BeyondInsight and Password Safe API Guide 7.1

Table of Contents

BeyondInsight and Password Safe API Overview

This document specifies the Representational State Transfer (REST) compliant Application Programmer Interface (API) over HTTPS for BeyondInsight and Password Safe. It is a way to integrate a portion of the BeyondInsight and Password Safe functionality into your own applications.

This resource is intended for readers with knowledge of HTTPS request and response processing, web development, and JSON notation.

> For more information about enabling API Access, please see the following:
>
> - BeyondInsight User Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/bi/index.htm
> - Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm

Usage

API Key

The API key is a cryptographically strong random sequence of numbers hashed into a 128-character string. It is encrypted and stored internally using AES 256 encryption. Any language with a Representational State Transfer (REST) compliant interface can access the API with the API key and Run As in the authorization header.

> ***Note:*** *Some environments may still use an old-style API Key, which is a formatted Globally Unique Identifier (GUID). Rotating the API Key will produce the new-style API key described above.*

Session State

Session State is maintained between API calls. The method is dependent on the scripting language. Initiate a session using API POST Auth/SignAppIn and always call POST Auth/Signout when you are done.

Base Endpoint

The following base endpoint will be used throughout this document. **the-server** is a placeholder and should be replaced for your environment. SSL is required to use the Password Safe Public API.

```
<base> = https://the-server/BeyondTrust/api/public/v3
```

> **IMPORTANT!**
>
> *The base endpoint has changed as of v6.2.0. The previous endpoint (https://the-server/eEye.RetinaCS.Server/api/public/v3) will continue to accept API calls, however new scripts should use the new endpoint above and existing scripts should be changed at the earliest opportunity after upgrading to v6.2.0 (or above). This change decouples the API from BeyondInsight and Password Safe, isolating resources and allowing standalone API configuration.*

Authorization Header

Use the web request authorization header to communicate the API application key, the **RunAs** username, and the user password:

- **key:** The API Key configured in BeyondInsight for your application.
- **runas:** The username of a BeyondInsight user that has been granted permission to use the API Key.
- **pwd:** The **RunAs** user password surrounded by square brackets (optional; required only if the **User Password** is required on the Application API Registration).

```
Authorization=PS-Auth key=c479a66f…c9484d; runas=doe-main\johndoe; pwd=[un1qu3];
```

> ***Note:*** *The API Keys in the examples have been shortened for brevity. A domain user is being used.*

Two-Factor Authentication

Depending on how the two-factor server is configured, a programmatic two-factor challenge is sometimes required.

No Challenge

If the two-factor server is configured to authenticate through a push or mobile two-factor challenge, a challenge response is often not required. The first call to POST Auth/SignAppIn should log the user in, as long as the authentication request to the two-factor server does not time out.

Challenge

When a two-factor challenge is configured, two calls to POST Auth/SignAppIn are required and session state must be maintained between these two calls to validate the two-factor challenge.

The initial call to POST Auth/SignAppIn results in a *401 Unauthorized response* which contains a header WWW-Authenticate-2FA containing the prompt from the authentication service. The prompt can be used to prompt the user for the challenge answer.

Note: If this header is not present, a two-factor authentication challenge has not been configured for the user.

When the challenge answer has been received from the user, POST Auth/SignAppIn is called again with the challenge answer in the authorization header, similar to the other authorization parameters:

- **challenge:** The answer to the two-factor challenge.

```
Authorization=PS-Auth key=c479a66f…c9484d; runas=doe-main\johndoe; pwd=[un1qu3]; challenge=543687;
```

> ***Note:** The challenge answer is only required on the second call to POST Auth/SignAppIn and not on subsequent requests.*

Common Response Codes

Below are response codes common to all APIs. Custom responses are detailed in the individual endpoints.

- 200 – Request successful.
- 204 – Request successful. No content in body.
- 400 – Bad Request – Validation failure or missing request body. Reason in response body.
- 401 – Unauthorized – User is not authenticated. Typical reasons include:
    - An invalid product license was detected.
    - The request headers were not set properly.
    - The server could not verify the validity of the request (due to one or more API factors).
    - The user session has expired.
    - The API key has been rotated but has not been updated in the calling script or application.

> ***Tip:** When you encounter a 401 error due to factor validation failure, a User Audit entry will be created in BeyondInsight and an email will be sent to the Administrator detailing the reason. Look here first for the reason why authorization failed.*

- 403: – Access forbidden. User does not have the appropriate role or permission.

***Tip:** A 403 can also occur when SSL trust cannot be established.*

- 404 – Object not found where expected. Reason in response body.
- 500 – Unexpected server error occurred. Please contact the developers.

Examples

***Example:** C#*

Create and reuse a persistent connection using the System.Net.Http.HttpClient class.

```
HttpClient client = new HttpClient();
client.DefaultRequestHeaders.Add("Authorization",
"PS-Auth key= c479a66f…c9484d; runas=doe-main\johndoe;");

string json = Newtonsoft.Json.JsonConvert.SerializeObject(null);
System.Net.Http.StringContent content = new StringContent(json);
content.Headers.ContentType = new System.Net.Http.Headers.MediaTypeHeaderValue("application/json");

HttpResponseMessage signInResponse = client.PostAsync("<base>/Auth/SignAppin", content).Result;
```

Subsequent calls:

```
HttpResponseMessage getResponse = client.GetAsync("<base>/ManagedAccounts").Result;
```

User Password Factor Enabled (header example only)

```
HttpClient client = new HttpClient();
client.DefaultRequestHeaders.Add("Authorization",
"PS-Auth key= c479a66f…c9484d; runas=doe-main\johndoe; pwd=[un1qu3];");
```

***Example:** Powershell*

Powershell internally creates a session variable to use for each subsequent call; Invoke-RestMethod CmdLet options -SessionVariable and -WebSession respectively. In the below example, the variable is named "session" and has script-level scope.

```
$headers = @{ Authorization="PS-Auth key=c479a66f…c9484d; runas=doe-main\johndoe;"; };
$uri = "<base>/Auth/SignAppin";
$signinResult = Invoke-RestMethod -Uri $uri -Method POST -Headers $headers -SessionVariable
script:session;
```

Subsequent calls:

```
$uri = "<base>/ManagedAccounts";
$accounts = Invoke-RestMethod -Uri $uri -Method GET -WebSession $script:session -Headers $headers;
```

Example: *Java*

Create and reuse a persistent connection using the java.net.HttpURLConnection class.

```
URL baseURL = new URL("HTTPS", "the-server", 443, "/BeyondTrust/api/public/v3/");
URL url = new URL(baseURL, "Auth/SignAppIn");
HttpURLConnection conn = (HttpURLConnection)url.openConnection();
conn.setRequestProperty("Authorization","PS-Auth key=c479a66f…c9484d; runas=doe-main\johndoe;");
```

Example: *Ruby*

Using the rest-client gem, carry over the ASP.NET_SessionId header.

```
samp_key = 'PS-Auth key= c479a66f…c9484d; runas=doe-main\johndoe;'
result = RestClient::Request.execute(method: :post, url: '<base>/Auth/SignAppin', :headers =>
{‘Authorization’ => samp_key} )
session_id = result.cookies["ASP.NET_SessionId"]
```

Subsequent calls:

```
result = RestClient::Request.execute(method: :get, url: '<base>/ManagedAccounts', :headers=>
{‘Authorization’ => samp_key, :cookies => {'ASP.NET_SessionId' => session_id}} )
```

Example: *Python*

Create and reuse a persistent connection using the requests module.

```
header = {'Authorization': 'PS-Auth key=c479a66f…c9484d; runas=doe-main\johndoe;'}
session = requests.Session()
session.headers.update(header)
response = session.post('<base>/Auth/SignAppin')
```

Subsequent calls:

```
accounts = session.get('<base>/ManagedAccounts')
```

Example: *Bash*

Using curl, option **–c** will create cookie file and **–b** will use it in subsequent API calls.

```
curl -i -c pbpscookie.txt -X POST https:<base>/Auth/SignAppin -H "Content-Type: application/json" -H
"Authorization: PS-Auth key=c479a66f…c9484d; runas=doe-main\johndoe;" -d ""
```

Subsequent calls:

```
curl -i -b pbpscookie.txt -X GET https:<base>/ManagedAccounts -H "Content-Type: application/json" -H
"Authorization: PS-Auth key=c479a66f…c9484d; runas=doe-main\johndoe;" -d ""
```

Workflow

There are some loose dependencies between the APIs. A typical sequence will be to list accounts or find an account, request a password, retrieve that password (once approved), and then release the password.

Create and Manage an Asset, Create User Group, Assign Roles

Case: Create and manage an asset, create a managed account, create a managed account quick rule, create/provision an LDAP/AD/BeyondInsight User Group, grant **Read** access to new smart rule with Requestor role and access policy.

- `POST <base>/Auth/SignAppin`
- `POST <base>/Workgroups/{ID}/Assets`
- `POST <base>/Assets/{assetId}/ManagedSystems`
- `POST <base>/ManagedSystems/{managedSystemId}/ManagedAccounts`
- `POST <base>/QuickRules`
- `POST <base>/UserGroups`
- `POST <base>/UserGroups/{userGroupId}/SmartRules/{smartRuleId}/Roles`
- `POST <base>/Auth/Signout`

Retrieve a Password

Case: Request, Retrieve, and Checkin a password for a managed account

- `POST <base>/Auth/SignAppin`
- `GET <base>/ManagedAccounts` OR `GET <base>/ManagedAccounts?systemName={systemName}&accountName={accountName}`
- `POST <base>/Requests`
- `GET <base>/Credentials/{requestId}`
- `PUT <base>/Requests/{requestId}/Checkin`
- `POST <base>/Auth/Signout`

Create a Session

Case: Request a Session, Create a Session, and Checkin the request for a managed account

- `POST <base>/Auth/SignAppin`
- `GET <base>/ManagedAccounts` OR `GET <base>/ManagedAccounts?systemName={systemName}&accountName={accountName}`
- `POST <base>/Requests (AccessType="RDP" or AccessType="SSH")`
- `POST <base>/Requests/{requestId}/Sessions (SessionType == Request.AccessType above)`

- `PUT <base>/Requests/{requestId}/Checkin`
- `POST <base>/Auth/Signout`

Retrieve a Password as an ISA

Case: Create an ISA Password Request

- `POST <base>/Auth/SignAppin`
- `GET <base>/ManagedAccounts` OR `GET <base>/ManagedAccounts?systemName={systemName}&accountName={accountName}`
- `POST <base>/ISARequests`
- `POST <base>/Auth/Signout`

Create a Session as an ISA

Case: Create an ISA Session

- `POST <base>/Auth/SignAppin`
- `GET <base>/ManagedAccounts` OR `GET <base>/ManagedAccounts?systemName={systemName}&accountName={accountName}`
- `POST <base>/ISASessions`
- `POST <base>/Auth/Signout`

Authentication

Quick Navigation

- POST Auth/SignAppin
- POST Auth/Signout

POST Auth/SignAppin

Purpose

Authenticates the provided credentials and creates a user session.

Required Permissions

A user group to which the user belongs must be granted access to the API key given in authorization header. Must be running script from a valid source address as configured in API Registration for the given API key.

Request Body

None

Response Body

Content-Type: application/json

```
{
UserId: int,
SID: string,
EmailAddress: string,
UserName: string,
Name: string
}
```

Response Codes

- 200 – Request successful. User model in the response body.
- 403 – Access forbidden. Returned if the Password Safe license is not valid.
- 410 – API version has been disabled.

For more information, please see "Common Response Codes" on page 13.

POST Auth/Signout

Purpose

Terminates the current user session.

Required Permissions

None

Request Body

None

Response Body

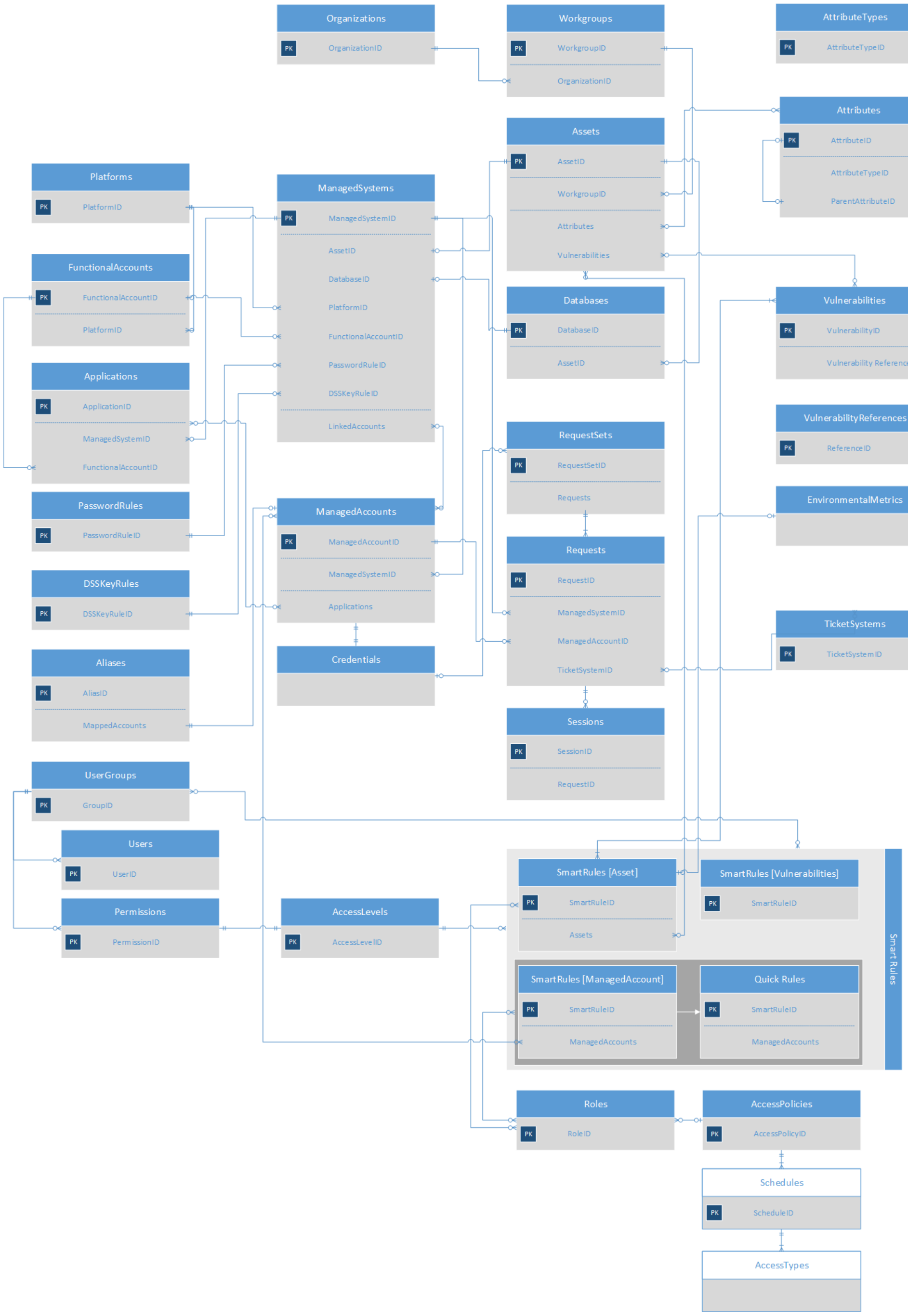
None

Response Codes

- 200 – Request successful.

For more information, please see "Common Response Codes" on page 13.

Entity Relationship Diagram



BeyondInsight APIs

The BeyondInsight APIs require a valid BeyondInsight license and are available to Password Safe-licensed installs.

Access Levels

(i.e. None, Read, Read/Write)

Quick Navigation

-
- POST UserGroups/{userGroupId}/SmartRules/{smartRuleId}/AccessLevels

GET AccessLevels

Purpose

Returns a list of access levels for permissions, for example, **None**, **Read**, and **Read/Write**

Required Permissions

User Accounts Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
AccessLevelID:int,
Name: string,
},
…
]
```

Response Codes

200 - Request successful. Access Levels in the response body.

For more information, please see "Common Response Codes" on page 13.

Get AccessLevels

Purpose

Returns a list of access levels for permissions, for example, **None**, **Read**, and **Read/Write**

POST UserGroups/{userGroupId}/SmartRules/{smartRuleId}/AccessLevels

Purpose

Sets the Access Level for a User Group Smart Rule.

Required Permissions

- User Accounts Management (Read/Write)

URL Parameters

- **userGroupId:** ID of the user group.
- **smartRuleId:** ID of the smart rule.

Request Body

Content-Type: application/json

```
{
AccessLevelID: int
}
```

Response Body

None

Response Codes

- 200 - Request successful.

For more information, please see "Common Response Codes" on page 13.

Address Groups

Quick Navigation

- GET Organizations/{orgID}/addressgroups
- GET Addressgroups
- GET Addressgroups/{addressGroupId}/addresses
- POST Addressgroups/{addressGroupId}
- DELETE Addressgroups/{addressGroupId}
- DELETE Addressgroups/{addressGroupId}/addresses

GET Organizations/{orgID}/addressgroups

Purpose

List the address groups for a given organization.

Required Permissions

- Current user has access to the organization.
- Asset Management (Read)

URL Parameters

orgId: Organization ID.

Request Body

None

Response Body

Content-Type: application/json

```
[
   {
      AddressGroupID: int,
      Name: string,
      OrganizationID: guid // can be null
   }
]
```

Response Codes

200 - Request successful

For more information, please see "Common Response Codes" on page 13.

GET Addressgroups

Purpose

List the address groups.

Required Permissions

- Current user has access to the organization.
- Asset Management (Read)

URL Parameters

None

Request Body

None

Response Body

Content-Type: application/json

```
[
   {
      AddressGroupID: int,
      Name: string,
      OrganizationID: guid // can be null
   {
]
```

Response Codes

200 - Request successful

i For more information, please see "Common Response Codes" on page 13.

GET Addressgroups/{addressGroupId}/addresses

Purpose

List the addresses for an address group.

Required Permissions

- Current user has access to the organization.
- Asset Management (Read)

URL Parameters

addressGroupId: Address Group ID.

Request Body

None

Response Body

Content-Type: application/json

```
[
  {
    AddressID: int,
    AddressGroupID: int,
    Omit: boolean,
    Type: string,
    Value: string,
    LastUpdatedDate: datetime
  }
]
```

Response Codes

200 - Request successful

For more information, please see "Common Response Codes" on page 13.

POST Addressgroups/{addressGroupId}

Purpose

Create an address for the supplied address group.

Required Permissions

- Current user has access to the organization.
- Asset Management (Read/Write)

URL Parameters

addressGroupId: Address Group ID.

Request Body

```
{
 Type: string,
 Value: string,
  Omit: boolean
}
```

Response Body

Content-Type: application/json

```
{
   AddressID: int,
   AddressGroupID: int,
   Omit: boolean,
   Type: string,
   Value: string,
   LastUpdatedDate: datetime
}
```

Response Codes

201 - Request successful. Address in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE Addressgroups/{addressGroupId}

Purpose

Delete the address group and all it's addresses.

Required Permissions

- Current user has access to the organization.
- Asset Management (Read/Write)

URL Parameters

addressGroupId: Address Group ID.

Request Body

None

Response Body

None

Response Codes

200 - Request successful

For more information, please see "Common Response Codes" on page 13.

DELETE Addressgroups/{addressGroupId}/addresses

Purpose

Delete the addresses within the address group.

Required Permissions

- Current user has access to the organization.
- Asset Management (Read/Write)

URL Parameters

addressGroupId: Address Group ID.

Request Body

None

Response Body

None

Response Codes

200 - Request successful

For more information, please see "Common Response Codes" on page 13.

Assets

Quick Navigation

- GET Assets/{id}
- GET Workgroups/{workgroupID}/Assets
- GET Workgroups/{workgroupName}/Assets
- GET Workgroups/{workgroupName}/Assets?name={name}
- POST Workgroups/{workgroupID}/Assets
- POST Workgroups/{workgroupName}/Assets
- PUT Assets/{id}
- POST Assets/Search
- DELETE Assets/{id}

ℹ For more information on related topics, please see:

- Workgroups
- Smart Rules
- Vulnerabilities
- Managed Systems

GET Assets/{id}

Purpose

Returns an Asset by ID.

Required Permissions

Asset Management (Read)

URL Parameters

id: ID of the asset.

Request Body

None

Response Body

Content-Type: application/json

```
{
WorkgroupID: int,
AssetID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
CreateDate: datetime,
LastUpdateDate: datetime
}
```

Response Codes

200 - Request successful. Asset in response body.

i For more information, please see "Common Response Codes" on page 13.

GET Workgroups/{workgroupID}/Assets

Purpose

Returns a list of Assets by Workgroup ID.

Required Permissions

Asset Management (Read)

URL Parameters

workgroupID: ID of the Workgroup.

Query Parameters (optional)

- **limit:** (default: 100000) Number of records to return.
- **offset:** (default: 0) Number of records to skip before returning <limit> records (can be used in conjunction only with limit).

Request Body

None

Response Body (when limit is not given)

Content-Type: application/json

```
[
{
```

```
WorkgroupID: int,
AssetID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
CreateDate: datetime,
LastUpdateDate: datetime
},
…
]
```

Response Body (when limit is given)

Content-Type: application/json

```
{
TotalCount : int,
Data :
    [
    {
    WorkgroupID: int,
    AssetID: int,
    AssetName: string,
    DnsName: string,
    DomainName: string,
    IPAddress: string,
    MacAddress: string,
    AssetType: string,
    OperatingSystem: string,
    CreateDate: datetime,
    LastUpdateDate: datetime
    },
    …
    ]
}
```

Response Codes

200 - Request successful. Assets in response body.

For more information, please see "Common Response Codes" on page 13.

GET Workgroups/{workgroupName}/Assets

Purpose

Returns a list of Assets by Work Group Name.

Required Permissions

Asset Management (Read)

URL Parameters

workgroupName: Name of the Workgroup.

Query Parameters (optional)

- **limit:** (default: 100000) Number of records to return.
- **offset:** (default: 0) Number of records to skip before returning <limit> records (can only be used in conjunction with limit).

Request Body

None

Response Body (when limit is not given)

Content-Type: application/json

