](#) for details on all possible search parameters.

To search for unresolved vulnerabilities attached to a specific host:

```
curl -X POST -H "Argus-API-Key: my/api/key" -H "Content-Type: application/json" https://api.mnemonic.no/assets/v1/vulnerability/search -d '{
  "hostID": [ "3fa85f64-5717-4562-b3fc-2c963f66afa6" ],
  "resolutionCode": [ "UNRESOLVED" ],
  "limit":100
}'
```

To search for unresolved vulnerabilities for a specific CVE (within a specific customer):

```
curl -X POST -H "Argus-API-Key: my/api/key" -H "Content-Type: application/json" https://api.mnemonic.no/assets/v1/vulnerability/search -d '{
  "customerID": [ 1 ],
  "vulnerabilityReference": [ "CVE-2020-11898" ],
  "resolutionCode": [ "UNRESOLVED" ],
  "limit":100
}'
```

Mark a vulnerability as resolved

A resolution may be marked as "resolved", with any resolution code of ACCEPTED, FALSE_POSITIVE, SERVICE_NOT_AVAILABLE or NO_LONGER_VULNERABLE:

```
curl -X PUT -H "Argus-API-Key: my/api/key" -H "Content-Type: application/json" https://api.mnemonic.no/assets/v1/vulnerability/3fa85f64-5717-4562-b3fc-2c963f66afa6/resolve -d '{
  "resolution": "ACCEPTED",
  "comment": "Nothing to see here"
}'
```

Unresolve a vulnerability

If a vulnerability must be reopened to state "unresolved", you can use the "reopen" endpoint:

```
curl -X PUT -H "Argus-API-Key: my/api/key" -H "Content-Type: application/json" https://api.mnemonic.no/assets/v1/vulnerability/3fa85f64-5717-4562-b3fc-2c963f66afa6/reopen
```