```
[
{
WorkgroupID: int,
AssetID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
CreateDate: datetime,
LastUpdateDate: datetime
},
…
]
```

Response Body (when limit is given)

Content-Type: application/json

```
{
TotalCount : int,
Data :
    [
    {
    WorkgroupID: int,
    AssetID: int,
    AssetName: string,
    DnsName: string,
    DomainName: string,
    IPAddress: string,
    MacAddress: string,
    AssetType: string,
    OperatingSystem: string,
    CreateDate: datetime,
    LastUpdateDate: datetime
    },
    …
    ]
}
```

Response Codes

200 - Request successful. Assets in response body.

For more information, please see "Common Response Codes" on page 13.

GET Workgroups/{workgroupName}/Assets?name={name}

Purpose

Returns an Asset by Workgroup Name and Asset Name.

Required Permissions

Asset Management (Read)

URL Parameters

workgroupName: Name of the workgroup.

Query Parameters

name: Name of the asset.

Request Body

None

Response Body

Content-Type: application/json

```
{
WorkgroupID: int,
AssetID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
CreateDate: datetime,
LastUpdateDate: datetime
}
```

Response Codes

200 - Request successful. Asset in response body.

For more information, please see "Common Response Codes" on page 13.

POST Workgroups/{workgroupID}/Assets

Purpose

Creates a new asset in the workgroup, referenced by ID.

Required Permissions

Asset Management (Read/Write)

URL Parameters

workgroupID: ID of the workgroup.

Request Body

Content-Type: application/json

```
{
IPAddress: string,
AssetName: string,
DnsName: string,
DomainName: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string
}
```

Request Body Details

- **IPAddress:** (required) Asset IP address.
- **AssetName:** (optional) Asset name. If not given, a padded IPAddress is used.
- **DnsName:** (optional) Asset DNS name.
- **DomainName:** (optional) Asset domain name.
- **MacAddress:** (optional) Asset MAC address.

- **AssetType:** (optional) Asset type.
- **OperatingSystem:** (optional) Asset operating system.

Response Body

Content-Type: application/json

```
{
WorkgroupID: int,
AssetID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
CreateDate: datetime,
LastUpdateDate: datetime
}
```

Response Codes

201 - Request successful. Asset in response body.

> For more information, please see "Common Response Codes" on page 13.

POST Workgroups/{workgroupName}/Assets

Purpose

Creates a new asset in the workgroup referenced by name.

Required Permissions

Asset Management (Read/Write)

URL Parameters

workgroupName: Name of the Workgroup.

Request Body

Content-Type: application/json

```
{
IPAddress: string,
AssetName: string,
DnsName: string,
DomainName: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string
}
```

Request Body Details

- **IPAddress:** (required) Asset IP address.
- **AssetName:** (optional) Asset name. If not given, a padded IPAddress is used.
- **DnsName:** (optional) Asset DNS name.
- **DomainName:** (optional) Asset domain name.
- **MacAddress:** (optional) Asset MAC address.
- **AssetType:** (optional) Asset type.
- **OperatingSystem:** (optional) Asset operating system.

Response Body

Content-Type: application/json

```
{
WorkgroupID: int,
AssetID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
CreateDate: datetime,
LastUpdateDate: datetime
}
```

Response Codes

201 - Request successful. Asset in response body.

For more information, please see "Common Response Codes" on page 13.

PUT Assets/{id}

Purpose

Updates an existing asset by ID.

***Tip:** Call "GET Assets/{id}" on page 29 (or equivalent) first to get the current state of the Asset before calling PUT Assets/{id} to update it with new values.*

Required Permissions

Asset Management (Read/Write)

URL Parameters

id: ID of the asset.

Request Body

Content-Type: application/json

```
{
WorkgroupID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
}
```

Request Body Details

- **WorkgroupID:** (required) ID of the Workgroup to which the asset belongs.
- **AssetName:** (required) Asset name.
- **DnsName:** (required) Asset DNS name.
- **DomainName:** (required) Asset domain name.
- **IPAddress:** (required) Asset IP address.
- **MacAddress:** (required) Asset MAC address. An empty value is accepted and will clear any existing value.
- **AssetType:** (required) Asset type. An empty value is accepted and will clear any existing value.
- **OperatingSystem:** (required) Asset operating system. An empty value is accepted and will clear any existing value.

Response Body

Content-Type: application/json

```
{
WorkgroupID: int,
AssetID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
CreateDate: datetime,
LastUpdateDate: datetime
}
```

Response Codes

200 - Request successful. Asset in response body.

For more information, please see "Common Response Codes" on page 13.

POST Assets/Search

Purpose

Returns a list of assets that match the given search criteria.

Required Permissions

Asset Management (Read)

Query Parameters (optional)

- **limit:** (default: 100000) Number of records to return.
- **offset:** (default: 0) Number of records to skip before returning <limit> records (can only be used in conjunction with limit).

Request Body

Content-Type: application/json

```
{
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
}
```

Request Body Details

At least one request body property should be provided; any property not provided will be ignored. All search criteria is case insensitive and is an exact match (equality), except for IPAddress.

IPAddress can be a single IP address (10.0.0.1), a comma-delimited list of IPs (10.0.0.1,10.0.0.2,10.0.0.3), an IP range (10.0.0.1-10.0.0.25), or CIDR notation (10.0.0.0/24).

Response Body (when limit is not given)

Content-Type: application/json

```
[
{
WorkgroupID: int,
AssetID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
CreateDate: datetime,
LastUpdateDate: datetime
},
…
]
```

Response Body (when limit is given)

Content-Type: application/json

```
{
TotalCount : int,
Data :
    [
    {
    WorkgroupID: int,
    AssetID: int,
    AssetName: string,
    DnsName: string,
    DomainName: string,
    IPAddress: string,
    MacAddress: string,
    AssetType: string,
    OperatingSystem: string,
    CreateDate: datetime,
    LastUpdateDate: datetime
    },
    …
    ]
}
```

Response Codes

200 - Request successful. Assets in response body.

For more information, please see "Common Response Codes" on page 13.

DELETE Assets/{id}

Purpose

Deletes an Asset by ID.

Required Permissions

Asset Management (Read/Write)

URL Parameters

id: ID of the asset.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

> For more information, please see "Common Response Codes" on page 13.

DELETE Workgroups/{workgroupName}/Assets?name={name}

Purpose

Deletes an Asset by Workgroup Name and Asset Name.

Required Permissions

Asset Management (Read/Write)

URL Parameters

workgroupName: Name of the Workgroup.

Query Parameters

name: Name of the asset.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

For more information, please see "Common Response Codes" on page 13.

Smart Rule Assets

GET SmartRules/{id}/Assets

Purpose

Returns a list of Assets by Smart Rule ID.

Required Permissions

Read access to the smart rule referenced by ID.

URL Parameters

id: ID of the Smart Rule.

Query Parameters (optional)

- **limit:** (default: 100000) Number of records to return.
- **offset:** (default: 0) Number of records to skip before returning <limit> records (can be used only in conjunction with limit).

Request Body

None

Response Body (when limit is not given)

Content-Type: application/json

```
[
{
WorkgroupID: int,
AssetID: int,
AssetName: string,
DnsName: string,
DomainName: string,
IPAddress: string,
MacAddress: string,
AssetType: string,
OperatingSystem: string,
CreateDate: datetime,
LastUpdateDate: datetime
},
…
]
```

Response Body (when limit is given)

Content-Type: application/json

```
{
TotalCount : int,
Data :
    [
    {
    WorkgroupID: int,
    AssetID: int,
    AssetName: string,
    DnsName: string,
    DomainName: string,
    IPAddress: string,
    MacAddress: string,
    AssetType: string,
    OperatingSystem: string,
    CreateDate: datetime,
    LastUpdateDate: datetime
    },
    …
    ]
}
```

Response Codes

200 - Request successful. Assets in response body.

> For more information, please see "Common Response Codes" on page 13.

Attribute Types

Quick Navigation

- "GET AttributeTypes" on page 43
- "GET AttributeTypes/{id}" on page 43
- "POST AttributeTypes" on page 44
- "DELETE AttributeTypes/{id}" on page 45

GET AttributeTypes

Purpose

Returns a list of Attribute Types.

Required Permissions

Attribute Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
AttributeTypeID : int,
Name : string,
IsReadOnly: bool
},
…
]
```

Response Codes

200 – Request successful. Attribute Types in the response body.

For more information, please see "Common Response Codes" on page 13.

GET AttributeTypes/{id}

Purpose

Returns an Attribute Type by ID.

Required Permissions

Attribute Management (Read)

URL Parameters

id: ID of the Attribute Type.

Request Body

None

Response Body

Content-type: application/json

```
{
AttributeTypeID : int,
Name : string,
IsReadOnly: bool
}
```

Response Codes

200 – Request successful. Attribute Type in the response body.

For more information, please see "Common Response Codes" on page 13.

POST AttributeTypes

Purpose

Creates a new Attribute Type.

Required Permissions

Attribute Management (Read/Write)

Request Body

Content-Type: application/json

```
{
Name : string
}
```

Response Body

Content-type: application/json

```
{
AttributeTypeID : int,
Name : string,
IsReadOnly: bool
}
```

Response Codes

201 – Request successful. Attribute Type in the response body.

ℹ For more information, please see "Common Response Codes" on page 13.

DELETE AttributeTypes/{id}

Purpose

Deletes an Attribute Type and all related Attributes by ID.

Required Permissions

Attribute Management (Read/Write)

URL Parameters

id: ID of the Attribute Type.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

ℹ For more information, please see "Common Response Codes" on page 13.

Attributes

Quick Navigation

- GET AttributeTypes/{attributeTypeID}/Attributes
- GET Attributes/{id}
- POST AttributeTypes/{attributeTypeID}/Attributes
- DELETE Attributes/{id}

GET AttributeTypes/{attributeTypeID}/Attributes

Purpose

Returns a list of attribute definitions by Attribute Type.

Required Permissions

Attribute Management (Read)

URL Parameters

attributeTypeID: ID of the Attribute Type.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
AttributeID : int,
AttributeTypeID : int,
ParentAttributeID : int, // can be null
ShortName : string,
LongName : string,
Description : string,
ValueInt : int, // can be null
IsReadOnly: bool,
ChildAttributes :
    [
    {
    AttributeID : int,
    AttributeTypeID : int,
    ParentAttributeID : int,
    ShortName : string,
    LongName : string,
    Description : string,
    ValueInt : int, // can be null
    IsReadOnly: bool,
    },
    …
    ]
},
…
]
```

Response Codes

200 – Request successful. Attributes in the response body.

For more information, please see "Common Response Codes" on page 13.

GET Attributes/{id}

Purpose

Returns an attribute definition by ID.

Required Permissions

Attribute Management (Read)

URL Parameters

id: ID of the Attribute.

Request Body

None

Response Body

Content-Type: application/json

```
{
AttributeID : int,
AttributeTypeID : int,
ParentAttributeID : int, // can be null
ShortName : string,
LongName : string,
Description : string,
ValueInt : int, // can be null
IsReadOnly: bool,
ChildAttributes :
    [
    {
    AttributeID : int,
    AttributeTypeID : int,
    ParentAttributeID : int,
    ShortName : string,
    LongName : string,
    Description : string,
    ValueInt : int, // can be null
    IsReadOnly: bool,
    },
    …
    ]
}
```

Response Codes

200 – Request successful. Attributes in the response body.

For more information, please see "Common Response Codes" on page 13.

POST AttributeTypes/{attributeTypeID}/Attributes

Purpose

Creates a new attribute definition by Attribute Type ID.

Required Permissions

Attribute Management (Read/Write)

URL Parameters

attributeTypeID: ID of the Attribute Type.

Request Body

Content-Type: application/json

```
{
ParentAttributeID : int, // can be null
ShortName : string,
LongName : string,
Description : string,
ValueInt : int // can be null
}
```

Response Body

Content-Type: application/json

```
{
AttributeID : int,
AttributeTypeID : int,
ParentAttributeID : int, // can be null
ShortName : string,
LongName : string,
Description : string,
ValueInt : int, // can be null
IsReadOnly: bool,
ChildAttributes :
    [
    {
    AttributeID : int,
    AttributeTypeID : int,
    ParentAttributeID : int,
    ShortName : string,
    LongName : string,
    Description : string,
    ValueInt : int, // can be null
    IsReadOnly: bool,
    },
    …
    ]
}
```

Response Codes

201 – Request successful. Attributes in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE Attributes/{id}

Purpose

Deletes an attribute definition by ID.

Required Permissions

Attribute Management (Read/Write)

URL Parameters

id: ID of the Attribute.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> For more information, please see "Common Response Codes" on page 13.

Asset Attributes

Quick Navigation

- GET Assets/{assetID}/Attributes
- POST Assets/{assetID}/Attributes/{attributeID}
- DELETE Assets/{assetID}/Attributes
- DELETE Assets/{assetID}/Attributes/{attributeID}

GET Assets/{assetID}/Attributes

Purpose

Returns a list of attributes by Asset ID.

Required Permissions

Asset Management (Read), Attribute Management (Read)

URL Parameters

assetID: ID of the Asset.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
AttributeID : int, AttributeTypeID : int,
ParentAttributeID : int, // can be null
ShortName : string,
LongName : string,
Description : string,
ValueInt : int, // can be null
IsReadOnly: bool
},
…
]
```

Response Codes

200 – Request successful. Attributes associated with the Asset in the response body.

For more information, please see "Common Response Codes" on page 13.

POST Assets/{assetID}/Attributes/{attributeID}

Purpose

Assigns an attribute to an Asset.

Required Permissions

Asset Management (Read/Write), Attribute Management (Read/Write)

URL Parameters

- **assetID:** ID of the asset.
- **attributeID:** ID of the attribute Request Body.

Response Body

Content-Type: application/json

```
{
AttributeID : int, AttributeTypeID : int,
ParentAttributeID : int, // can be null
ShortName : string,
LongName : string,
Description : string,
ValueInt : int, // can be null
IsReadOnly: bool,
}
```

Response Codes

200 – Request successful. Attributes in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE Assets/{assetID}/Attributes

Purpose

Deletes all asset attributes by Asset ID.

Required Permissions

Asset Management (Read/Write), Attribute Management (Read/Write)

URL Parameters

assetID: ID of the Asset.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

For more information, please see "Common Response Codes" on page 13.

DELETE Assets/{assetID}/Attributes/{attributeID}

Purpose

Deletes an asset attribute by Asset ID and Attribute ID.

Required Permissions

- Asset Management (Read/Write)
- Attribute Management (Read/Write)

URL Parameters

assetID: ID of the Asset attributeID and ID of the Attribute.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

For more information, please see "Common Response Codes" on page 13.

Configuration

GET Configuration/Version

Purpose

Returns the current system version.

Request Body

None

Response Body

Content-Type: application/json

```
{
Version : string
}
```

Response Codes

200 – Request successful. Version model in the response body.

For more information, please see "Common Response Codes" on page 13.

Databases

Quick Navigation

- GET Databases
- GET Databases/{id}
- GET Assets/{id}/Databases
- POST Assets/{id}/Databases
- PUT Databases/{id}
- DELETE Databases/{id}

For more information on related topics, please see:

- Assets
- Platforms
- Managed Systems

GET Databases

Purpose

Returns a list of Databases.

Required Permissions

Asset Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
AssetID: int,
DatabaseID : int,
PlatformID : int,
InstanceName : string,
IsDefaultInstance : bool,
Port : int,
Version : string,
Template:string
},
…
]
```

Response Codes

201 – Request successful. Databases in the response body.

For more information, please see "Common Response Codes" on page 13.

GET Databases/{id}

Purpose

Returns a Database by ID.

Required Permissions

Asset Management (Read)

URL Parameters

id: ID of the Database.

Request Body

None

Response Body

Content-Type: application/json

```
{
AssetID:int,
DatabaseID : int,
PlatformID : int,
InstanceName : string,
IsDefaultInstance : bool,
Port : int,
Version : string
}
```

Response Codes

201 – Request successful. Databases in the response body.

For more information, please see "Common Response Codes" on page 13.

GET Assets/{id}/Databases

Purpose

Returns a list of Databases for the given Asset.

Required Permissions

Asset Management (Read)

URL Parameters

id: ID of the Asset.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
AssetID: int,
DatabaseID : int,
PlatformID : int,
InstanceName : string,
IsDefaultInstance : bool,
Port : int,
Version : string,
Template:string
},
…
```

Response Codes

200 – Request successful. Databases in the response body.

> For more information, please see "Common Response Codes" on page 13.

POST Assets/{id}/Databases

Purpose

Creates a new Database in the Asset referenced by ID.

Required Permissions

Asset Management (Read/Write)

URL Parameters

id: ID of the Asset.

Request Body

Content-Type: application/json

```
{
PlatformID : int,
InstanceName : string,
IsDefaultInstance : bool,
Port : int,
Version : string,
Template : string,
}
```

Request Body Details

- **PlatformID:** (required) ID of the Platform
- **InstanceName:** Name of the database instance. Required when IsDefaultInstance is false.
- **IsDefaultInstance:** True if the database instance is the default instance, otherwise false. **Note:** Only Platforms MS SQL Server and MySQL support setting this value to true.
- **Port:** (required) The Database port.
- **Template:** The database connection template.

Response Body

Content-Type: application/json

```
{
AssetID: int,
DatabaseID : int,
PlatformID : int,
InstanceName : string,
IsDefaultInstance : bool,
Port : int,
Version : string,
Template:string
}
```

Response Codes

200 – Request successful. Databases in the response body.

For more information, please see "Common Response Codes" on page 13.

PUT Databases/{id}

Purpose

Updates an existing Database by ID.

Required Permissions

Asset Management (Read/Write)

URL Parameters

id: ID of the Database.

Request Body

Content-Type: application/json

```
{
PlatformID: int,
InstanceName: string,
IsDefaultInstance: bool,
Port: int,
Version: string,
Template: string
}
```

Request Body Details

- **PlatformID:** (required) ID of the Platform.
- **InstanceName:** Name of the database instance. Required when IsDefaultInstance is false.
- **IsDefaultInstance:** True if the database instance is the default instance, otherwise false. Note: Only Platforms MS SQL Server and MySQL support setting this value totrue.
- **Port:** The Database port.
- **Template:**The database connection template.

Response Body

Content-Type: application/json

```
{
AssetID: int,
DatabaseID: int,
PlatformID: int,
InstanceName: string,
IsDefaultInstance: bool,Port: int,
Version: string,
Template: string
}
```

Response Codes

200 – Request successful. Databases in the response body.

> ℹ For more information, please see "Common Response Codes" on page 13.

DELETE Databases/{id}

Purpose

Deletes a Database by ID.

Required Permissions

Asset Management (Read/Write)

URL Parameters

id: ID of the Database.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> ℹ For more information, please see "Common Response Codes" on page 13.

Environmental Metrics

GET SmartRules/{id}/EnvironmentalMetrics

Purpose

Returns a list of Environmental Metrics by Smart Rule ID. Applies to all Assets in a Smart Rule.

Required Permissions

Read access to the Smart Rule referenced by ID

URL Parameters

id: ID of the Smart Rule.

Request Body

None

Response Body

Content-Type: application/json

```
{
Availability : string,
Confidentiality : string,
Integrity : string,
CollateralDamagePotential : string,
TargetDistribution : string
}
```

Response Codes

- 200 - Request successful. Environmental Metrics in the response body.
- 204 - Request successful. No Environmental Metrics were found for the Smart Rule.

For more information, please see "Common Response Codes" on page 13.

Imports

POST Imports

Purpose

Queues a third-party import.

Required Permissions

Scan Management (Read/Write)

Request Body

Content-Type: application/json

```
{
WorkgroupID: int,
ImportType: string,
Filter: string,
FileName: string,
FileContents: byte[],
Base64FileContents: string
}
```

Note: *Provide either* ***FileContents*** *or* ***Base64FileContents****.*

Request Body Details

- **WorkgroupID:** ID of the Workgroup to import the Assets into
- **ImportType:** (case-senstitive, default: PASSWORDSAFE) Type of Import being queued
- **PASSWORDSAFE:** Password Safe import file. Expected file extension: .xml
- **RETINARTD:** Retina© RTD file. Expected file extension: .rtd
- **NESSUS:** Nessus© import file. Expected file extension: .csv
- **NESSUSSECCEN:** NessusSecurityCenter© import file. Expected file extension: .csv
- **NEXPOSE:** Nexpose© import file. Expected file extension: .csv or .xml
- **QUALYSGUARD:** QualysGuard© file. Expected file extension: .csv or .xml
- **METASPLOIT:** METASPLOIT© import file. Expected file extension: .xml
- **MCAFEEVM:** McAfee Vulnerability Management© import file. Expected file extension: .csv
- **TRIPWIRE:** Tripwire© import file. Expected file extension: .csv
- **Filter:** (default: All Assets) Asset selection filter:
    - All Assets: No filter, import all
    - Single IPv4 address (i.e. 10.0.0.1)
    - IPv4 range (i.e. 10.0.0.1-10.0.0.5)
    - CIDR (i.e. 10.0.0.0/24).

- **FileName:** Filename (including extension) of the import file. One of the following is required:
  - **FileContents:** The array containing the content of the import file.
  - **Base64FileContents:** Base64 string containing the content of the import file.

> For accepted file types, please see "ImportType: (case-senstitive, default: PASSWORDSAFE) Type of Import being queued" on page 62.

Response Body

Content-Type: application/json

```
{
ImportID: int
}
```

Response Codes

200 – Request successful. Import ID in the response body.

> For more information, please see "Common Response Codes" on page 13.

Operating Systems

For more information on related topics, please see Vulnerability Export Report.

GET OperatingSystems

Purpose

Returns a list of Operating Systems.

Required Permissions

Asset Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
OperatingSystemID : int,
Name : string
},
…
]
```

Response Codes

200 – Request successful. Operating Systems in the response body.

For more information, please see "Common Response Codes" on page 13.

Organizations

Quick Navigation

- GET Organizations
- GET Organizations/{id}
- GET Organizations?name={name}

GET Organizations

Purpose

Returns a list of Organizations to which the current user has permission.

Required Permissions

Asset Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
OrganizationID : string,
Name : string,
IsActive : bool
},
…
]
```

Response Codes

200 – Request successful. Organizations in the response body.

For more information, please see "Common Response Codes" on page 13.

GET Organizations/{id}

Purpose

Returns an Organization by ID.

Required Permissions

- Current user has permission to the Organization
- Asset Management (Read)

URL Parameters

id: ID of the Organization.

Request Body

None

Response Body

Content-Type: application/json

```
{
OrganizationID : string,
Name : string,
IsActive : bool
}
```

Response Codes

200 – Request successful. Organizations in the response body.

> i For more information, please see "Common Response Codes" on page 13.

GET Organizations?name={name}

Purpose

Returns an Organization by name.

Required Permissions

- Current user has permission to the Organization
- Asset Management (Read)

Query Parameters

name: Name of the Organization.

Request Body

None

Response Body

Content-Type: application/json

```
{
OrganizationID : string,
Name : string,
IsActive : bool
}
```

Response Codes

200 – Request successful. Organizations in the response body.

> For more information, please see "Common Response Codes" on page 13.

Permissions

(i.e. Asset Management, User Accounts Management, Scan Management, etc.)

Quick Navigation

- GET Permissions
- User Group Permissions
- GET UserGroups/{userGroupID}/Permissions
- POST UserGroups/{userGroupId}/Permissions
- DELETE UserGroups/{userGroupId}/Permissions

GET Permissions

Purpose

Returns a list of Permissions.

Required Permissions

User Accounts Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
PermissionID : int,
Name : string
},
…
]
```

Response Codes

200 – Request successful. Permissions in the response body.

For more information, please see "Common Response Codes" on page 13.

User Group Permissions

GET UserGroups/{userGroupID}/Permissions

Purpose

Gets all Permissions for the User Group referenced by ID.

Required Permissions

User Accounts Management (Read)

URL Parameters

userGroupId: ID of the User Group

Request Body

None

Response Body

Content-Type: application/json

```
[{
PermissionID : int,
AccessLevelID : int
},
…]
```

Response Codes

200 – Request successful. Permissions in the response body.

For more information, please see "Common Response Codes" on page 13.

POST UserGroups/{userGroupId}/Permissions

Purpose

Sets Permissions for the User Group referenced by ID.

Required Permissions

User Accounts Management (Read/Write)

URL Parameters

userGroupId: ID of the User Group.

Request Body

Content-Type: application/json

```
[
{
PermissionID : int,
AccessLevelID : int
},
…
]
```

Response Body

None

Response Codes

204 – Request successful. No content in body.

For more information, please see "Common Response Codes" on page 13.

DELETE UserGroups/{userGroupId}/Permissions

Purpose

Deletes all Permissions for the User Group referenced by ID.

Required Permissions

User Accounts Management (Read/Write)

URL Parameters

userGroupId: ID of the User Group.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

For more information, please see "Common Response Codes" on page 13.

Smart Rules

Quick Navigation

- GET SmartRules
- GET SmartRules/{id}
- GET SmartRules?title={title}
- GET Organizations/{orgID}/SmartRules?title={title}
- POST SmartRules/FilterAssetAttribute
- POST SmartRules/{id}/Process
- DELETE SmartRules/{id}
- DELETE SmartRules?title={title}
- DELETE Organizations/{orgID}/SmartRules?title={title}

For more information on related topics, please see:

- Quick Rules
- Asset-based Smart Rules
- "GET SmartRules/{id}/Assets" on page 41
- Managed Account-based Smart Rules
- "GET SmartRules/{smartRuleID}/ManagedAccounts" on page 209
- "GET QuickRules/{quickRuleID}/ManagedAccounts" on page 198
- Managed System-based Smart Rules
- GET SmartRules/{smartRuleID}/ManagedSystems
- Vulnerability-based Smart Rules
- Vulnerabilities
- Environmental Metrics

GET SmartRules

Purpose

Returns a list of Smart Rules to which the current user has at least Read access.

Query Parameters

type: (optional, default: all) Type of Smart Rules to return (all, ManagedAccount, ManagedSystem, Asset, Vulnerabilities).

Request Body

None

Response Body

Content-Type: application/json

```
[
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool,
RuleType: string
},
...
]
```

Response Codes

200 – Request successful. Smart Rule in the response body.

For more information, please see "Common Response Codes" on page 13.

GET SmartRules/{id}

Purpose

Returns a Smart Rule by ID.

Required Permissions

Read access to the Smart Rule referenced by ID

URL Parameters

id: ID of the Smart Rule.

Request Body

None

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool,
RuleType: string
}
```

Response Codes

200 – Request successful. Smart Rule in the response body.

> For more information, please see "Common Response Codes" on page 13.

GET SmartRules?title={title}

Purpose

Returns a Smart Rule by title.

In a multi-tenant environment, assumes Global Organization.

Required permissions

Read access to the Smart Rule referenced by title.

Query Parameters

title: Title of the Smart Rule.

Request Body

None

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool,
RuleType: string
}
```

Response Codes

200 – Request successful. Smart Rule in the response body.

For more information, please see "Common Response Codes" on page 13.

GET Organizations/{orgID}/SmartRules?title={title}

Purpose

Returns a Smart Rule by Organization ID and title. This is only valid in a multi-tenant environment.

Required Permissions

Read access to the Smart Rule referenced by Organization and title

URL Parameters

orgID: ID of the Organization.

Query Parameters

title: Title of the Smart Rule

Request Body

None

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool,
RuleType: string
}
```

Response Codes

200 – Request successful. Smart Rule in the response body.

For more information, please see "Common Response Codes" on page 13.

POST SmartRules/FilterAssetAttribute

Purpose

Specialized action for creating an Asset-type Smart Rule for filtering Assets by Attributes.

Required Permissions

Asset Management (Read/Write)

Request Body

Content-Type: application/json

```
{
AttributeIDs: [ int, …],
Title: string,
Category: string,
Description: string,
ProcessImmediately: bool
}
```

Request Body Details

- **AttributeIDs:** (required) A list of Attribute IDs to filter by. All the Attributes must be of the same Attribute Type.
- **Title:** (required) The title/name of the new Smart Rule. Must be unique across all Smart Rules.
- **Category:** (required) The category in which to place the Smart Rule.
- **Description:** (optional, default: <value of Title>) The Smart Rule description.
- **ProcessImmediately:** (optional, default: true) True to process the Smart Rule immediately, otherwise false to defer processing to the background Smart Rule processor.

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool
}
```

Response Codes

201 - Request successful. Smart Rulein response body.

For more information, please see "Common Response Codes" on page 13.

POST SmartRules/{id}/Process

Purpose

Immediately process a Smart Rule by ID.

Required Permissions

Read/Write access to the Smart Rule

URL Parameters

ID: ID of the Smart Rule

Request Body

None

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool,
RuleType: string
}
```

Response Codes

- 200 – Request successful. Smart Rule in the response body.
- 409 – Conflict: the Smart Rule is currently processing.

For more information, please see "Common Response Codes" on page 13.

DELETE SmartRules/{id}

Purpose

Deletes a Smart Rule by ID.

Required Permissions

Read/Write access to the Smart Rule referenced by ID

URL Parameters

ID: ID of the Smart Rule.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> For more information, please see "Common Response Codes" on page 13.

DELETE SmartRules?title={title}

Purpose

Deletes a Smart Rule by title.

In a mult-tenant environment, assumes Global Organization.

Required Permissions

Read/Write access to the Smart Rule referenced by title

Query Parameters

title: Title of the Smart Rule.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> For more information, please see "Common Response Codes" on page 13.

DELETE Organizations/{orgID}/SmartRules?title={title}

Purpose

Deletes a Smart Rule by Organization ID and title.

Only valid in a multi-tenant environment.

Required permissions

Read/Write access to the Smart Rule referenced by Organization and title

URL Parameters

orgID: ID of the Organization.

Query Parameters

title: Title of the Smart Rule.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

For more information, please see "Common Response Codes" on page 13.

User Groups

Quick Navigation

- GET UserGroups
- GET UserGroups/{id}
- GET UserGroups?name={name}
- POST UserGroups
- DELETE UserGroups/{id}
- DELETE UserGroups?name={name}

GET UserGroups

Purpose

Returns a list of active and inactive User Groups.

Required Permissions

User Accounts Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
GroupID : int,
Name : string,
DistinguishedName : string,
GroupType : string,
AccountAttribute : string,
MembershipAttribute : string,
IsActive : bool
},
…
]
```

Response Codes

200 – Request successful. User Group in the response body.

For more information, please see "Common Response Codes" on page 13.

GET UserGroups/{id}

Purpose

Returns a User Group by ID.

Required Permissions

User Accounts Management (Read)

URL Parameters

id: ID of the User Group.

Request Body

None

Response Body

Content-Type: application/json

```
{
GroupID : int,
Name : string,
DistinguishedName : string,
GroupType : string,
AccountAttribute : string,
MembershipAttribute : string,
IsActive : bool
}
```

Response Codes

200 – Request successful. User Group in the response body.

> For more information, please see "Common Response Codes" on page 13.

GET UserGroups?name={name}

Purpose

Returns a User Group by name.

Required Permissions

User Accounts Management (Read)

Query Parameters

name: Name of the User Group.

Request Body

None

Response Body

Content-Type: application/json

```
{
GroupID : int,
Name : string,
DistinguishedName : string,
GroupType : string,
AccountAttribute : string,
MembershipAttribute : string,
IsActive : bool
}
```

Response Codes

200 – Request successful. User Group in the response body.

> For more information, please see "Common Response Codes" on page 13.

POST UserGroups

Purpose

Creates a new User Group with Permissions, Smart Rule access, and optionally Application Registration IDs.

Required Permissions

User Accounts Management (Read/Write)

Request Body

The request body differs for the different group types available: **BeyondInsight**, **ActiveDirectory**, **LdapDirectory**.

BeyondInsight Group Type

Request Body

Content-Type: application/json

```
{
groupType : string = "BeyondInsight",
groupName : string,
description : string,
isActive : bool,
Permissions : [ { PermissionID: int, AccessLevelID: int }, ... ],
SmartRuleAccess : [ { SmartRuleID: int, AccessLevelID: int }, ... ],
ApplicationRegistrationIDs: [ int, … ]
}
```

Request Body Details

- **groupName:** (required) Name of the BeyondInsight User Group.
- **description:** (required) Description of the User Group.

For more information, please see "Common Request Body Details" on page 84.

ActiveDirectory Group Type

Request Body

Content-Type: application/json

```
{
groupType : string = "ActiveDirectory",
groupName : string,
forestName : string,
domainName : string,
description : string,
bindUser : string,
bindPassword : string,
useSSL : bool,

isActive : bool,
Permissions : [ { PermissionID: int,
AccessLevelID: int }, ... ],
SmartRuleAccess : [ { SmartRuleID: int, AccessLevelID: int }, ... ],
ApplicationRegistrationIDs: [ int, … ]
}
```

Request Body Details

- **groupName:** (required) Name of the Active Directory group.
- **description:** (required) Description of the User Group.
- **domainName:** (required) The directory domain name.
- **bindUser:** Username for directory binding. If not given, attempts to use existing credentials for the directory. If specifying an existing credential, you also need **Credential Management – Read**. If specifying a new credential, you also need **Credential Management – Read/Write**.

    - **bindPassword:** Password for directory binding (required if bindUser is given).
    - **forestName:** The directory forest name (required when bindUser is given).
- **useSSL:** (default: false) Flag indicating whether to use SSL.

> For more information, please see "Common Request Body Details" on page 84.

LdapDirectory Group Type

Request Body

Content-Type: application/json

```
{
groupType : string = "LdapDirectory",
groupName : string,
groupDistinguishedName : string,
hostName : string,
bindUser : string,
bindPassword : string,
port : int,
useSSL : bool,
membershipAttribute : string,
accountAttribute : string,

isActive : bool,
Permissions : [ { PermissionID: int,
AccessLevelID: int }, ... ],
SmartRuleAccess : [ { SmartRuleID: int, AccessLevelID: int }, ... ],
ApplicationRegistrationIDs: [ int, … ]
}
```

Request Body Details

- **groupDistinguishedName:** (required) Distinguished name of the LDAP group.
- **groupName:** (required) Name of the LDAP group.
- **hostName:** (required) The directory server host name or IP.
- **bindUser:** Username for directory binding. If not given, attempts to use existing credentials for the directory. If specifying an existing credential, you also need **Credential Management – Read**. If specifying a new credential, you also need **Credential Management – Read/Write**.
    - **bindPassword:** Password for directory binding (Note: required if bindUser is given).
    - **port:** Directory server port (valid range: 1 to 65535) (required if bindUser is given).
    - **useSSL:** (default: false) Flag indicating whether to use SSL (required if bindUser is given).
- **membershipAttribute:** (required) Directory group membership attribute.
- **accountAttribute:** (required) Directory account naming attribute.

For more information, please see "Common Request Body Details" on page 84.

Common Request Body Details

- **isActive:** (default: true) True if the group should be created as active, otherwise false.
- **Permissions:** One or more Permissions and Access Levels to set for the new User Group.
- **SmartRuleAccess:** One or more Smart Rules and Access Levels to set for the new User Group.
- **ApplicationRegistrationIDs:** Zero or more IDs representing the API Application Registrations to grant the new user Group. If given, enables API for the User Group.

Response Body

Content-Type: application/json

```
{
GroupID : int, Name : string,
DistinguishedName : string,
GroupType : string,
AccountAttribute : string,
MembershipAttribute : string,
IsActive : bool
}
```

Response Codes

201 – Request successful. User Group in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE UserGroups/{id}

Purpose

Deletes a User Group by ID.

Required Permissions

User Accounts Management (Read/Write)

URL Parameters

id: ID of the User Group.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> For more information, please see "Common Response Codes" on page 13.

DELETE UserGroups?name={name}

Purpose

Deletes a User Group by name.

Required Permissions

User Accounts Management (Read/Write)

Query Parameters

name: Name of the User Group.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> For more information, please see "Common Response Codes" on page 13.

User Group Memberships

Quick Navigation

- GET Users/{userID}/UserGroups
- POST Users/{userID}/UserGroups/{userGroupID}
- DELETE Users/{userID}/UserGroups/{userGroupID}

GET Users/{userID}/UserGroups

Purpose

Returns the User Group Memberships for an existing User.

Required Permissions

User Accounts Management (Read)

URL Parameters

userID: ID of the User.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
GroupID : int,
Name : string,
DistinguishedName : string,
GroupType : string,
AccountAttribute : string,
MembershipAttribute : string,
IsActive : bool
},
…
]
```

Response Codes

200 – Request successful. User Group in the response body.

For more information, please see "Common Response Codes" on page 13.

POST Users/{userID}/UserGroups/{userGroupID}

Purpose

Adds an existing User to a User Group.

Required Permissions

User Accounts Management (Read/Write)

URL Parameters

- **userID:** ID of the User.
- **userGroupID:** ID of the User Group.

Request Body

None

Response Body

Content-Type: application/json

```
{
GroupID : int, Name : string,
DistinguishedName : string,
GroupType : string,
AccountAttribute : string,
MembershipAttribute : string,
IsActive : bool
}
```

Response Codes

201 – Request successful. User Group in the response body.

> For more information, please see "Common Response Codes" on page 13.

DELETE Users/{userID}/UserGroups/{userGroupID}

Purpose

Removes a User from a User Group.

Required Permissions

User Accounts Management (Read/Write)

URL Parameters

- **userID:** ID of the User.
- **userGroupID:** ID of the User Group.

Request Body

None

Response Codes

200 – Request successful.

For more information, please see "Common Response Codes" on page 13.

Users

Quick Navigation

- GET Users
- GET UserGroups/{userGroupId}/Users
- GET Users/{id}
- POST Users
- POST Users/{id}/Quarantine
- POST UserGroups/{userGroupId}/Users
- PUT Users/{id}
- DELETE Users/{id}

GET Users

Purpose

Returns a list of all users if username parameter is not supplied. Otherwise returns the requested user.

> ***Note:*** *Some usernames may be in the format hostname\username, if not represented by an email address.*

Required Permissions

User Accounts Management (Read)

Query Parameters

username: (optional) The username of the user to return.

> ***Note:*** *Use of the optional* ***username*** *parameter will result in the supplied value being recorded in the web server logfile.*

Request Body

None

Response Body

Content-Type: application/json

```
[
{
UserID : int,
UserName : string,
DomainName : string,
DistinguishedName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
IsQuarantined: bool
},
…
]
```

Response Codes

200 – Request successful. Users in the response body.

For more information, please see "Common Response Codes" on page 13.

GET UserGroups/{userGroupId}/Users

Purpose

Returns a list of Users for the User Group referenced by ID.

Required Permissions

User Accounts Management (Read)

URL Parameters

userGroupId: ID of the User Group.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
UserID : int,
UserName : string,
DomainName : string,
DistinguishedName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
IsQuarantined: bool
},
…
]
```

Response Codes

200 – Request successful. Users in the response body.

For more information, please see "Common Response Codes" on page 13.

GET Users/{id}

Purpose

Returns a User by ID.

Required Permissions

User Accounts Management (Read)

URL Parameters

id: ID of the User.

Request Body

None

Response Body

Content-Type: application/json

```
{
UserID : int,
UserName : string,
DomainName : string,
DistinguishedName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
IsQuarantined: bool
}
```

Response Codes

200 – Request successful. User in the response body.

For more information, please see "Common Response Codes" on page 13.

POST Users

Purpose

Creates a new User with no User Group associations.

Required Permissions

User Accounts Management (Read/Write)

Request Body

The request body differs for the different user types available: **BeyondInsight**, **ActiveDirectory**, **LdapDirectory**

BeyondInsight User Type

Content-Type: application/json

```
{
UserType : string = "BeyondInsight",
UserName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
Password : string
}
```

Request Body Details

- **UserName:** (required) Username of the User account.
- **FirstName:** (required) First name of the user.
- **LastName:** (optional) Last name of the user.
- **EmailAddress:** (required must be a properly formatted address) - Email address for the user.
- **Password:** (required) The password they would use to login to BeyondInsight.

ActiveDirectory User Type

Content-Type: application/json

```
{
UserType : string = "ActiveDirectory",
UserName : string,
ForestName : string,
DomainName : string,
BindUser : string,
BindPassword : string,
UseSSL : bool,
}
```

Request Body Details

- **UserName:** (required) Name of the Active Directory user.
- **DomainName:** (required) The directory domain name.
- **BindUser:** Username for directory binding. If not given, attempts to use existing credentials for the directory.
    - **BindPassword:** Password for directory binding (required when BindUser is given).
    - **ForestName:** The directory forest name (required when BindUser is given).
- **UseSSL:** (default: false) Flag indicating whether to use SSL.

For more information, please see"Common Request Body Details" on page 84.

LdapDirectory User Type

Content-Type: application/json

```
{
UserType: string = "LdapDirectory",
HostName: string,
DistinguishedName: string,
AccountNameAttribute: string,
BindUser: string,
BindPassword: string,
Port: int,
UseSSL: bool
}
```

Request Body Details

- **HostName:** (required) The directory server host name or IP.
- **DistinguishedName:** (required) The DistinguishedName of the user to create.
- **AccountNameAttribute:** (required) The Ldap attribute to use for creating the username.
- **BindUser:** Username for directory binding. If not given, attempts to use existing credentials for the directory.
    - **BindPassword:** Password for directory binding. (required if BindUser is given).
    - **Port:** The directory server port. (used when BindUser and BindPassword are given).
    - **UseSSL:** Flag indicating whether to use SSL (used when BindUser and BindPassword are given).

Response Body

Content-Type: application/json

```
{
UserID : int,
UserName : string,
DomainName : string,
DistinguishedName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
IsQuarantined: bool
}
```

Response Codes

200 – Request successful. User in the response body.

For more information, please see "Common Response Codes" on page 13.

POST Users/{id}/Quarantine

Purpose

Quarantines the User referenced by ID.

Required Permissions

Password Safe API Global Quarantine (Read/Write)

URL Parameters

id: ID of the BeyondInsight User.

Request Body

None

Response Body

Content- Type: application/json

```
{
UserID : int,
UserName : string,
DomainName : string,
DistinguishedName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
IsQuarantined: bool
}
```

Response Codes

200 – Request successful. User in the response body.

For more information, please see "Common Response Codes" on page 13.

POST UserGroups/{userGroupId}/Users

Purpose

Creates a User in a BeyondInsight-type User Group.

Required Permissions

User Accounts Management (Read/Write)

URL Parameters

userGroupId: ID of the User Group.

Request Body

Content-Type: application/json

```
{
UserName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
Password : string
}
```

Request Body Details

- **UserName:** (required) Username of the User account.
- **FirstName:** (required) First name of the user.
- **LastName:** (optional) Last name of the user
- **EmailAddress:** (required and must be a properly formatted address) Email address for the user.
- **Password:** (required) The password they would use to login to BeyondInsight.

Response Body

Content-Type: application/json

```
{
UserID : int,
UserName : string,
DomainName : string,
DistinguishedName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
IsQuarantined: bool
}
```

Response Codes

201 – Request successful. User in the response body.

For more information, please see "Common Response Codes" on page 13.

PUT Users/{id}

Purpose

Updates a BeyondInsight User by ID.

***Note:** Cannot update ActiveDirectory or LDAP users.*

Required Permissions

User Accounts Management (Read/Write)

URL Parameters

id: ID of the BeyondInsight User.

Request Body

Content-Type application/json

```
{
UserName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
Password: string
}
```

Request Body Details

- **UserName:** (required) Username of the User account.
- **FirstName:** (required) First name of the user.
- **LastName:** (optional) Last name of the user.
- **EmailAddress:** (required and must be a properly formatted address) Email address for the user.
- **Password:** (optional) The password they would use to log in to BeyondInsight. If given, replaces the current password.

Response Body

Content- Type: application/json

```
{
UserID : int,
UserName : string,
DomainName : string,
DistinguishedName : string,
FirstName : string,
LastName : string,
EmailAddress : string,
IsQuarantined: bool
}
```

Response Codes

200 – Request successful. User in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE Users/{id}

Purpose

Deletes a BeyondInsight User by ID.

***Note:** Cannot delete ActiveDirectory or LDAP users.*

Required Permissions

User Accounts Management (Read/Write)

URL Parameters

id: ID of the BeyondInsight User.

Request Body

None

Response Codes

200 – Request successful.

For more information, please see "Common Response Codes" on page 13.

Vulnerabilities

For more information on related topics, please see:

- Assets
- Vulnerability References

GET Assets/{id}/Vulnerabilities?smartRuleID={srID}

Purpose

Returns a list of Vulnerabilities by Asset ID, optionally including Temporal Metrics related to the Asset and Smart Rule referenced by ID.

Required Permissions

- Asset Management (Read)
- If smartRuleID is given, Read access to the Smart Rule referenced by ID.

URL Parameters

id: ID of the Asset.

Query Parameters

- **smartRuleID:** (optional) ID of an Asset-based Smart Rule.
- **delta:** (optional) An option for returning vulnerability deltas instead of the entire list for an Asset.
    - **lastScan:** Returns vulnerabilities detected during the last completed scan.
        - i.e.: delta=lastScan
    - **<date time>:** Returns vulnerabilities detected on or after the given date and time
        - format: YYYY-MM-DD 24HH:MI:SS
        - i.e.: delta=2017-01-31 23:59:59
- **includeReferences:** (optional, default: false) true to include Vulnerability References as part of the response body, otherwise false. Supplying true can replace a separate call to API GET Vulnerabilities/{id}/VulnerabilityReferences.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
VulnerabilityID : string,
Name : string,
Description : string,
Severity : int,
Solution : string,
BaseScore : decimal,
BaseVector : string,
LastDiscoveryDate : DateTime,
CreatedDate : DateTime,
UpdatedDate : DateTime,
Port : int, // can be null
Protocol : string,
IsExploitable : bool,
IsVulnerable : bool,
TemporalScore : decimal, // null if smartRuleID is not given
TemporalVector : string, // null if smartRuleID is not given
Events :
    [
    {
    TestedValue : string,
    FoundValue : string,
    Context : string
    }
    ],
    …
]
},
References :
    [
    {
    ReferenceID : string,
    Name : string,
    Type : string,
    URL : string
    }
    ],
    …
]
},
…
]
```

***Note:** The API response will return NULL for the **BaseScore** if the vulnerability does not have a CVSS score.*

Response Body Details

References: Always present but only populated if query parameter include **References=true**.

Response Codes

200 – Request successful. Vulnerabilities in the response body.

For more information, please see "Common Response Codes" on page 13.

Vulnerabilities Export Report

For more information on related topics, please see:

- Operating Systems
- Workgroups

POST Vulnerabilities/ExportReport

Purpose

Returns a Vulnerability export report.

Required Permissions

- Asset Management (Read)
- Read access to the Smart Rule referenced by ID.

Request Body

Content-Type: application/json

```
{
SmartRuleID : int,
DateRange : string,
WorkgroupIDs : [int, int, ...],
OperatingSystemID : int
}
```

Valid **DateRange**

```
<?xml version="1.0" encoding="utf-8" ?>
<root>
<!-- ============================================
NOTES:
-->
<!-- ============================================ -->
<!-- Valid date ranges for the /BeyondTrust/api/public/v3/Vulnerabilities/ExportReport endpoint -->
<!-- ============================================ -->
<Test>
<dateRange>all</dateRange>
</Test>
<Test>
<dateRange>today</dateRange>
</Test>
<Test>
<dateRange>yesterday</dateRange>
</Test>
<Test>
```

```
<dateRange>this_week</dateRange>
</Test>
<Test>
<dateRange>last_week</dateRange>
</Test>
<Test>
<dateRange>this_month</dateRange>
</Test>
<Test>
<dateRange>last_month</dateRange>
</Test>
<Test>
<dateRange>this_quarter</dateRange>
</Test>
<Test>
<dateRange>last_quarter</dateRange>
</Test>
<Test>
<dateRange>last_7days</dateRange>
</Test>
<Test>
<dateRange>last_30days</dateRange>
</Test>
<Test>
<dateRange>last_60days</dateRange>
</Test>
<Test>
<dateRange>last_90days</dateRange>
</Test>
<Test>
<dateRange>last_2week</dateRange>
</Test>
<Test>
<dateRange>last_2month</dateRange>
</Test>
<Test>
<dateRange>last_3month</dateRange>
</Test>
<Test>
<dateRange>last_365days</dateRange>
```

Response Body

Content-Type: application/json

```
[
{
AssetId : int,
AuditId : int,
Criticality : string,
CVEName : string,
CVSSScore : string,
CWEName : string,
DateOfFinding : datetime,
DNSName : string,
FixInformation : string,
FoundValue : string,
IPAddress : string,
MACAddress : string,
NetBIOSName : string,
OperatingSystems : string,
PCISeverityV2 : string,
ScanDetail : string,
ScanInformation : string,
Status : string
TestedValue : string,
VulnerabilityAge : string,
VulnerabilityDescription : string,
VulnerabilityExclusion : bool,
VulnerabilityExclusionNote : string,
VulnerabilityName : string,
VulnerabilitySeverity : string,
},
…
]
```

Response Codes

200 – Request successful. Vulnerability Export Report in the response body.

For more information, please see "Common Response Codes" on page 13.

Vulnerability References

Vulnerability Reference types include CVE, CPE, BID, MSSB, LINK, SevCode, etc.

GET Vulnerabilities/{id}/VulnerabilityReferences

Purpose

Returns a list of Vulnerability References by Vulnerability ID.

Required Permissions

Asset Management (Read)

URL Parameters

id: ID of the Vulnerability.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
ReferenceID : string,
Name : string,
Type : string,
URL : string
},
…
]
```

Response Codes

200 – Request successful. Vulnerability References in the response body.

For more information, please see "Common Response Codes" on page 13.

Workgroups

Quick Navigation

- GET Workgroups
- GET Workgroups/{id}
- GET Workgroups?name={name}
- POST Workgroups

GET Workgroups

Purpose

Returns a list of Workgroups to which the current user has permission.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
OrganizationID : string, ID : int,
Name : string
},
…
]
```

Response Codes

200 – Request successful. Workgroups in the response body.

For more information, please see "Common Response Codes" on page 13.

GET Workgroups/{id}

Purpose

Returns a Workgroup by ID.

Required Permissions

- Current user has permission to the Workgroup Organization.
- Asset Management (Read) or Scan Management (Read/Write)

Query Parameters

id: ID of the Workgroup.

Request Body

None

Response Body

Content-Type: application/json

```
{
OrganizationID : string,
ID : int,
Name : string
}
```

Response Codes

200 – Request successful. Workgroups in the response body.

> For more information, please see "Common Response Codes" on page 13.

GET Workgroups?name={name}

Purpose

Returns a Workgroup by name.

Required Permissions

Current user has permission to the Workgroup Organization. Asset Management (Read) or Scan Management (Read/Write)

Query Parameters

name: Name of the Workgroup.

Request Body

None

Response Body

Content-Type: application/json

```
{
OrganizationID : string,
ID : int,
Name : string
}
```

Response Codes

200 – Request successful. Workgroups in the response body.

For more information, please see "Common Response Codes" on page 13.

POST Workgroups

Purpose

Creates a Workgroup.

Required Permissions

Asset Management (Read/Write)

Request Body

Content-Type: application/json

```
{
OrganizationID: string,
Name : string
}
```

Request Body Details

- **Organization ID:** (optional) The ID of the organization in which to place the new Workgroup. If empty, the Workgroup is placed in the default organization.
- **Name:** The name of the Workgroup.

Response Body

Content-Type: application/json

```
{
OrganizationID : string,
ID : int,
Name : string
}
```

Response Codes

201 – Request successful. Workgroups in the response body.

> For more information, please see "Common Response Codes" on page 13.

Deprecated

The content in this section of the guide has been deprecated and is compatible with earlier versions only.

Quick Navigation

- "Deprecated" on page 109
- "Deprecated" on page 109
- "[deprecated] GET UserGroups/{name}" on page 112
- "[deprecated] DELETE UserGroups/{name}" on page 112
- "[deprecated] GET Workgroups/{name}" on page 113

Imports

[deprecated] POST Imports/QueueImportFile

> ***Note:** This API has been deprecated and is available for backwards compatibility only. Use **POST Imports with Base64FileContents** instead.*

Purpose

Queues a Password Safe XML import using multi-part form-data content.

Required Permissions

Scan Management (Read/Write)

Request Body

Content-Type: multipart/form-data

```
{
Content-type: application/json
    {
    WorkgroupID: int,
    FileName: string
    }
application/octet-stream
    {
    <string-encoded byte array representing the file>
    }
}
```

Request Body Details

- **WorkgroupID:** ID of the Workgroup to import the Assets into
- **FileName:** Filename (including extension) of the import file

Response Body

Content-Type: application/json

```
{
ImportID: int
}
```

Response Codes

- 200 – Request successful. Import ID in the response body.
- 400 – The import file was not found in the body of the request, or a request body validation error has occurred.

Smart Rules

[deprecated] POST SmartRules/FilterSingleAccount

***Note:** This API has been deprecated and is available for backwards compatibility only. Use **QuickRules** instead.*

Purpose

Specialized action for creating a Managed Account-type Smart Rule for filtering a single Managed Account by System Name and Account Name.

Required Permissions

Smart Rule Management - Managed Account (Read/Write)

Request Body

Content-type: application/json

```
{
AccountID: int,
Title: string
}
```

Request Body Details

- **AccountID:** (required) ID of the Managed Account you want to filter by parent System Name and Account Name.
- **Title:** (optional) The title/name of the new Smart Rule. If omitted, a unique title is auto-generated.

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool,
RuleType: string
}
```

Response Codes

201 – Request successful. Smart Rule in the response body.

User Groups

[deprecated] GET UserGroups/{name}

> ***Note:*** *This API has been deprecated and is available for backwards compatibility only. Use* ***GET UserGroups?name={name}*** *instead.*

Purpose

Returns a User Group by name.

Required Permissions

User Accounts Management (Read)

URL Parameters

name: Name of the User Group.

Request Body

None

Response Body

Content-Type: application/json

```
{
GroupID : int, Name : string,
DistinguishedName : string,
GroupType : string,
AccountAttribute : string,
MembershipAttribute : string,
IsActive : bool
}
```

Response Codes

200 – Request successful. User Group in the response body.

[deprecated] DELETE UserGroups/{name}

> ***Note:*** *This API has been deprecated and is available for backwards compatibility only****. Use DELETE UserGroups?name={name}*** *instead.*

Purpose

Deletes a User Group by name.

Required Permissions

User Accounts Management (Read/Write)

URL Parameters

name: Name of the User Group.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

Workgroups

[deprecated] GET Workgroups/{name}

> ***Note:*** *This API has been deprecated and is available for backwards compatibility only. Use* ***GET Workgroups?name={name}*** *instead.*

Purpose

Returns a Workgroup by name.

Required Permissions

Current user has permission to the Workgroup Organization. Asset Management (Read) or Scan Management (Read/Write)

Query Parameters

name: Name of the Workgroup.

Request Body

None

Response Body

Content-Type: application/json

```
{
OrganizationID : string,
ID : int,
Name : string
}
```

Response Codes

200 – Request successful. Workgroups in the response body.

Password Safe APIs

The Password Safe APIs require a valid Password Safe license.

Access Policies

Quick Navigation

- GET AccessPolicies
- POST AccessPolicies/Test

GET AccessPolicies

Purpose

Returns a list of Password Safe Access Policies.

Required Permissions

Password Safe Role Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
AccessPolicyID:int,
Name:string,
Description:string,
Schedules :
    [
    {
    ScheduleID : int,
    AccessTypes :
        [
        {
        AccessType : string,
        IsSession : bool,
        RecordSession : bool,
        MinApprovers : int,
        MaxConcurrent : int
        },
        …
        ]
    },
    …
    ]
},
…
]
```

Response Codes

200 - Request successful. Access Policies in response body.

For more information, please see "Common Response Codes" on page 13.

POST AccessPolicies/Test

Purpose

Tests access to a Managed Account and returns a list of Password Safe Access Policies that are available in the request window.

Required Roles

Requestor Role

Request Body

Content-Type: application/json

```
{
SystemId: int,
AccountId: int,
DurationMinutes : int
}
```

Response Body

Content-Type: application/json

```
[
{
AccessPolicyID:int,
Name:string,
Description:string,
Schedules :
    [
    {
    ScheduleID : int,
    AccessTypes :
        [
        {
        AccessType : string,
        IsSession : bool,
        RecordSession : bool,
        MinApprovers : int,
        MaxConcurrent : int
        },
        …
        ]
    },
    …
    ]
},
…
]
```

Response Codes

- 200 - Request successful. Access Policies in response body.
- 403 - User does not have permissions to request the indicated account or the account does not have API access enabled. Response body contains a status code indicating the reason for this forbidden access:
    - 4031 - User does not have permission to request the account or the account is not valid for the system.

For more information, please see "Common Response Codes" on page 13.

Aliases

Quick Navigation

- GET Aliases
- GET Aliases/{id}
- GET Aliases?name={name}

GET Aliases

Purpose

Returns a list of requestable Managed Account Aliases.

Required Roles

Requestor or Requestor/Approver role for the preferred Managed Account referenced by the Alias.

Query Parameters

- **state** (optional, default: 1, 2): Zero or more state values, i.e '**state=2**', '**state=1,2**', '**state=0,1,2**'.
  - 0 - Unmapped
  - 1 - Mapped
  - 2 - Highly Available

Note: *Only Aliases with a mapped* ***state*** *of* ***1*** *or* ***2*** *can be used for API '****POST Aliases/{id}/Requests****'.*

Request Body

None

Response Body

Content-Type: application/json

```
[
{
AliasId: int,
AliasName: string,
AliasState: int,
SystemId: int,
SystemName: string,
AccountId: int,
AccountName: string,
DomainName: string,
InstanceName: string,
DefaultReleaseDuration: int,
MaximumReleaseDuration: int,
LastChangeDate: datetime,
NextChangeDate: datetime,
IsChanging: bool,
ChangeState: int,

MappedAccounts :
    [
    {
    AliasID: int,
    ManagedSystemID: int,
    ManagedAccountID: int,
    Status: string
    },
    …
    ]
}
…
]
```

Response Codes

200 - Request successful. Aliases in response body.

For more information, please see "Common Response Codes" on page 13.

GET Aliases/{id}

Purpose

Returns a requestable Managed Account Alias by ID.

Required Roles

Requestor or Requestor/Approver role for the preferred Managed Account referenced by the Alias.

URL Parameters

id: ID of the Managed Account Alias.

Request Body

None

Response Body

Content-Type: application/json

```
{
AliasId: int,
AliasName: string,
AliasState: int,
SystemId: int,
SystemName: string,
AccountId: int,
AccountName: string,
DomainName: string,
InstanceName: string,
DefaultReleaseDuration: int,
MaximumReleaseDuration: int,
LastChangeDate: datetime,
NextChangeDate: datetime,
IsChanging: bool,
ChangeState: int,

MappedAccounts :
    [
    {
    AliasID: int,
    ManagedSystemID: int,
    ManagedAccountID: int,
    Status: string
    },
    …
    ]
}
```

Response Codes

200 - Request successful. Alias in response body.

For more information, please see "Common Response Codes" on page 13.

GET Aliases?name={name}

Purpose

Returns a requestable Managed Account Alias by name.

Required Roles

Requestor or Requestor/Approver role for the preferred Managed Account referenced by the Alias.

URL Parameters

name: Name of the Managed Account Alias.

Request Body

None

Response Body

Content-Type: application/json

```
{
AliasId: int,
AliasName: string,
AliasState: int,
SystemId: int,
SystemName: string,
AccountId: int,
AccountName: string,
DomainName: string,
InstanceName: string,
DefaultReleaseDuration: int,
MaximumReleaseDuration: int,
LastChangeDate: datetime,
NextChangeDate: datetime,
IsChanging: bool,
ChangeState: int,

MappedAccounts :
    [
    {
    AliasID: int,
    ManagedSystemID: int,
    ManagedAccountID: int,
    Status: string
    },
    …
    ]
}
```

Response Codes

200 - Request successful. Alias in response body.

> For more information, please see "Common Response Codes" on page 13.

Applications

Quick Navigation

- GET Applications
- GET Applications/{id}

GET Applications

Purpose

Returns a list of Applications.

Required Permissions

Password Safe Account Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
ApplicationID : int,
Name : string,
DisplayName : string,
Version : string,
Command : string,
Parameters : string,
Publisher : string,
ApplicationType : string,
FunctionalAccountID : int, // can be null
ManagedSystemID : int, // can be null
IsActive : bool
}
…
]
```

Response Codes

200 - Request successful. Applications in response body.

For more information, please see "Common Response Codes" on page 13.

GET Applications/{id}

Purpose

Returns an Application by ID.

Required Permissions

Password Safe Account Management (Read)

URL Parameters

id: ID of the Application.

Request Body

None

Response Body

Content-Type: application/json

```
{
ApplicationID : int,
Name : string,
DisplayName : string,
Version : string,
Command : string,
Parameters : string,
Publisher : string,
ApplicationType : string,
FunctionalAccountID : int, // can be null
ManagedSystemID : int, // can be null
IsActive : bool
}
```

Response Codes

200 - Request successful. Application in response body.

For more information, please see "Common Response Codes" on page 13.

Managed Account Applications

Quick Navigation

- GET ManagedAccounts/{accountID}/Applications
- POST ManagedAccounts/{accountID}/Applications/{applicationID}
- DELETE ManagedAccounts/{accountID}/Applications/{applicationID}
- DELETE ManagedAccounts/{accountID}/Applications

GET ManagedAccounts/{accountID}/Applications

Purpose

Returns a list of Applications assigned to a Managed Account.

Required Permissions

Password Safe Account Management (Read)

URL Parameters

accountID: ID of the Managed Account

Request Body

None

Response Body

Content-Type: application/json

```
[
{
ApplicationID : int,
Name : string,
DisplayName : string,
Version : string,
Command : string,
Parameters : string,
Publisher : string,
ApplicationType : string,
FunctionalAccountID : int, // can be null
ManagedSystemID : int, // can be null
IsActive : bool
}
…
]
```

Response Codes

200 - Request successful. Applications in response body.

> For more information, please see "Common Response Codes" on page 13.

POST ManagedAccounts/{accountID}/Applications/{applicationID}

Purpose

Assigns an Application to a Managed Account.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

- **accountID:** ID of the Managed Account.
- **applicationID:** ID of the Application.

Request Body

None

Response Body

Content-Type: application/json

```
{
ApplicationID : int,
Name : string,
DisplayName : string,
Version : string,
Command : string,
Parameters : string,
Publisher : string,
ApplicationType : string,
FunctionalAccountID : int, // can be null
ManagedSystemID : int, // can be null
IsActive : bool
}
```

Response Codes

201 - Request successful. Application in response body.

> For more information, please see "Common Response Codes" on page 13.

DELETE ManagedAccounts/{accountID}/Applications/{applicationID}

Purpose

Unassigns an Application from a Managed Account by Managed Account ID and Application ID.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

- **accountID:** ID of the Managed Account.
- **applicationID:** ID of the Application.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

> For more information, please see "Common Response Codes" on page 13.

DELETE ManagedAccounts/{accountID}/Applications

Purpose

Unassigns all Managed Account Applications by Managed Account ID.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

- **accountID:** ID of the Managed Account.
- **applicationID:** ID of the Application.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

For more information, please see "Common Response Codes" on page 13.

Attributes

Quick Navigation

- GET ManagedAccounts/{managedAccountID}/Attributes
- POST ManagedAccounts/{managedAccountID}/Attributes/{attributeID}
- DELETE ManagedAccounts/{managedAccountID}/Attributes
- DELETE ManagedAccounts/{managedAccountID}/Attributes/{attributeID}

For more information on related topics, please see Attributes.

GET ManagedAccounts/{managedAccountID}/Attributes

Purpose

Returns a list of Attributes by ManagedAccount ID.

Required Permissions

Password Safe Account Management (Read), Attribute Management (Read)

URL Parameters

managedAccountID: ID of the ManagedAccount.

Request Body

None

Response Body

Content-Type: application/json

```
[
}
AttributeID : int,
AttributeTypeID : int,
ParentAttributeID : int, // can be null
ShortName : string,
LongName : string,
Description : string, ValueInt : int, // can be null
IsReadOnly: bool
},
...
]
```

Response Codes

201 - Request successful. Attributes associated with the Asset in the response body.

For more information, please see "Common Response Codes" on page 13.

POST ManagedAccounts/{managedAccountID}/Attributes/{attributeID}

Purpose

Assigns an Attribute to a Managed Account.

Required Permissions

Password Safe Account Management (Read/Write), Attribute Management (Read/Write)

URL Parameters

- **managedAccountID:** ID of the ManagedAccount.
- **attributeID:** ID of the Attribute.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
AttributeID : int,
AttributeTypeID : int,
ParentAttributeID : int, // can be null
ShortName : string,
LongName : string,
Description : string, ValueInt : int, // can be null
IsReadOnly: bool
},
...
]
```

Response Codes

201 - Request successful. Attribute in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE ManagedAccounts/{managedAccountID}/Attributes

Purpose

Deletes all Managed Account Attributes by ManagedAccount ID.

Required Permissions

Password Safe Account Management (Read/Write), Attribute Management (Read/Write)

URL Parameters

managedAccountID: ID of the ManagedAccount.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

> ℹ For more information, please see "Common Response Codes" on page 13.

DELETE ManagedAccounts/{managedAccountID}/Attributes/{attributeID}

Purpose

Deletes a Managed Account Attribute by ManagedAccount ID and Attribute ID.

Required Permissions

Password Safe Account Management (Read/Write), Attribute Management (Read/Write)

URL Parameters

- **managedAccountID:** ID of the ManagedAccount.
- **attributeID:** ID of the Attribute.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

For more information, please see "Common Response Codes" on page 13.

Credentials

Quick Navigation

- GET Credentials/{requestId}
- GET Aliases/{aliasId}/Credentials/{requestId}

> For more information on related topics, please see:
> - Requests
> - Aliases
> - Managed Accounts

GET Credentials/{requestId}

Purpose

Retrieves the credentials for an approved and active (not expired) credentials release request.

Required Permissions

None

URL Parameters

requestId: ID of the Request for which to retrieve the credentials.

Query Parameters

- **type:** (optional, default: password) Type of credentials to retrieve.
  - **password:** Returns the password in the response body.
  - **dsskey:** Returns the DSS private key in the response body.

    > ***Note:*** *The key will be returned in the state in which it was set. For example, an encrypted key will be returned encrypted.*

  - **passphrase:** Returns the dss key passphrase in the response body.

    > ***Note:*** *Only supported for encrypted DSS keys.*

Request Body

None

Response Body

```
Credentials: string
```

Response Codes

- 200 - Request successful. Credentials in the response body.
- 403 - User does not have permissions to request credentials for the indicated account or the account does not have API access enabled.
  - 4031 - User does not have permission to request credentials. 4034 - Request is not yet approved.
- 404 - Could not find the request to release. The specified request ID may have already been released or has expired.

For more information, please see "Common Response Codes" on page 13.

GET Aliases/{aliasId}/Credentials/{requestId}

Purpose

Retrieves the credentials and alias details for an approved and active (not expired) credentials release request for an Alias.

Required Permissions

None

URL Parameters

- **aliasId:** ID of the Alias.
- **requestId:** ID of the Request for which to retrieve the credentials.

Query Parameters

- **type:** (optional, default: password) Type of credentials to retrieve.
- **password:** Returns the password in response body property **Password**.
- **dsskey:** Returns the DSS private key in response body property **PrivateKey**.

***Note:** The key will be returned in the state in which it was set. For example, an encrypted key will be returned encrypted.*

- **passphrase:** returns the dss key passphrase in response body property **Passphrase**.

***Note:** Only supported for encrypted DSS keys.*

Request Body

None

Response Body

Content-Type: application/json

```
{
AliasID: int,
AliasName: string,
SystemID: int,
SystemName: string,
AccountID: int,
AccountName: string,
DomainName: string,
Password: string,
PrivateKey: string,
Passphrase: string
}
```

Response Codes

- 200 - Request successful. Account details and credentials in the response body.
- 403 - User does not have permissions to request credentials for the indicated alias or the account referenced by the alias does not have API access enabled.
    - 4031 - User does not have permission to request credentials.
    - 4034 - Request is not yet approved.
- 404 - Could not find the request to release. The specified request ID may have already been released or has expired.

For more information, please see "Common Response Codes" on page 13.

Managed Account Credentials

Quick Navigation

- PUT ManagedAccounts/{managedAccountID}/Credentials
- PUT Credentials?workgroupName={workgroupName}&assetName={assetName}&accountName={accountName}
- POST ManagedAccounts/{managedAccountID}/Credentials/Test
- POST ManagedAccounts/{managedAccountID}/Credentials/Change
- POST ManagedSystems/{systemId}/ManagedAccounts/Credentials/Change

PUT ManagedAccounts/{managedAccountID}/Credentials

Purpose

Updates the credentials for a Managed Account, optionally applying the change to the Managed System.

Required Permissions

- Password Safe Account Management (Read/Write) or
- ISA Role or Credentials Manager Role on a Smart Rule referencing the account.

URL Parameters

managedAccountID: ID of the Managed Account for which to set the credentials.

Request Body

Content-Type: application/json

```
{
Password: string,
PublicKey: string,
PrivateKey: string,
Passphrase: string,
UpdateSystem: bool
}
```

Request Body Details

- **Password:** (optional) The new password to set. If not given, generates a new random password.
- **PublicKey:** (required if PrivateKey is given and updateSystem=true) The new public key to set on the host.
- **PrivateKey:** The private key to set (provide Passphrase if encrypted).
- **Passphrase:** (optional) The passphrase to use for an encrypted private key.
- **UpdateSystem:** (default: true) Whether to update the credentials on the referenced system.

Response Body

None

Response Codes

204 - Request successful. No content in body.

For more information, please see "Common Response Codes" on page 13.

PUT Credentials?workgroupName={workgroupName}&assetName={assetName}&accountName={accountName}

Purpose

Updates the credentials for a Managed Account by Workgroup name, Asset name, and Managed Account name, optionally applying the change to the Managed System.

Required Permissions

- Password Safe Account Management (Read/Write) or
- ISA Role or Credentials Manager Role on a Smart Rule referencing the account.

Query Parameters

- **workgroupName:** Name of the Workgroup.
- **assetName:** Name of the Asset.
- **accountName:** Name of the Managed Account for which to set the credentials.

Request Body

Content-Type: application/json

```
{
    Password: string,
    PublicKey: string,
    PrivateKey: string,
    Passphrase: string,
    UpdateSystem: bool
}
```

Request Body Details

- **Password:** (optional) The new password to set. If not given, generates a new random password.
- **PublicKey:** (required if PrivateKey is given and updateSystem=true) The new public key to set on the host.
- **PrivateKey:** The private key to set (provide Passphrase if encrypted).

- **Passphrase:** (optional) The passphrase to use for an encrypted private key.
- **UpdateSystem:** (default: true) Whether to update the credentials on the referenced system.

Response Body

None

Response Codes

204 - Request Successful. No Response Body.

For more information, please see "Common Response Codes" on page 13.

POST ManagedAccounts/{managedAccountID}/Credentials/Test

Purpose

Tests the current credentials of a Managed Account.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

managedAccountID: ID of the Managed Account.

Request Body

None

Response Body

Content-Type: application/json

```
{
Success : bool
}
```

Response Body Details

Success: True if the credential test succeeded, otherwise false.

Response Codes

200 - Request Successful.

For more information, please see "Common Response Codes" on page 13.

POST ManagedAccounts/{managedAccountID}/Credentials/Change

Purpose

Changes the current Credentials of a Managed Account.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

managedAccountID: ID of the Managed Account.

Request Body (optional)

Content-Type: application/json

```
{
Queue : bool
}
```

Request Body Details

Queue: (default: false) True to queue the change for background processing, otherwise false. When Queue is false the credentials change is immediate.

Response Body

None

Response Codes

204 - Request successful. No content in body.

For more information, please see "Common Response Codes" on page 13.

POST ManagedSystems/{systemId}/ManagedAccounts/Credentials/Change

Purpose

Queues Credentials' changes for all active Managed Accounts for a Managed System.

Required Permissions

Password Safe API Global Quarantine (Read/Write)

URL Parameters

systemId: ID of the Managed System.

Request Body

None

Response Body

None

Response Codes

204 - Request successful. No content in body.

> ℹ For more information, please see "Common Response Codes" on page 13.

Directories

Quick Navigation

- GET Directories
- GET Directories/{id}
- POST Workgroups/{id}/Directories
- PUT Directories/{id}
- DELETE Directories

For more information on related topics, please see Managed Systems.

GET Directories

Purpose

Returns a list of Directories.

Required Permissions

One of: Password Safe System Management (Read), Password Safe Domain Management (Read)

Request Body

None

Response Body

```
Content-type: application/json [
{
DirectoryID : int,
WorkgroupID : int,
PlatformID : int,
DomainName : string,
ForestName : string,
NetBiosName : string,
UseSSL : bool,
Port : int, // can be null
Timeout : short,
Description : string,
ContactEmail : string,
PasswordRuleID : int,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AccountNameFormat : int,
AutoManagementFlag : bool,
FunctionalAccountID : int, // can be null
CheckPasswordFlag : bool,
ChangePasswordAfterAnyReleaseFlag : bool,
ResetPasswordOnMismatchFlag : bool,
ChangeFrequencyType : string,
ChangeFrequencyDays : int,
```

Response Codes

200 - Request successful. Directory in response body.

For more information, please see "Common Response Codes" on page 13.

GET Directories/{id}

Purpose

Returns a Directory by ID.

Required Permissions

One of: Password Safe System Management (Read), Password Safe Domain Management (Read)

URL Parameters

id: ID of the Directory.

Request Body

None

Response Body

Content-Type: application/json

```
{
DirectoryID : int,
WorkgroupID : int,
PlatformID : int,
DomainName : string,
ForestName : string,
NetBiosName : string,
UseSSL : bool,
Port : int, // can be null
Timeout : short,
Description : string,
ContactEmail : string,
PasswordRuleID : int,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AccountNameFormat : int,
AutoManagementFlag : bool,
FunctionalAccountID : int, // can be null
CheckPasswordFlag : bool,
ChangePasswordAfterAnyReleaseFlag : bool,
ResetPasswordOnMismatchFlag : bool,
ChangeFrequencyType : string,
ChangeFrequencyDays : int,
ChangeTime : string,
}
```

POST Workgroups/{id}/Directories

Purpose

Creates a new Directory in the Workgroup referenced by ID.

Required Permissions

One of: Password Safe System Management (Read/Write), Password Safe Domain Management (Read/Write)

URL Parameters

id: ID of the Workgroup

Request Body

Content-Type: application/json

```
{
PlatformID : int,
DomainName : string,
ForestName : string,
NetBiosName : string,
UseSSL : bool,
Port : int, // can be null
Timeout : short,
Description : string,
ContactEmail : string,
PasswordRuleID : int,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AccountNameFormat : int,
AutoManagementFlag : bool,
FunctionalAccountID : int, // can be null
CheckPasswordFlag : bool,
ChangePasswordAfterAnyReleaseFlag : bool,
ResetPasswordOnMismatchFlag : bool,
ChangeFrequencyType : string,
ChangeFrequencyDays : int,
ChangeTime : string,
}
```

Request Body Details

- **PlatformID:** (required) ID of the Platform
- **DomainName:** (required) Name of the Domain
- **ForestName:** (required for Active Directory only, not applicable to LDAP) Name of the Directory Forest
- **NetBiosName:** (required for Active Directory, optional for LDAP) NetBIOS Name of the Directory.
- **UseSSL:** (default: false) True to use an SSL connection, otherwise false.
- **Port:** (set automatically for Active Directory, optional for LDAP,) The port used to connect to the host. If null and the related Platform is LDAP, Password Safe uses Platform.DefaultPort.
- **Timeout:** (seconds, default: 30) Connection timeout – Length of time in seconds before a slow or unresponsive connection to the system fails.
- **PasswordRuleID:** (default: 0) ID of the default Password Rule assigned to Managed Accounts created under this Managed System.
- **ReleaseDuration:** (minutes: 1-525600, default: 120) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600, default: 525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600, default: 120) Default Information Systems Administrator (ISA) release duration.
- **AccountNameFormat:** (Active Directory only, default: 0) Account Name format to use:
  - **0:** Domain and Account - Use ManagedAccount.DomainName\ManagedAccount.AccountName
  - **1:** UPN – Use the Managed Account UPN
  - **2:** SAM – Use the Managed Account SAM Account Name
- **AutoManagementFlag:** (default: false) True if password auto-management is enabled, otherwise false. Can be set if Platform.AutoManagementFlag is true.
  - **FunctionalAccountID:** (required if AutoManagementFlag is true) ID of the Functional Account used for Managed Account password changes. FunctionalAccount.PlatformID must match the PlatformID.
  - **CheckPasswordFlag:** True to enable password testing, otherwise false.
  - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
  - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
  - **ChangeFrequencyType:** (default: first) The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month
    - **last:** Changes scheduled for the last day of the month
    - **xdays:** Changes scheduled every x days (see ChangeFrequencyDays)
  - **ChangeFrequencyDays:** (days: 1-90, required if ChangeFrequencyType is xdays) When ChangeFrequencyType is “xdays”, password changes take place this configured number of days.
  - **ChangeTime:** (24hr format: 00:00-23:59, default: 23:30) UTC time of day scheduled password changes take place.

Response Body

Content-Type: application/json

```
{
DirectoryID : int,
WorkgroupID : int,
PlatformID : int,
DomainName : string,
ForestName : string,
NetBiosName : string,
UseSSL : bool,
Port : int, // can be null
Timeout : short,
Description : string,
ContactEmail : string,
PasswordRuleID : int,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AccountNameFormat : int,
AutoManagementFlag : bool,
FunctionalAccountID : int, // can be null
CheckPasswordFlag : bool,
ChangePasswordAfterAnyReleaseFlag : bool,
ResetPasswordOnMismatchFlag : bool,
ChangeFrequencyType : string,
ChangeFrequencyDays : int,
ChangeTime : string,
}
```

Response Codes

201 - Request successful. Directory in response body.

For more information, please see "Common Response Codes" on page 13.

PUT Directories/{id}

Purpose

Updates an existing Directory by ID.

Required Permissions

One of: Password Safe System Management (Read/Write), Password Safe Domain Management (Read/Write)

URL Parameters

id: ID of the Directory

Request Body

Content-Type: application/json

```
{
PlatformID : int,
WorkgroupID : int,
DomainName : string,
ForestName : string,
NetBiosName : string,
UseSSL : bool,
Port : int, // can be null
Timeout : short,
Description : string,
ContactEmail : string,
PasswordRuleID : int,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AccountNameFormat : int,
AutoManagementFlag : bool,
FunctionalAccountID : int, // can be null
CheckPasswordFlag : bool,
ChangePasswordAfterAnyReleaseFlag : bool,
ResetPasswordOnMismatchFlag : bool,
ChangeFrequencyType : string,
ChangeFrequencyDays : int,
ChangeTime : string,
}
```

Request Body Details

- **WorkgroupID:** (required) ID of the Workgroup.
- **PlatformID:** (required) ID of the Platform.
- **DomainName:** (required) Name of the Domain.
- **ForestName:** (required for Active Directory only, not applicable to LDAP) Name of the Directory Forest.
- **NetBiosName:** (required for Active Directory, optional for LDAP) NetBIOS Name of the Directory.
- **UseSSL:** (default: false) True to use an SSL connection, otherwise false.
- **Port:** (set automatically for Active Directory, optional for LDAP,) The port used to connect to the host. If null and the related Platform is LDAP, Password Safe uses Platform.DefaultPort.
- **Timeout:** (seconds, default: 30) Connection timeout – Length of time in seconds before a slow or unresponsive connection to the system fails.
- **PasswordRuleID:** (default: 0) ID of the default Password Rule assigned to Managed Accounts created under this Managed System.
- **ReleaseDuration:** (minutes: 1-525600, default: 120) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600, default: 525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600, default: 120) Default Information Systems Administrator (ISA) release duration.
- **AccountNameFormat:** (Active Directory only, default: 0) Account Name format to use:
    - **0:** Domain and Account - Use ManagedAccount.DomainName\ManagedAccount.AccountName.
    - **1:** UPN – Use the Managed Account UPN.
    - **2:** SAM – Use the Managed Account SAM Account Name.
- **AutoManagementFlag:** (default: false) True if password auto-management is enabled, otherwise false. Can be set if Platform.AutoManagementFlag is true.

- **FunctionalAccountID:** (required if AutoManagementFlag is true) ID of the Functional Account used for Managed Account password changes. FunctionalAccount.PlatformID must match the PlatformID.
- **CheckPasswordFlag:** True to enable password testing, otherwise false.
- **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
- **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
- **ChangeFrequencyType:** (default: first) The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month.
    - **last:** Changes scheduled for the last day of the month.
    - **xdays:** Changes scheduled every x days (see ChangeFrequencyDays).
- **ChangeFrequencyDays:** (days: 1-90, required if ChangeFrequencyType is xdays) When ChangeFrequencyType is “xdays”, password changes take place this configured number of days.
- **ChangeTime:** (24hr format: 00:00-23:59, default: 23:30) UTC time of day scheduled password changes take place.

Response Body

Content-Type: application/json

```
{
DirectoryID : int,
WorkgroupID : int,
PlatformID : int,
DomainName : string,
ForestName : string,
NetBiosName : string,
UseSSL : bool,
Port : int, // can be null
Timeout : short,
Description : string,
ContactEmail : string,
PasswordRuleID : int,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AccountNameFormat : int,
AutoManagementFlag : bool,
FunctionalAccountID : int, // can be null
CheckPasswordFlag : bool,
ChangePasswordAfterAnyReleaseFlag : bool,
ResetPasswordOnMismatchFlag : bool,
ChangeFrequencyType : string,
ChangeFrequencyDays : int,
ChangeTime : string,
}
```

Response Codes

200 - Request successful. Directory in response body.

For more information, please see "Common Response Codes" on page 13.

DELETE Directories

Purpose

Deletes a Directory by ID.

Required Permissions

One of: Password Safe System Management (Read/Write), Password Safe Domain Management (Read/Write)

URL Parameters

id: ID of the Directory.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

For more information, please see "Common Response Codes" on page 13.

Oracle Internet Directories

Quick Navigation

- GET OracleInternetDirectories
- GET OracleInternetDirectories/{id}
- GET Organizations/{id}/OracleInternetDirectories
- POST OracleInternetDirectories/{id}/Services/Query
- POST OracleInternetDirectories/{id}/Test

GET OracleInternetDirectories

Purpose

Returns a list of Oracle Internet Directories.

Required Permissions

Password Safe System Management (Read)

Request Body

None

Response Body

Content-type: application/json

```
[{
OrganizationID : Guid,
OracleInternetDirectoryID : Guid,
Name : string,
Description : string,
},
…]
```

Response Codes

200 – Request successful. Oracle Internet Directories in response body.

For more information, please see "Common Response Codes" on page 13.

GET OracleInternetDirectories/{id}

Purpose

Returns an Oracle Internet Directory by ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

id: ID of the Oracle Internet Directory.

Request Body

None

Response Body

Content-Type: application/json

```
{
OrganizationID : Guid,
OracleInternetDirectoryID : Guid,
Name : string,
Description : string,
}
```

Response Codes

200 – Request successful. Oracle Internet Directory in response body.

For more information, please see "Common Response Codes" on page 13.

GET Organizations/{id}/OracleInternetDirectories

Purpose

Returns a list of Oracle Internet Directories by Organization ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

id: ID of the Organization.

Request Body

None

Response Body

Content-Type: application/json

```
[{
OrganizationID : Guid,
OracleInternetDirectoryID : Guid,
Name : string,
Description : string,
},
...]
```

Response Codes

200 – Request successful. Oracle Internet Directories in response body.

For more information, please see "Common Response Codes" on page 13.

POST OracleInternetDirectories/{id}/Services/Query

Purpose

Queries and returns DB Services for an Oracle Internet Directory by ID.

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

id: ID of the Oracle Internet Directory.

Request Body

None

Response Body

Content-Type: application/json

```
{
Success : bool,
Message : string,
Services : [{
  Name : string,
  },
  ...]
}
```

Response Codes

200 - Request successful. Oracle Internet Directory query result in response body.

For more information, please see "Common Response Codes" on page 13.

POST OracleInternetDirectories/{id}/Test

Purpose

Tests the connection to an Oracle Internet Directory by ID.

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

id: ID of the Oracle Internet Directory.

Request Body

None

Response Body

Content-Type: application/json

```
{
Success : bool,
}
```

Response Codes

200 – Request successful. Oracle Internet Directory test result in response body.

For more information, please see "Common Response Codes" on page 13.

DSS Key Policies

> ***Note:*** *DSS Key Policies are formerly known as DSS Key Rules but the API remains* ***DSSKeyRules*** *to be compatible with earlier versions.*

Quick Navigation

- GET DSSKeyRules
- GET DSSKeyRules/{id}

GET DSSKeyRules

Purpose

Returns a list of DSS Key Rules.

Required Permissions

Password Safe System Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
DSSKeyRuleID: int,
Name: string,
Description: string,
KeyType: string,
KeySize: int,
EncryptionType: char,
PasswordRuleID: int, // can be null
},
…
]
```

Response Body Details

- **KeyType:** (RSA, DSA) The type of key to generate
- **EncryptionType:** The type of key encryption to use:
  - **A** - Auto-managed passphrase, generated using the associated Password Rule (see PasswordRuleID)
  - **N** - No encryption

- **PasswordRuleID:** (given when **EncryptionType** is **A**) ID of the Password Rule used to auto-generate the passphrase for DSS Key encryption

Response Codes

200 - Request successful. DSS Key Rules in the response body.

For more information, please see "Common Response Codes" on page 13.

GET DSSKeyRules/{id}

Purpose

Returns a DSS Key Rule by ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

id: ID of the DSS Key Rule.

Request Body

None

Response Body

Content-Type: application/json

```
{
DSSKeyRuleID: int,
Name: string,
Description: string,
KeyType: string,
KeySize: int,
EncryptionType: char,
PasswordRuleID: int, // can be null
}
```

Response Body Details

- **KeyType:** The type of key to generate (RSA, DSA)
- **EncryptionType:** The type of key encryption to use:
  - **A** - Auto-managed passphrase, generated using the associated Password Rule (see PasswordRuleID)
  - **N** - No encryption
- **PasswordRuleID:** (given when **EncryptionType** is **A**) ID of the Password Rule used to auto-generate the passphrase for DSS Key encryption

Response Codes

200 - Request successful. DSS Key Rule in the response body.

For more information, please see "Common Response Codes" on page 13.

Entity Types

Entity Types define the types of entities within Password Safe (i.e. Asset, Database, Directory, and Cloud).

For more information on related topics, please see Platforms.

GET EntityTypes

Purpose

Returns a list of Entity Types.

Required Permissions

None

Request Body

None

Response Body

Content-Type: application/json

```
[
{
EntityTypeID: int,
Name: string,
Description: string,
},
…
]
```

Response Codes

200 - Request successful. Entity Types in the response body.

For more information, please see "Common Response Codes" on page 13.

Functional Accounts

Quick Navigation

- GET FunctionalAccounts
- GET FunctionalAccounts/{id}
- GET FunctionalAccounts/{id}/ManagedSystems
- POST FunctionalAccounts
- DELETE FunctionalAccounts/{id}

GET FunctionalAccounts

Purpose

Returns a list of Functional Accounts.

Required Permissions

Password Safe Account Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
FunctionalAccountID: int,
PlatformID: int,
DomainName: string,
AccountName: string,
DisplayName: string,
Description: string,
ElevationCommand: string,
SystemReferenceCount: int,
},
…
]
```

Response Body Details

- **PlatformID:** ID of the Platform to which the account belongs.
- **DomainName:** Domain Name of the account.
- **AccountName:** Name of the account (does not include domain name).
- **DisplayName:** The display name or alias for the account.

- **Description:** Description of the account.
- **ElevationCommand:** Elevation Command used for SSH connections (sudo, pbrun, pmrun).
- **SystemReferenceCount:** The count of Managed Systems that reference the Functional Account.

Response Codes

200 - Request successful. Functional Accounts in the response body.

For more information, please see "Common Response Codes" on page 13.

GET FunctionalAccounts/{id}

Purpose

Returns a Functional Account by ID.

Required Permissions

Password Safe Account Management (Read)

URL Parameters

id: ID of the Functional Account.

Request Body

None

Response Body

Content-Type: application/json

```
{
FunctionalAccountID: int,
PlatformID: int, DomainName: string,
AccountName: string,
DisplayName: string,
Description: string,
ElevationCommand: string,
SystemReferenceCount: int,
}
```

Response Body Details

- **PlatformID:** ID of the Platform to which the account belongs.
- **DomainName:** Domain Name of the account.
- **AccountName:** Name of the account (does not include domain name).
- **DisplayName:** The display name or alias for the account.
- **Description:** Description of the account.

- **ElevationCommand:** Elevation Command used for SSH connections (sudo, pbrun, pmrun).
- **SystemReferenceCount:** The count of Managed Systems that reference the Functional Account.

Response Codes

200 - Request successful. Functional Account in the response body.

For more information, please see "Common Response Codes" on page 13.

GET FunctionalAccounts/{id}/ManagedSystems

Purpose

Returns a list of Managed Systems auto-managed by the Functional Account referenced by ID.

Required Permissions

- Password Safe System Management (Read)
- Password Safe Account Management (Read)

URL Parameters

id: ID of the Functional Account.

Query Parameters (optional)

- **limit:** (optional) Number of records to return (default: 1000) .
- **offset:** (optional) Number of records to skip before returning <limit> records (default: 0).

Request Body

None

Response Body (when limit is not given)

Content-Type: application/json

```
[
{
ManagedSystemID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
PlatformID : int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
},
…
]
```

Response Body (when limit is given)

Content-Type: application/json

```
{
TotalCount : int,
Data :
[
{
ManagedSystemID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
PlatformID : int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
},
…
]
}
```

Response Body Details

- **ManagedSystemID:** ID of the Managed System.
- **AssetD:** Asset ID; set if the Managed System is an Asset or a Database.
- **DatabaseID:** Database ID; set if the Managed System is a Database.
- **DirectoryID:** Directory ID; set if the Managed System is a Directory.
- **CloudID:** Cloud System ID; set if the Managed System is a Cloud System.
- **SystemName:** Name of the related entity (Asset, Directory, Database, or Cloud).
- **PlatformID:** ID of the Managed System Platform.
- **NetBiosName:** (Managed Domains only) Domain NetBIOS name. Setting this value will allow Password Safe to fall back to the NetBIOS name if needed.
- **Port:** The port used to connect to the host. If null and the related Platform.PortFlag is true, Password Safe uses Platform.DefaultPort for communication.
- **Timeout:** (seconds) Connection timeout - Length of time in seconds before a slow or unresponsive connection to the system fails.
- **SshKeyEnforcementMode:** Enforcement mode for SSH host keys
  - **0:** None
  - **1:** Auto - Auto Accept Initial Key
  - **2:** Strict - Manually Accept Keys
- **PasswordRuleID:** ID of the default Password Rule assigned to Managed Accounts created under this Managed System.
- **DSSKeyRuleID:** ID of the default DSS Key Rule assigned to Managed Accounts created under this Managed System.
- **LoginAccountID:** ID of the Functional Account used for SSH Session logins.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
  - **FunctionalAccountID:** ID of the Functional Account used for local Managed Account password changes.
  - **ElevationCommand:** Elevation Command to use (sudo, pbrun, pmrun).
  - **CheckPasswordFlag:** True to enable password testing, otherwise false.
  - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
  - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.

  - **ChangeFrequencyType:** The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month
    - **last:** Changes scheduled for the last day of the month
    - **xdays:** Changes scheduled every x days (see ChangeFrequencyDays)
  - **ChangeFrequencyDays:** (days: 1-999) When ChangeFrequencyType is "xdays", password changes take place this configured number of days.
  - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.

POST FunctionalAccounts

Purpose

Creates a Functional Account.

Required Permissions

Password Safe Account Management (Read/Write)

Request Body

Content-Type: application/json

Request Body Details

- **FunctionalAccountID:** (required) ID of the Functional Account.
- **PlatformID:** (required) ID of the Platform to which the account belongs.
- **DomainName:** (optional) Domain Name of the account. Can be set if Platform.DomainNameFlag is true
- **AccountName:** (required) Name of the account (do not include domain name).
- **DisplayName:** (optional) The display name or alias for the account. If not given, uses the AccountName. Must be unique for the Platform.
- **Password:** (required) The current account password.
- **PrivateKey:** (optional) DSS Private Key. Can be set if Platform.DSSFlag is true.
- **Passphrase:** (required when PrivateKey is an encrypted DSS key) DSS Passphrase. Can be set if Platform.DSSFlag is true.
- **Description:** (optional) Description of the account.
- **ElevationCommand:** (optional) Elevation Command to use for SSH connections. Can be set if Platform.SupportsElevationFlag is true. (sudo, pbrun, pmrun).

Response Body

Content-Type: application/json

```
{
FunctionalAccountID: int,
PlatformID: int,
DomainName: string,
AccountName: string,
DisplayName: string,
Password: string,
PrivateKey : string,
Passphrase : string,
Description: string,
ElevationCommand: string,
}
```

Response Body Details

- **PlatformID:** ID of the Platform to which the account belongs.
- **DomainName:** Domain Name of the account.
- **AccountName:** Name of the account (does not include domain name).
- **DisplayName:** The display name or alias for the account.
- **Description:** Description of the account.
- **ElevationCommand:** Elevation Command used for SSH connections (sudo, pbrun, pmrun).
- **SystemReferenceCount:** The count of Managed Systems that reference the Functional Account.

Response Codes

201 - Request successful. Functional Account in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE FunctionalAccounts/{id}

Purpose

Deletes a Functional Account by ID.

Required Permissions

Password Safe Account Management (Read/Write)

Other Requirements

The Functional Account cannot be referenced by any Managed Systems.

URL Parameters

id: ID of the Functional Account.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

For more information, please see "Common Response Codes" on page 13.

ISA Requests

The ISARequests endpoint is for Information Systems Administrator (ISA) role access.

> For Requestor and Requestor/Approver role access see POST Requests at "Requests" on page 265.

POST ISARequests

Purpose

Creates a new Information Systems Administrator (ISA) release request and returns the requested credentials.

Similar to POST Requests (AccessType=View) and GET Credentials in a single call.

Required Roles

ISA Role to Managed Account referenced by ID

Query Parameters

- **type:** (optional, default: password) Type of credentials to retrieve.
  - **password:** Returns the password in the response body.
  - **dsskey:** Returns the DSS private key in the response body.

    > ***Note:*** *The key will be returned in the state in which it was set. For example, an encrypted key will be returned encrypted.*

  - **passphrase:** Returns the dss key passphrase in the response body.

    > ***Note:*** *Only supported for encrypted DSS keys.*

Request Body

Content-Type: application/json

```
{
SystemID: int,
AccountID: int,
DurationMinutes: int, // can be null
Reason: string
}
```

Request Body Details

- **SystemID:** (required) ID of the Managed System to request.
- **AccountID:** (required) ID of the Managed Account to request.

- **DurationMinutes:** (optional) The request duration (in minutes). If omitted, uses the value ManagedAccount.ISAReleaseDuration.
- **Reason:** (optional) The reason for the request.

Response Body

```
{
Credentials: string
}
```

Response Codes

201 - Request successful. Credentials in the response body.

For more information, please see "Common Response Codes" on page 13.

ISA Sessions

The ISASessions endpoint is for Information Systems Administrator (ISA) role access.

For Requestor and Requestor/Approver role access see POST Requests and POST Sessions.

POST ISASessions

Purpose

Creates a new Information Systems Administrator (ISA) release request and returns the requested session.

Similar to POST Requests (AccessType=**RDP** or AccessType=**SSH**) and POST Sessions in a single call.

Required Roles

- ISA Role to Managed Account referenced by ID

Request Body

Content-Type: application/json

```
{
SessionType : string,
SystemID: int,
AccountID: int,
DurationMinutes : int, // can be null
Reason : string
}
```

Request Body Details

- **SessionType:** (required) The type of session to create (SSH or sshticket, RDP or rdpticket, or rdpfile)
- **SystemID:** (required) ID of the Managed System to request.
- **AccountID:** (required) ID of the Managed Account to request.
- **DurationMinutes:** (optional) The request duration (in minutes). If omitted, uses the value ManagedAccount.ISAReleaseDuration.
- **Reason:** (optional) The reason for the request.

Response Body (SSH or sshticket)

Content-Type: application/json

```
{
ID : string,
Ticket : string,
Host : string,
Port : string,
TicketAtHost : string,
Link : string,
Command : string
}
```

Response Body (RDP or rdpticket)

Content-Type: application/json

```
{
ID : string,
Ticket : string,
Host : string,
Port : string
}
```

Response Body (rdpfile)

RDP File as an attachment.

Response Codes

201- Request successful. Session details or RDP file in the response body.

For more information, please see "Common Response Codes" on page 13.

Keystrokes

Quick Navigation

- GET Sessions/{sessionId:int}/Keystrokes
- GET Keystrokes/{id:long}
- POST Keystrokes/Search

GET Sessions/{sessionId:int}/Keystrokes

Purpose

Returns a list of Keystrokes by Session ID.

Required Roles

Password Safe Auditor Role, ISA Role, or a member of BeyondInsight Administrators group.

URL Parameters

sessionId: ID of recorded RDP/SSH session

Response Body

Content-Type: application/json

```
[
{
KeystrokeID: long,
SessionID: int,
TimeMarker: long,
Type: byte,
Data: string
},
...
]
```

Response Codes

200 - Request successful. Keystrokes are in response body.

For more information, please see "Common Response Codes" on page 13.

GET Keystrokes/{id:long}

Purpose

Returns a Keystroke by ID.

Required Roles

Password Safe Auditor Role, ISA Role, or a member of BeyondInsight Administrators group.

URL Parameters

id: ID of a Keystroke.

Response Body

Content-Type: application/json

```
{
KeystrokeID: long,
SessionID: int,
TimeMarker: long,
Type: byte ,
Data: string
}
```

Response Codes

200 - Request successful. Keystroke in response body.

For more information, please see "Common Response Codes" on page 13.

POST Keystrokes/Search

Purpose

Search for Keystrokes.

Required Roles

Password Safe Auditor Role, ISA Role, or a member of BeyondInsight Administrators group.

Request Body

Content-Type: application/json

```
{

Data: string,

Type: byte

}
```

Request Body Details

- **Data:** (required) Keyword(s) for which to search.
- **Type:** (default: 0) Type of keystrokes: 0 - All, 1 - StdIn, 2 - StdOut, 4 - Window Event, 5 - User Event.

Response Body

Content-Type: application/json

```
[
{
KeystrokeID: long,
SessionID: int,
TimeMarker: long,
Type: byte,
Data: string
},
...
]
```

Response Codes

200 - Request successful. Keystrokes are in response body.

For more information, please see "Common Response Codes" on page 13.

Linked Accounts

Linked Accounts are Directory Managed Accounts that are linked to Asset-based Managed Systems.

> ***Note:*** *Directory Accounts can be linked only to managed Assets and managed Databases.*

Quick Navigation

- GET ManagedSystems/{systemID}/LinkedAccounts
- POST ManagedSystems/{systemID}/LinkedAccounts/{accountID}
- DELETE ManagedSystems/{systemID}/LinkedAccounts
- DELETE ManagedSystems/{systemID}/LinkedAccounts/{accountID}

GET ManagedSystems/{systemID}/LinkedAccounts

Purpose

Returns a list of linked Directory Managed Accounts by Managed System ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

systemID: ID of the Managed System.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
},
...
]
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (see ChangeServicesFlag), otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 means unlimited) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.

- **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
- **ChangeFrequencyType:** The change frequency for scheduled password changes:
  - **first:** Changes scheduled for the first day of the month
  - **last:** Changes scheduled for the last day of the month
  - **xdays:** Changes scheduled every x days (see ChangeFrequencyDays)
- **ChangeFrequencyDays:** (days: 1-999) When ChangeFrequencyType is **xdays**, password changes take place this configured number of days.
- **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.

- **ParentAccountID:** If this is a subscribed account (see IsSubscribedAccount), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

> For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 - Request successful. Linked Managed Accounts in the response body.

> For more information, please see "Common Response Codes" on page 13.

POST ManagedSystems/{systemID}/LinkedAccounts/{accountID}

Purpose

Links a Directory Managed Account to the Managed System referenced by ID.

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

- **systemID:** ID of the Managed System.
- **accountID:** ID of the Directory Managed Account.

Request Body

None

Response Body

Content-Type: application/json

```
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
}
```

Response Body Details

- **AccountName:** The name of the account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (see ChangeServicesFlag), otherwise false.

- **ReleaseDuration:**(minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 is unlimited) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
  - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
  - **CheckPasswordFlag:** True to enable password testing, otherwise false.
  - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
  - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
  - **ChangeFrequencyType:** The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month
    - **last:** Changes scheduled for the last day of the month
    - **xdays:** Changes scheduled every x days (ChangeFrequencyDays)
  - **ChangeFrequencyDays:** (days: 1-999) When ChangeFrequencyType is **xdays**, password changes take place this configured number of days.
  - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
- **ParentAccountID:** If this is a subscribed account (IsSubscribedAccount), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

> i For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

- 200 - Account was already linked. Directory Managed Account in the response body.
- 201 - Account was linked successfully. Directory Managed Account in the response body.

> i For more information, please see "Common Response Codes" on page 13.

DELETE ManagedSystems/{systemID}/LinkedAccounts

Purpose

Unlinks all Directory Managed Accounts from the Managed System by ID.

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

systemID: ID of the Managed System.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

> ℹ For more information, please see "Common Response Codes" on page 13.

DELETE ManagedSystems/{systemID}/LinkedAccounts/{accountID}

Purpose

Unlinks a Directory Managed Account from the Managed System by ID.

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

- **systemID:** ID of the Managed System.
- **accountID:** ID of the Directory Managed Account.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

For more information, please see "Common Response Codes" on page 13.

Managed Accounts

There are two different ways to interact with managed accounts:

1. **Role-based:Requestor**, **Requestor/Approver**, or **ISA** role assigned for requesting access to a specific managed account.
2. **Permission-based**: A user with appropriate **Password Safe Account Management** permission for provisioning accounts and viewing the definition of a managed account.

Role-based Access

Quick Navigation

- GET ManagedAccounts
- GET ManagedAccounts?systemName={systemName}&accountName={accountName}

For more information on related topics, please see:

- Managed Systems
- Requests
- Quick Rules
- Smart Rules

GET ManagedAccounts

Purpose

Returns a list of managed accounts (or a single managed account depending on the query parameters provided) that can be requested by the current user.

Required Roles

Requestor, Requestor/Approver, or ISA role

Other Requirements

Only managed accounts with the **Enable for API Access** setting enabled will be returned.

Query Parameters

- **systemName:** (optional) Name of the managed system.
- **accountName:** (optional) Name of the managed account.
- **workgroupName:** (optional) Name of the Workgroup.
- **applicationDisplayName:** (optional, when given, type must be application) Display name of the application.
- **ipAddress:** (optional, when given type must be one of system, domainlinked, or database) IP Address of the managed asset.

- **type:** (optional/recommended) Type of the managed account to return.
    - **system:** Returns local accounts.
    - **domainlinked:** Returns domain accounts linked to systems.
    - **database:** Returns database accounts.
    - **cloud:** Returns cloud system accounts.
    - **application:** Returns application accounts
- **limit:** (optional) (default: 1000) Number of records to return
- **offset:** (optional) (default: 0) Number of records to skip before returning <limit> records

Request Body

None

Response Body (when both systemName and accountName are given)

Content-Type: application/json

```
{
PlatformID: int,
SystemId: int,
SystemName: string,
DomainName: string,
AccountId: int,
AccountName: string,
InstanceName: string,
ApplicationID: int,
ApplicationDisplayName: string,
DefaultReleaseDuration: int,
MaximumReleaseDuration: int,
LastChangeDate: datetime,
NextChangeDate: datetime,
IsChanging: bool,
IsISAAccess: bool,
PreferredNodeID: string
}
```

Response Body (all other combinations of query parameters)

Content-Type: application/json

```
[
{
PlatformID: int,
SystemId: int,
SystemName: string,
DomainName: string,
AccountId: int,
AccountName: string,
InstanceName: string,
ApplicationID: int,
ApplicationDisplayName: string,
DefaultReleaseDuration: int,
MaximumReleaseDuration: int,
LastChangeDate: datetime,
NextChangeDate: datetime,
IsChanging: bool,
IsISAAccess: bool,
PreferredNodeID: string
},
...
]
```

Response Body Details

- **PlatformID:** ID of the managed system Platform.
- **SystemId:** ID of the managed system.
- **SystemName:** Name of the managed system.
- **DomainName:** The domain name for a domain-type account.
- **AccountId:** ID of the managed account.
- **AccountName:** Name of the managed account.
- **InstanceName:** Database instance name of a database-type managed system, or empty for the default instance.
- **ApplicationID:** ID of the application for application-based access

> For more information, please see the query parameter "type: (optional/recommended) Type of the managed account to return.system: Returns local accounts.domainlinked: Returns domain accounts linked to systems.database: Returns database accounts.cloud: Returns cloud system accounts.application: Returns application accounts" on page 179.

- **ApplicationDisplayName:** Display name of the application for application-based access

> For more information, please see the query parameter "type: (optional/recommended) Type of the managed account to return.system: Returns local accounts.domainlinked: Returns domain accounts linked to systems.database: Returns database accounts.cloud: Returns cloud system accounts.application: Returns application accounts" on page 179.

- **DefaultReleaseDuration (minutes):** Default release duration.
- **MaximumReleaseDuration (minutes):** Maximum release duration.
- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.
- **IsISAAccess:** True if the account is for Information Systems Administrator (ISA) access, otherwise false.

Note:
*If **true**, credential access is through POST ISA Requests and session access is through POST ISA Sessions.*

*If **false**, credential access is through "POST Requests" on page 266 and GET Credentials; session access is through "POST Requests" on page 266 and "POST Requests/{requestID}/Sessions" on page 287.*

- **PreferredNodeID:** ID of the node that is preferred for establishing Sessions. If no node is preferred, returns the local node ID.

Response Codes

200 - Request successful. Requestable Account(s) in the response body.

For more information, please see "Common Response Codes" on page 13.

GET ManagedAccounts?systemName={systemName}&accountName={accountName}

***Note:** This API has been replaced by optional query parameters on GET ManagedAccounts.*

Provisioning

Quick Navigation

- GET ManagedAccounts/{id}
- GET ManagedSystems/{systemID}/ManagedAccounts
- GET ManagedSystems/{systemID}/ManagedAccounts?name={name}
- PUT ManagedAccounts/{id}
- POST ManagedSystems/{systemID}/ManagedAccounts
- DELETE ManagedAccounts/{id}
- DELETE ManagedSystems/{systemID}/ManagedAccounts/{accountName}

GET ManagedAccounts/{id}

Purpose

Returns a Managed Account by ID.

Required Permissions

Password Safe Account Management (Read)

URL Parameters

id: ID of the Managed Account.

Request Body

None

Response Body

Content-Type: application/json

```
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
UserPrincipalName : string,
SAMAccountName : string,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
}
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **UserPrincipalName:** (Active Directory Managed Systems only) The account User Principal Name of an Active Directory account.
- **SAMAccountName:** (Active Directory Managed Systems only) The account SAM Account Name of an Active Directory account.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.

- **MaxConcurrentRequests:** (0-999, default: 1) Maximum number of concurrent password requests for this account. A value of zero denotes unlimited requests.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
  - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
  - **CheckPasswordFlag:** True to enable password testing, otherwise false.
  - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
  - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
  - **ChangeFrequencyType:** The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month
    - **last:** Changes scheduled for the last day of the month
    - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
  - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
  - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
- **ParentAccountID:** If this is a subscribed account (**IsSubscribedAccount**), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 - Request successful. Managed Account in the response body.

For more information, please see "Common Response Codes" on page 13.

GET ManagedSystems/{systemID}/ManagedAccounts

Purpose

Returns a list of Managed Accounts by Managed System ID.

Required Permissions

Password Safe Account Management (Read)

URL Parameters

systemID: ID of the Managed System.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
UserPrincipalName : string,
SAMAccountName : string,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
},
...
]
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.

- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **UserPrincipalName:** (Active Directory Managed Systems only) The account User Principal Name of an Active Directory account.
- **SAMAccountName:** (Active Directory Managed Systems only) The account SAM Account Name of an Active Directory account.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, default: 1) Maximum number of concurrent password requests for this account. A value of zero denotes unlimited requests.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
    - **ChangeFrequencyType:** The change frequency for scheduled password changes:
        - **first:** Changes scheduled for the first day of the month
        - **last:** Changes scheduled for the last day of the month
        - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
    - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
    - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
- **ParentAccountID:** If this is a subscribed account (**IsSubscribedAccount**), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

> For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.

- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 - Request successful. Managed Accounts in the response body.

For more information, please see "Common Response Codes" on page 13.

GET ManagedSystems/{systemID}/ManagedAccounts?name={name}

Purpose

Returns a Managed Account by Managed System ID and Managed Account name.

Required Permissions

Password Safe Account Management (Read)

URL Parameters

systemID: ID of the Managed System.

Query Parameters

name: Name of the Managed Account.

Request Body

None

Response Body

Content-Type: application/json

```
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
UserPrincipalName : string,
SAMAccountName : string,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
}
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **UserPrincipalName:** (Active Directory Managed Systems only) The account User Principal Name of an Active Directory account.
- **SAMAccountName:** (Active Directory Managed Systems only) The account SAM Account Name of an Active Directory account.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, default: 1) Maximum number of concurrent password requests for this account. A value of zero denotes unlimited requests.

- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
  - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
  - **CheckPasswordFlag:** True to enable password testing, otherwise false.
  - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
  - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
  - **ChangeFrequencyType:** The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month
    - **last:** Changes scheduled for the last day of the month
    - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
  - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
  - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
- **ParentAccountID:** If this is a subscribed account (**IsSubscribedAccount**), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 - Request successful. Managed Accounts in the response body.

PUT ManagedAccounts/{id}

Purpose

Updates an existing Managed Account by ID.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

id: ID of the Managed Account.

Request Body

Content-Type: application/json

```
{
AccountName : string,
ManagedSystemID: int,
Password : string,
PrivateKey : string,
Passphrase : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
    NextChangeDate : date-formatted string
}
```

Request Body Details

- **AccountName:** (required) The name of the account. Must be unique on the system.
- **ManagedSystemID:** (required) ID of the Managed System.
- **Password:** (required if **AutoManagementFlag** is false) The account password.
- **PrivateKey:** DSS Private Key. Can be set if **Platform.DSSFlag** is true.
- **Passphrase:** (required when **PrivateKey** is an encrypted DSS key) DSS Passphrase. Can be set if **Platform.DSSFlag** is true.
- **PasswordFallbackFlag:** (default: false) True if failed DSS authentication can fall back to password authentication, otherwise false. Can be set if **Platform.DSSFlag** is true.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false. Can be set when the **ManagedSystem.LoginAccountID** is set.
- **Description:** A description of the account.
- **PasswordRuleID:** (default: 0) ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** (default: false) True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** (default: false) True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** (default: false) True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.

- **ChangeTasksFlag:** (default: false) True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600, default: 120) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600, default: 525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600, default: 120) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 is unlimited, default: 1) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** (default: false) True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** (default: false) True if DSS Key auto-management is enabled, otherwise false. If set to true, and no **PrivateKey** is provided, immediately attempts to generate and set a new public key on the Server. Can be set if **Platform.DSSAutoManagementFlag** is true.
    - **CheckPasswordFlag:** (default: false) True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** (default: false) True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** (default: false) True to queue a password change when scheduled password test fails, otherwise false.
    - **ChangeFrequencyType:** (default: first) The change frequency for scheduled password changes:
        - **first:** Changes scheduled for the first day of the month
        - **last:** Changes scheduled for the last day of the month
        - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
    - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
    - **ChangeTime:** (24hr format: 00:00-23:59, default: 23:30) UTC time of day scheduled password changes take place.
    - **NextChangeDate:** (date format: YYYY-MM-DD) UTC date when next scheduled password change will occur. If the **NextChangeDate** + **ChangeTime** is in the past, password change will occur at the nearest future **ChangeTime**.

Response Body

Content-Type: application/json

```
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
}
```

Response Codes

200 - Request successful. Managed Accounts in the response body.

For more information, please see "Common Response Codes" on page 13.

POST ManagedSystems/{systemID}/ManagedAccounts

Purpose

Creates a new Managed Account in the Managed System referenced by ID.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

systemID: ID of the Managed System.

Request Body

Content-Type: application/json

```
{
AccountName : string,
Password : string,
DomainName : string,
UserPrincipalName : string,
SAMAccountName : string,
DistinguishedName : string,
PrivateKey : string,
Passphrase : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
    NextChangeDate : date-formatted string
}
```

Request Body Details

- **AccountName:** (required) The name of the account. Must be unique on the system.
- **Password:** (required if **AutoManagementFlag** is false) The account password.
- **DomainName:** (optional) This can be given but it must be exactly the same as the Directory. If empty or null, it will be automatically populated from the parent Managed System/Directory.
- **UserPrincipalName:** (required for Active Directory Managed Systems only) The Active Directory User Principal Name.
- **SAMAccountName:** (required for Active Directory Managed Systems only) The Active Directory SAM Account Name (Maximum 20 characters).
- **DistinguishedName:** (required for LDAP Directory managed systems only) The LDAP Distinguished Name.
- **PrivateKey:** DSS Private Key. Can be set if **Platform.DSSFlag** is true.
- **Passphrase:** (required when **PrivateKey** is an encrypted DSS key) DSS Passphrase. Can be set if **Platform.DSSFlag** is true.
- **PasswordFallbackFlag:** (default: false) True if failed DSS authentication can fall back to password authentication, otherwise false. Can be set if **Platform.DSSFlag** is true.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false. Can be set when the **ManagedSystem.LoginAccountID** is set.
- **Description:** A description of the account.
- **PasswordRuleID:** (default: 0) ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** (default: false) True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** (default: false) True if services run as this user should be updated with the new password after a password change, otherwise false.

- **RestartServicesFlag:** (default: false) True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** (default: false) True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600, default: 120) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600, default: 525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600, default: 120) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 is unlimited, default: 1) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** (default: false) True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** (default: false) True if DSS Key auto-management is enabled, otherwise false. If set to true, and no **PrivateKey** is provided, immediately attempts to generate and set a new public key on the server. Can be set if **Platform.DSSAutoManagementFlag** is true.
    - **CheckPasswordFlag:** (default: false) True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** (default: false) True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** (default: false) True to queue a password change when scheduled password test fails, otherwise false.
    - **ChangeFrequencyType:** (default: first) The change frequency for scheduled password changes:
        - **first:** Changes scheduled for the first day of the month
        - **last:** Changes scheduled for the last day of the month
        - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
    - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
    - **ChangeTime:** (24hr format: 00:00-23:59, default: 23:30) UTC time of day scheduled password changes take place.
    - **NextChangeDate:** (date format: YYYY-MM-DD) UTC date when next scheduled password change will occur. If the **NextChangeDate** + **ChangeTime** is in the past, password change will occur at the nearest future **ChangeTime**.

Response Body

Content-Type: application/json

```
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
UserPrincipalName : string,
SAMAccountName : string,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
}
```

Response Body Details

- **AccountName:** The name of the account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **UserPrincipalName:** (Active Directory Managed Systems only) The account User Principal Name of an Active Directory account.
- **SAMAccountName:** (Active Directory Managed Systems only) The account SAM Account Name of an Active Directory account.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 means unlimited) Maximum number of concurrent password requests for this account.

- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
  - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
  - **CheckPasswordFlag:** True to enable password testing, otherwise false.
  - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
  - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
  - **ChangeFrequencyType:** The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month
    - **last:** Changes scheduled for the last day of the month
    - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
  - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
  - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
- **ParentAccountID:** If this is a subscribed account, this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

> For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

201 - Request successful. Managed Account in the response body.

> For more information, please see "Common Response Codes" on page 13.

DELETE ManagedAccounts/{id}

Purpose

Deletes a Managed Account by ID.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

id: ID of the Managed Account.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

For more information, please see "Common Response Codes" on page 13.

DELETE ManagedSystems/{systemID}/ManagedAccounts/{accountName}

Purpose

Deletes a Managed Account by Managed System ID and Managed Account name.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

- **systemID:** ID of the Managed System.
- **accountName:** Name of the Managed Account.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

For more information, please see "Common Response Codes" on page 13.

Quick Rule Managed Accounts

Quick Navigation

- GET QuickRules/{quickRuleID}/ManagedAccounts
- PUT QuickRules/{quickRuleID}/ManagedAccounts
- DELETE QuickRules/{quickRuleID}/ManagedAccounts/{accountID}

GET QuickRules/{quickRuleID}/ManagedAccounts

Purpose

Returns a list of Managed Accounts by Quick Rule ID.

Required Permissions

Read access to the Quick Rule

URL Parameters

quickRuleID: ID of the Quick Rule.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,
AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
},
...
]
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 is unlimited) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.

- ○ **ChangeFrequencyType:** The change frequency for scheduled password changes:
  - **first:** Changes scheduled for the first day of the month
  - **last:** Changes scheduled for the last day of the month
  - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
- ○ **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
- ○ **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.

- **ParentAccountID:** If this is a subscribed account (**IsSubscribedAccount**), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 – Request successful. Managed Accounts in the response body.

For more information, please see "Common Response Codes" on page 13.

PUT QuickRules/{quickRuleID}/ManagedAccounts

Purpose

Updates the entire list of Managed Accounts in a Quick Rule by removing all **Managed Account Fields - Quick Group ID** filters and adding a new one with the Managed Accounts referenced by ID.

***Note:** If the Quick Rule contains complex filters or actions created via the UI, the rule must reprocess before returning. It is better for performance to use a Quick Rule that contains a single filter of type **Managed Account Fields - Quick Group ID** and a single action of type **Show as Smart Group**, as is created using **POST QuickRules**.*

Required Permissions

- Password Safe Account Management (Read)
- Read/Write access to the Quick Rule

URL Parameters

quickRuleID: ID of the Quick Rule.

Request Body

Content-Type: application/json

```
{
AccountIDs: [ int, …]
}
```

Response Body

Content-Type: application/json

```
[
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
    ParentAccountID : int, // can be null

IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
},
...
]
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 is unlimited) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
    - **ChangeFrequencyType:** The change frequency for scheduled password changes:
        - **first:** Changes scheduled for the first day of the month
        - **last:** Changes scheduled for the last day of the month
        - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
    - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
    - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
- **ParentAccountID:** If this is a subscribed account (**IsSubscribedAccount**), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 – Request successful. Managed Accounts in the response body.

> For more information, please see "Common Response Codes" on page 13.

POST QuickRules/{quickRuleID}/ManagedAccounts/{accountID}

Purpose

Adds the Managed Account referenced by ID to the Quick Rule by adding it to the first 'Managed Account Fields - Quick Group ID' filter found.

> ***Note:** If the Quick Rule contains complex filters or actions created via the UI, the rule must reprocess before returning. It is better for performance to use a Quick Rule that contains a single filter of type **Managed Account Fields - Quick Group ID** and a single action of type **Show as Smart Group**, as is created using **POST QuickRules**.*

Required Permissions

- Password Safe Account Management (Read)
- Read/Write access to the Quick Rule

URL Parameters

- **quickRuleID:** ID of the Quick Rule
- **accountID:** ID of the Managed Account

Request Body

None

Response Body

Content-Type: application/json

```
 [
 {
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
},
...
]
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.

- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 means unlimited) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
  - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
  - **CheckPasswordFlag:** True to enable password testing, otherwise false.
  - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
  - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
  - **ChangeFrequencyType:** The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month
    - **last:** Changes scheduled for the last day of the month
    - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
  - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
  - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
- **ParentAccountID:** If this is a subscribed account (**IsSubscribedAccount**), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 – Request successful. Managed Accounts in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE QuickRules/{quickRuleID}/ManagedAccounts/{accountID}

Purpose

Removes the Managed Account referenced by ID from the Quick Rule by removing it from all **Managed Account Fields - Quick Group ID** filters found.

IMPORTANT!

A rule cannot be left in an invalid state. If removing the account would result in an empty filter, the filter itself will be removed. If there are no filters left in the rule, a "400 Bad Request" is returned.

- *If you intend to replace all accounts in the rule, see "PUT QuickRules/{quickRuleID}/ManagedAccounts" on page 200.*
- *If you intend to delete the rule, see "DELETE QuickRules/{id}" on page 258.*

Note: *If the Quick Rule contains complex filters or actions created via the UI, the rule must reprocess before returning. It is better for performance to use a Quick Rule that contains a single filter of type* ***Managed Account Fields - Quick Group ID*** *and a single action of type* ***Show as Smart Group****, as is created using* ***POST QuickRules****.*

Required Permissions

- Read access to the Quick Rule

URL Parameters

- **quickRuleID:** ID of the Quick Rule
- **accountID:** ID of the Managed Account

Request Body

None

Response Body

Content-Type: application/json

```
 [
 {
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
},
...
]
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 means unlimited) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.

  - **ChangeFrequencyType:** The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month
    - **last:** Changes scheduled for the last day of the month
    - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
  - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
  - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
  - **ParentAccountID:** If this is a subscribed account (**IsSubscribedAccount**), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

> For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 – Request successful. Managed Accounts in the response body.

> For more information, please see "Common Response Codes" on page 13.

Smart Rule Managed Accounts

GET SmartRules/{smartRuleID}/ManagedAccounts

Purpose

Returns a list of Managed Accounts by Smart Rule ID.

Required Permissions

Read access to the Smart Rule

URL Parameters

smartRuleID: ID of the Smart Rule.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ChangeTasksFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
},
...
]
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ChangeTasksFlag:** True if scheduled tasks run as this user should be updated with the new password after a password change, otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 means unlimited) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
    - **ChangeFrequencyType:** The change frequency for scheduled password changes:
        - **first:** Changes scheduled for the first day of the month
        - **last:** Changes scheduled for the last day of the month
        - **xdays:** Changes scheduled every x days (ChangeFrequencyDays)
    - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this

configured number of days.
  - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
- **ParentAccountID:** If this is a subscribed account (IsSubscribedAccount), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

> ℹ For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 - Request successful. Managed Accounts in the response body.

> ℹ For more information, please see "Common Response Codes" on page 13.

Managed Systems

Quick Navigation

- GET ManagedSystems
- GET ManagedSystems/{id}
- GET Assets/{assetId}/ManagedSystems
- GET Databases/{databaseID}/ManagedSystems
- GET FunctionalAccounts/{id}/ManagedSystems
- GET Workgroups/{id}/ManagedSystems
- PUT ManagedSystems/{id}
- POST Databases/{databaseID}/ManagedSystems
- POST Databases/{databaseID}/ManagedSystems
- POST Workgroups/{id}/ManagedSystems
- DELETE ManagedSystems/{id}

For more information on related topics, please see:

- Assets
- Managed Accounts
- Password Rules
- DSS Key Rules
- Platforms

GET ManagedSystems

Purpose

Returns a list of Managed Systems.

Required Permissions

Password Safe System Management (Read)

Query Parameters (optional)

- **type:** The EntityType of the Managed System.
- **name:** The name of the Managed System.
- **limit:** (default: 100000) Number of records to return.
- **offset:** (default: 0) Number of records to skip before returning records (can only be used in conjunction with **limit**).

Request Body

None

Response Body (when limit is not given)

Content-Type: application/json

```
[
{
WorkgroupID : int

HostName : string

IPAddress : string
DNSName : string
InstanceName : string
IsDefaultInstance : bool // can be null
Template : string
ForestName : string
UseSSL : bool // can be null

ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
Timeout : short,
PlatformID: int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
},
…
]
```

Response Body (when limit is given)

Content-Type: application/json

```
{
TotalCount : int,
Data : [{
WorkgroupID : int

HostName : string

IPAddress : string
DNSName : string
InstanceName : string
IsDefaultInstance : bool // can be null
Template : string
ForestName : string
UseSSL : bool // can be null

ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
Timeout : short,
PlatformID: int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
       },
…]
}
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

200 - Request successful. Managed Systems in response body.

For more information, please see "Common Response Codes" on page 13.

GET ManagedSystems/{id}

Purpose

Returns a Managed System by ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

id: ID of the Managed System.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
WorkgroupID : int

HostName : string

IPAddress : string
DNSName : string
InstanceName : string
IsDefaultInstance : bool // can be null
Template : string
ForestName : string
UseSSL : bool // can be null

ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
Timeout : short,
PlatformID: int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
},
…
]
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

200 - Request successful. Managed System in response body.

For more information, please see "Common Response Codes" on page 13.

GET Assets/{assetId}/ManagedSystems

Purpose

Returns a Managed System for the Asset referenced by ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

assetId: ID of the Asset.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
WorkgroupID : int

HostName : string

IPAddress : string
DNSName : string
InstanceName : string
IsDefaultInstance : bool // can be null
Template : string
ForestName : string
UseSSL : bool // can be null

ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
Timeout : short,
PlatformID: int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
},
…
]
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

200 - Request successful. Managed System in response body.

For more information, please see "Common Response Codes" on page 13.

GET Databases/{databaseID}/ManagedSystems

Purpose

Returns a Managed System for the Database referenced by ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

databaseID: ID of the Database.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
WorkgroupID : int

HostName : string

IPAddress : string
DNSName : string
InstanceName : string
IsDefaultInstance : bool // can be null
Template : string
ForestName : string
UseSSL : bool // can be null

ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
Timeout : short,
PlatformID: int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
},
…
]
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

200 - Request successful. Managed System in response body.

For more information, please see "Common Response Codes" on page 13.

GET FunctionalAccounts/{id}/ManagedSystems

Purpose

Returns a list of Managed Systems auto-managed by the Functional Account referenced by ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

id: ID of the Functional Account.

Query Parameters (optional)

- **type:** The EntityType of the Managed System.
- **name:** The name of the Managed System.
- **limit:** (default: 100000) Number of records to return.
- **offset:** (default: 0) Number of records to skip before returning records (can only be used in conjunction with limit).

Request Body

None

Response Body (when limit is not given)

Content-Type: application/json

```
[
{
```

```
WorkgroupID : int

HostName : string

IPAddress : string
DNSName : string
InstanceName : string
IsDefaultInstance : bool // can be null
Template : string
ForestName : string
UseSSL : bool // can be null

ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
Timeout : short,
PlatformID: int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
},
…
]
```

Response Body (when limit is given)

Content-Type: application/json

```
{
TotalCount : int,
Data : [{
WorkgroupID : int

HostName : string

IPAddress : string
DNSName : string
InstanceName : string
IsDefaultInstance : bool // can be null
Template : string
ForestName : string
UseSSL : bool // can be null

ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
Timeout : short,
PlatformID: int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
       },
…]
}
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

200 - Request successful. Managed Systems in response body.

For more information, please see "Common Response Codes" on page 13.

GET Workgroups/{id}/ManagedSystems

Purpose

Returns a list of Managed Systems by Workgroup ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

id: ID of the Workgroup.

Query Parameters (optional)

- **limit:** (default: 100000) Number of records to return.
- **offset:** (default: 0) Number of records to skip before returning <limit> records (can be used only in conjunction with limit).

Request Body

None

Response Body (when limit is not given)

Content-Type: application/json

```
 [{
WorkgroupID : int,
ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
HostName : string,
IPAddress : string,
DnsName : string,
InstanceName : string,
IsDefaultInstance : bool, // can be null
Template : string,
ForestName : string,
UseSSL : bool, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,

SystemName : string,
PlatformID : int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AutoManagementFlag : bool,
       FunctionalAccountID : int, // can be null
       ElevationCommand : string, // can be null
       CheckPasswordFlag : bool,
       ChangePasswordAfterAnyReleaseFlag : bool,
       ResetPasswordOnMismatchFlag : bool,
       ChangeFrequencyType : string,
       ChangeFrequencyDays : int,
       ChangeTime : string,
},
…]
```

Response Body (when limit is given)

Content-Type: application/json

```
{
TotalCount : int,
Data :
       [{
       WorkgroupID : int,
       ManagedSystemID : int,
       EntityTypeID: int,
       AssetID : int, // can be null
       DatabaseID : int, // can be null
       DirectoryID : int, // can be null
       CloudID : int, // can be null
```

```
        HostName : string,
        IPAddress : string,
        DnsName : string,
        InstanceName : string,
        IsDefaultInstance : bool, // can be null
        Template : string,
        ForestName : string,
        UseSSL : bool, // can be null
        AccountNameFormat : int,
        OracleInternetDirectoryID : guid, // can be null
        OracleInternetDirectoryServiceName : string,

        SystemName : string,
        PlatformID : int,
        NetBiosName : string,
        ContactEmail : string,
        Description : string,
        Port : int, // can be null
        Timeout : short,
        PasswordRuleID : int,
        DSSKeyRuleID : int, // can be null
        LoginAccountID : int, // can be null
        ReleaseDuration : int,
        MaxReleaseDuration : int,
        ISAReleaseDuration : int,
        AutoManagementFlag : bool,
                FunctionalAccountID : int, // can be null
                ElevationCommand : string, // can be null
                CheckPasswordFlag : bool,
                ChangePasswordAfterAnyReleaseFlag : bool,
                ResetPasswordOnMismatchFlag : bool,
                ChangeFrequencyType : string,
                ChangeFrequencyDays : int,
                ChangeTime : string,
        },
        …]
}
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

200 - Request successful. Managed Systems in response body.

For more information, please see "Common Response Codes" on page 13.

PUT ManagedSystems/{id}

Purpose

Updates an existing Managed System by ID.

> ***Note:** PUT ManagedSystems/{id} supports all Managed System Types: Dynamic Asset, Static Asset, Dynamic Database, Static Database, Directory, and Cloud.*

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

id: ID of the Managed System.

Request Body

Content-Type: application/json

```
{
WorkgroupID : int,

HostName : string,

IPAddress : string,
DnsName : string,
InstanceName : string,
IsDefaultInstance : bool, // can be null
Template : string,
ForestName : string,
UseSSL : bool, // can be null

PlatformID : int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AutoManagementFlag : bool,
       FunctionalAccountID : int, // can be null
       ElevationCommand : string, // can be null
       CheckPasswordFlag : bool,
       ChangePasswordAfterAnyReleaseFlag : bool,
       ResetPasswordOnMismatchFlag : bool,
       ChangeFrequencyType : string,
       ChangeFrequencyDays : int,
       ChangeTime : string,
}
```

Request Body Details

- **WorkgroupID:** ID of the Workgroup.
- **HostName:** (required) Name of the host (applies to Static Asset, Static Database, Directory, Cloud).
  - **Static Asset:** Asset Name.
  - **Static Database:** Database Host Name.
  - **Directory:** Directory/Domain Name.
  - **Cloud:** Cloud System Name.
- **IPAddress:** IPv4 address of the host (applies to Static Asset, Static Database).
- **DnsName:** DNS name of the host (applies to Static Asset, Static Database).
- **InstanceName:** Name of the database instance. Required when **IsDefaultInstance** is false (applies to Static Database only).
- **IsDefaultInstance:** True if the database instance is the default instance, otherwise false. Only Platforms MS SQL Server and MySQL support setting this value to true (applies to Static Database only).
- **Template:** The database connection template (applies to Static Database only).
- **ForestName:** Name of the Directory Forest (applies to Directory only).
- **UseSSL** (default: false) True to use an SSL connection, otherwise false (applies to Directory only).
- **PlatformID:** (required) ID of the Managed System Platform.
- **NetBiosName:** The NetBIOS name of the host. Can be set if **Platform.NetBiosNameFlag** is true.
- **Port:** (optional) The port used to connect to the host. If null and the related **Platform.PortFlag** is true, Password Safe uses **Platform.DefaultPort** for communication.
- **Timeout:** (seconds, default: 30) Connection timeout. Length of time in seconds before a slow or unresponsive connection to the system fails.
- **SshKeyEnforcementMode:** (default: 0/None) Enforcement mode for SSH host keys.
  - **0:** None.
  - **1:** Auto - Auto Accept Initial Key.
  - **2:** Strict - Manually Accept Keys.
- **PasswordRuleID:** (default: 0) ID of the default Password Rule assigned to Managed Accounts created under this Managed System.
- **DSSKeyRuleID:** (default: 0) ID of the default DSS Key Rule assigned to Managed Accounts created under this Managed System. Can be set when **Platform.DSSFlag** is true.
- **LoginAccountID:** (optional) ID of the Functional Account used for SSH Session logins. Can be set if the **Platform.LoginAccountFlag** is true.
- **ReleaseDuration:** (minutes: 1-525600, default: 120) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600, default: 525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600, default: 120) Default Information Systems Administrator (ISA) release duration.
- **AutoManagementFlag:** (default: false) True if password auto-management is enabled, otherwise false. Can be set if **Platform.AutoManagementFlag** is true.

  - ◦ **FunctionalAccountID:** (required if **AutoManagementFlag** is true) ID of the Functional Account used for local Managed Account password changes. **FunctionalAccount.PlatformID** must either match the **ManagedSystem.PlatformID** or be a Directory Platform (AD, LDAP).
  - ◦ **ElevationCommand:** (optional) Elevation Command to use. Can be set if **Platform.SupportsElevationFlag** is true.
    - ▪ **sudo**
      - **pbrun**
      - **pmrun**

- **CheckPasswordFlag:** True to enable password testing, otherwise false.
- **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
- **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.

Response Body

Content-Type: application/json

```
{
WorkgroupID : int,
ManagedSystemID : int,
EntityTypeID: int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null

HostName : string,
IPAddress : string,
DnsName : string,
InstanceName : string,
IsDefaultInstance : bool, // can be null
Template : string,
ForestName : string,
UseSSL : bool, // can be null

SystemName : string,
PlatformID : int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AutoManagementFlag : bool,
       FunctionalAccountID : int, // can be null
       ElevationCommand : string, // can be null
       CheckPasswordFlag : bool,
       ChangePasswordAfterAnyReleaseFlag : bool,
       ResetPasswordOnMismatchFlag : bool,
       ChangeFrequencyType : string,
       ChangeFrequencyDays : int,
       ChangeTime : string,
}
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

200 - Request successful. Managed System in response body.

For more information, please see "Common Response Codes" on page 13.

POST Assets/{assetId}/ManagedSystems

Purpose

Creates a Managed System for the Asset referenced by ID.

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

assetId: ID of the Asset.

Request Body

Content-Type: application/json

```
{
PlatformID : int,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
}
```

Request Body Details

- **PlatformID:**(required) ID of the Managed System Platform.
- **Port:** (optional) The port used to connect to the host. If null and the related **Platform.PortFlag** is true, Password Safe uses **Platform.DefaultPort** for communication.
- **Timeout:** (seconds, default: 30) Connection timeout. Length of time in seconds before a slow or unresponsive connection to the system fails.
- **SshKeyEnforcementMode:** (default: 0/None) Enforcement mode for SSH host keys.
  - **0:** None.
  - **1:** Auto - Auto Accept Initial Key.
  - **2:** Strict - Manually Accept Keys.
- **PasswordRuleID:** (default: 0) ID of the default Password Rule assigned to Managed Accounts created under this Managed System.
- **DSSKeyRuleID:** (default: 0) ID of the default DSS Key Rule assigned to Managed Accounts created under this Managed System. Can be set when **Platform.DSSFlag** is true.
- **LoginAccountID:** (optional) ID of the Functional Account used for SSH Session logins. Can be set if the **Platform.LoginAccountFlag** is true.
- **ReleaseDuration:** (minutes: 1-525600, default: 120) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600, default: 525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600, default: 120) Default Information Systems Administrator (ISA) release duration.
- **AutoManagementFlag:** (default: false) True if password auto-management is enabled, otherwise false. Can be set if **Platform.AutoManagementFlag** is true.
  - **FunctionalAccountID:** (required if **AutoManagementFlag** is true) ID of the Functional Account used for local Managed Account password changes. **FunctionalAccount.PlatformID** must either match the **ManagedSystem.PlatformID** or be a Domain Platform (AD, LDAP).
  - **ElevationCommand:** (optional) Elevation Command to use. Can be set if **Platform.SupportsElevationFlag** is true (**sudo**, **pbrun**, **pmrun**).
  - **CheckPasswordFlag:** True to enable password testing, otherwise false.
  - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
  - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.

- **ChangeFrequencyType:** (default: first) The change frequency for scheduled password changes:
    - **first:** Changes scheduled for the first day of the month.
    - **last:** Changes scheduled for the last day of the month.
    - **xdays:** Changes scheduled every x days (ChangeFrequencyDays).
- **ChangeFrequencyDays:** (days: 1-999, required if **ChangeFrequencyType** is **xdays**) When ChangeFrequencyType is **xdays**, password changes take place this configured number of days.
- **ChangeTime:** (24hr format: 00:00-23:59, default: 23:30) UTC time of day scheduled password changes take place.

Response Body

Content-Type: application/json

```
[
{
WorkgroupID : int

HostName : string

IPAddress : string
DNSName : string
InstanceName : string
IsDefaultInstance : bool // can be null
Template : string
ForestName : string
UseSSL : bool // can be null

ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
Timeout : short,
PlatformID: int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
},
…
]
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

- 200 - Request successful - Asset was already managed. Managed System in response body.
- 201 - Request successful - Asset is now managed. Managed System in response body.

For more information, please see "Common Response Codes" on page 13.

POST Databases/{databaseID}/ManagedSystems

Purpose

Creates a Managed System for the Database referenced by ID.

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

databaseID: ID of the Database.

Request Body

Content-Type: application/json

```
{
ContactEmail : string,
Description : string,
Timeout : short,
PasswordRuleID : int,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
}
```

Request Body Details

- **Timeout:** (seconds, default: 30) Connection timeout. Length of time in seconds before a slow or unresponsive connection to the system fails.
- **PasswordRuleID:** (default: 0) ID of the default Password Rule assigned to Managed Accounts created under this Managed System.
- **ReleaseDuration:** (minutes: 1-525600, default: 120) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600, default: 525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600, default: 120) Default Information Systems Administrator (ISA) release duration.
- **AutoManagementFlag:** (default: false) True if password auto-management is enabled, otherwise false. Can be set if **Platform.AutoManagementFlag** is true.
    - **FunctionalAccountID:** (required if **AutoManagementFlag** is true) ID of the Functional Account used for local Managed Account password changes. **FunctionalAccount.PlatformID** must either match the **ManagedSystem.PlatformID** or be a Domain Platform (AD, LDAP).
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
    - **ChangeFrequencyType:** (default: first) The change frequency for scheduled password changes:
        - **first:** Changes scheduled for the first day of the month.
        - **last:** Changes scheduled for the last day of the month.
        - **xdays:** Changes scheduled every x days (ChangeFrequencyDays).
    - **ChangeFrequencyDays:** (days: 1-999, required if **ChangeFrequencyType** is **xdays**) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
    - **ChangeTime:** (24hr format: 00:00-23:59, default: 23:30) UTC time of day scheduled password changes take place.

Response Body

Content-Type: application/json

```
[
{
WorkgroupID : int
HostName : string

IPAddress : string
DNSName : string
InstanceName : string
IsDefaultInstance : bool // can be null
Template : string
ForestName : string
UseSSL : bool // can be null

ManagedSystemID : int,
EntityTypeID: int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
Timeout : short,
PlatformID: int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,

AutoManagementFlag : bool,
    FunctionalAccountID : int, // can be null
    ElevationCommand : string, // can be null
    CheckPasswordFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,
},
…
]
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

- 200 - Request successful - Asset was already managed. Managed System in response body.
- 201 - Request successful - Asset is now managed. Managed System in response body.

For more information, please see "Common Response Codes" on page 13.

POST Workgroups/{id}/ManagedSystems

Purpose

Creates a Managed System in the Workgroup referenced by ID.

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

id: ID of the Workgroup.

Request Body

Content-Type: application/json

```
{
EntityTypeID : int,

HostName : string,
IPAddress : string,
DnsName : string,
InstanceName : string,
IsDefaultInstance : bool, // can be null
Template : string,
ForestName : string,
UseSSL : bool, // can be null

PlatformID : int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AutoManagementFlag : bool,
       FunctionalAccountID : int, // can be null
       ElevationCommand : string, // can be null
       CheckPasswordFlag : bool,
       ChangePasswordAfterAnyReleaseFlag : bool,
       ResetPasswordOnMismatchFlag : bool,
       ChangeFrequencyType : string,
       ChangeFrequencyDays : int,
       ChangeTime : string,
}
```

Request Body Details

- **EntityTypeID:** (required) Type of the Entity being created.
- **HostName:** (required) Name of the host (applies to Static Asset, Static Database, Directory, Cloud).
  - **Static Asset:** Asset Name
  - **Static Database:** Database Host Name
  - **Directory:** Directory/Domain Name
  - **Cloud:** Cloud System Name
- **IPAddress:** IPv4 address of the host (applies to Static Asset, Static Database).
- **DnsName:** DNS name of the host (applies to Static Asset, Static Database).
- **InstanceName:** Name of the database instance. Required when **IsDefaultInstance** is false (applies to Static Database only).
- **IsDefaultInstance:** True if the database instance is the default instance, otherwise false. Only Platforms MS SQL Server and MySQL support setting this value to true (applies to Static Database only).
- **Template:** The database connection template (applies to Static Database only).
- **ForestName:** Name of the Directory Forest (applies to Directory only).
- **UseSSL** (default: false) True to use an SSL connection, otherwise false (applies to Directory only).
- **PlatformID:** (required) ID of the Managed System Platform.
- **NetBiosName:** The NetBIOS name of the host. Can be set if **Platform.NetBiosNameFlag** is true.
- **Port:** (optional) The port used to connect to the host. If null and the related **Platform.PortFlag** is true, Password Safe uses **Platform.DefaultPort** for communication.
- **Timeout:** (seconds, default: 30) Connection timeout. Length of time in seconds before a slow or unresponsive connection to the system fails.
- **SshKeyEnforcementMode:** (default: 0/None) Enforcement mode for SSH host keys
  - **0:** None
  - **1:** Auto - Auto Accept Initial Key
  - **2:** Strict - Manually Accept Keys
- **PasswordRuleID:** (default: 0) ID of the default Password Rule assigned to Managed Accounts created under this Managed System.
- **DSSKeyRuleID:** (default: 0) ID of the default DSS Key Rule assigned to Managed Accounts created under this Managed System. Can be set when **Platform.DSSFlag** is true.
- **LoginAccountID:** (optional) ID of the Functional Account used for SSH Session logins. Can be set if the **Platform.LoginAccountFlag** is true.

- **AccountNameFormat:** (Active Directory only, default: 0) – Account Name format to use:
    - **0:** Domain and Account - Use **ManagedAccount.DomainName\ManagedAccount.AccountName**
    - **1:** UPN – Use the Managed Account UPN
    - **2:** SAM – Use the Managed Account SAM Account Name
- **OracleInternetDirectoryID:** The Oracle Internet Directory ID (applies to Database Entity Types and Oracle platform only).
- **OracleInternetDirectoryServiceName:** (required when **OracleInternetDirectoryID** is set) The Database Service Name related to the given **OracleInternetDirectoryID** (applies to Database Entity Types and Oracle platform only).
- **ReleaseDuration:** (minutes: 1-525600, default: 120) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600, default: 525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600, default: 120) Default Information Systems Administrator (ISA) release duration.
- **AutoManagementFlag:** (default: false) True if password auto-management is enabled, otherwise false. Can be set if **Platform.AutoManagementFlag** is true.
    - **FunctionalAccountID:** (required if **AutoManagementFlag** is true) ID of the Functional Account used for local Managed Account password changes. **FunctionalAccount.PlatformID** must either match the **ManagedSystem.PlatformID** or be a Directory Platform (AD, LDAP).
    - **ElevationCommand:** (optional) Elevation Command to use. Can be set if **Platform.SupportsElevationFlag** is true
        - **sudo**
        - **pbrun**
        - **pmrun**
- **CheckPasswordFlag:** True to enable password testing, otherwise false.
- **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
- **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.

Response Body (when limit is not given)

Content-Type: application/json

```
{
WorkgroupID : int,
ManagedSystemID : int,
EntityTypeID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null

HostName : string,
IPAddress : string,
DnsName : string,
InstanceName : string,
IsDefaultInstance : bool, // can be null
Template : string,
ForestName : string,
UseSSL : bool, // can be null
AccountNameFormat : int,

SystemName : string,
PlatformID : int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
AccountNameFormat : int,
OracleInternetDirectoryID : guid, // can be null
OracleInternetDirectoryServiceName : string,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AutoManagementFlag : bool,
       FunctionalAccountID : int, // can be null
       ElevationCommand : string, // can be null
       CheckPasswordFlag : bool,
       ChangePasswordAfterAnyReleaseFlag : bool,
       ResetPasswordOnMismatchFlag : bool,
       ChangeFrequencyType : string,
       ChangeFrequencyDays : int,
       ChangeTime : string,
}
```

Response Body Details

For more detailed information about the **Response Body**, please see Response Body Details.

Response Codes

200 - Request successful. Managed System in response body.

For more information, please see "Common Response Codes" on page 13.

DELETE ManagedSystems/{id}

Purpose

Deletes a Managed System by ID.

***Note: DELETE ManagedSystems/{id}** supports all Managed System Types: Dynamic Asset, Static Asset, Dynamic Database, Static Database, Directory, and Cloud.*

Required Permissions

Password Safe System Management (Read/Write)

URL Parameters

id: ID of the ManagedSystem.

Request Body

None

Response Body

None

Response Codes

200 - Request successful.

ℹ For more information, please see "Common Response Codes" on page 13.

Response Body Details

- **WorkgroupID:** ID of the Workgroup.
- **HostName:** (required) Name of the host (applies to Static Asset, Static Database, Directory, Cloud).
  - **Static Asset:** Asset Name.
  - **Static Database:** Database Host Name.
  - **Directory:** Directory/Domain Name.
  - **Cloud:** Cloud System Name.
- **IPAddress:** IPv4 address of the host (applies to Static Asset, Static Database).
- **DnsName:** DNS name of the host (applies to Static Asset, Static Database).
- **InstanceName:** Name of the database instance. Required when **IsDefaultInstance** is false (applies to Static Database only).
- **IsDefaultInstance:** True if the database instance is the default instance, otherwise false. Only Platforms MS SQL Server and MySQL support setting this value to true (applies to Static Database only).
- **Template:** The database connection template (applies to Static Database only).
- **ForestName:** Name of the Directory Forest (applies to Directory only).
- **UseSSL** (default: false) True to use an SSL connection, otherwise false (applies to Directory only).
- **ManagedSystemID:** ID of the Managed System.
- **AssetID:** Asset ID; set if the Managed System is an Asset or a Database.
- **DatabaseID:** Database ID; set if the Managed System is a Database.
- **DirectoryID:** Directory ID; set if the Managed System is a Directory.
- **CloudID:** Cloud System ID; set if the Managed System is a Cloud System.
- **EntitTypeID:** The Managed System Entity Type.
- **SystemName:** Name of the related entity (Asset, Directory, Database, or Cloud).
- **PlatformID:** ID of the Managed System Platform.

- **NetBiosName:** (Managed Domains only) Domain NetBIOS name. Setting this value will allow Password Safe to fall back to the NetBIOS name if needed.
- **Port:** The port used to connect to the host. If null and the related **Platform.PortFlag** is true, Password Safe uses **Platform.DefaultPort** for communication.
- **Timeout:** (seconds) Connection timeout. Length of time in seconds before a slow or unresponsive connection to the system fails.
- **SshKeyEnforcementMode:** Enforcement mode for SSH host keys
    - **0:** None.
    - **1:** Auto - Auto Accept Initial Key.
    - **2:** Strict - Manually Accept Keys.
- **PasswordRuleID:** ID of the default Password Rule assigned to Managed Accounts created under this Managed System.
- **DSSKeyRuleID:** ID of the default DSS Key Rule assigned to Managed Accounts created under this Managed System.
- **LoginAccountID:** ID of the Functional Account used for SSH Session logins.
- **AccountNameFormat:** (ActiveDirectory only, default: 0) Account Name format to use:
    - **0:** Domain and Account. Use **ManagedAccount.DomainName\ManagedAccount.AccountName**.
    - **1:** UPN. Use the Managed Account UPN.
    - **2:** SAM. Use the Managed Account SAM Account Name.
- **OracleInternetDirectoryID:** The Oracle Internet Directory ID (applies to Database Entity Types and Oracle platform only)
- **OracleInternetDirectoryServiceName:** (required when **OracleInternetDirectoryID** is set) The Database Service Name related to the given **OracleInternetDirectoryID** (applies to Database Entity Types and Oracle platform only)
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
    - **FunctionalAccountID:** ID of the Functional Account used for local Managed Account password changes.
    - **ElevationCommand:** Elevation Command to use (**sudo**, **pbrun**, **pmrun**).
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
    - **ChangeFrequencyType:** The change frequency for scheduled password changes:
        - **first:** Changes scheduled for the first day of the month.
        - **last:** Changes scheduled for the last day of the month.
        - **xdays:** Changes scheduled every x days (ChangeFrequencyDays).
    - **ChangeFrequencyDays:** (days: 1-999) When ChangeFrequencyType is **xdays**, password changes take place this configured number of days.
    - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.

Smart Rule Managed Systems

GET SmartRules/{id}/ManagedSystems

Purpose

Returns a list of Managed Systems by Smart Rule ID.

Required Permissions

- Read access to the smart rule referenced by ID.

URL Parameters

id: ID of the smart rule.

Query Parameters (optional)

- **limit:** (default: 100000) Number of records to return.
- **offset:** (default: 0) Number of records to skip before returning <limit> records (can be used only in conjunction with limit).

Request Body

None

Response Body (when limit is not given)

Content-Type: application/json

```
 [
 {
ManagedSystemID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
PlatformID : int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
SshKeyEnforcementMode : int, // can be null
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AutoManagementFlag : bool,
       FunctionalAccountID : int, // can be null
       ElevationCommand : string, // can be null
       CheckPasswordFlag : bool,
       ChangePasswordAfterAnyReleaseFlag : bool,
       ResetPasswordOnMismatchFlag : bool,
       ChangeFrequencyType : string,
       ChangeFrequencyDays : int,
       ChangeTime : string,
},
…]
```

Response Body (when limit is given)

Content-Type: application/json

```
{
TotalCount : int,
Data :
[{
ManagedSystemID : int,
AssetID : int, // can be null
DatabaseID : int, // can be null
DirectoryID : int, // can be null
CloudID : int, // can be null
SystemName : string,
PlatformID : int,
NetBiosName : string,
ContactEmail : string,
Description : string,
Port : int, // can be null
Timeout : short,
PasswordRuleID : int,
DSSKeyRuleID : int, // can be null
LoginAccountID : int, // can be null
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
AutoManagementFlag : bool,
       FunctionalAccountID : int, // can be null
       ElevationCommand : string, // can be null
       CheckPasswordFlag : bool,
       ChangePasswordAfterAnyReleaseFlag : bool,
       ResetPasswordOnMismatchFlag : bool,
       ChangeFrequencyType : string,
       ChangeFrequencyDays : int,
       ChangeTime : string,
},
…]
}
```

Response Codes

200 - Request successful. Managed Systems in response body.

For more information, please see "Common Response Codes" on page 13.

Nodes

Nodes represent the Session Monitoring Agent Nodes that can be used for establishing Sessions.

For more information on related topics, please see Sessions.

GET Nodes

Purpose

Returns a list of Session Monitoring Agent Nodes.

Query Parameters

includeInactive: (optional, default: false) **True** to return all nodes including nodes that are inactive, otherwise **False**.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
NodeID: string,
HostName: string,
DisplayName: string,
LastHeartbeat: DateTime, // can be null
IsActive: bool,
},
…
]
```

Response Body Details

- **NodeID:** Node unique ID.
- **HostName:** Node host name.
- **DisplayName:** Node display name.
- **LastHeartbeat:** The date and time of the last Session Monitoring Agent heartbeat from this Node.
- **IsActive:** True if the Session Monitoring Agent is considered active and running, otherwise false.

Response Codes

200 - Request successful. Nodes in the response body.

For more information, please see "Common Response Codes" on page 13.

Password Policies

***Note:** Password Policies are formerly known as Password Rules but the API remains **PasswordRules** to be compatible with earlier versions.*

Quick Navigation

- GET PasswordRules
- GET PasswordRules/{id}

GET PasswordRules

Purpose

Returns a list of Password Rules.

Required Permissions

Password Safe System Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
PasswordRuleID: int,
Name: string,
Description: string,
MinimumLength: int,
MaximumLength: int,
FirstCharacterRequirement: char,
LowercaseRequirement: char,
UppercaseRequirement: char,
NumericRequirement: char,
SymbolRequirement: char,
ValidLowercaseCharacters: char[],
ValidUppercaseCharacters: char[],
ValidSymbols: char[],
},
…
]
```

Response Body Details

- **FirstCharacterRequirement:** The first character of the password must be:
  - **C:** Characters (alpha) only
  - **N:** Numeric permitted, in addition to alpha characters
  - **A:** Any character permitted
- **LowercaseRequirement:** Lowercase character requirements:
- **UppercaseRequirement:** Uppercase character requirements:
- **NumericRequirement:** Numeric requirements:
- **SymbolRequirement:** Symbol requirements:
  - **N:** Not permitted
  - **P:** Permitted, not required
  - **R:** Required

Response Codes

200 - Request successful. Password Rules in the response body.

For more information, please see "Common Response Codes" on page 13.

GET PasswordRules/{id}

Purpose

Returns a Password Rule by ID.

Required Permissions

Password Safe System Management (Read)

URL Parameters

id: ID of the Password Rule.

Request Body

None

Response Body

Content-Type: application/json

```
{
PasswordRuleID: int,
Name: string,
Description: string,
MinimumLength: int,
MaximumLength: int,
FirstCharacterRequirement: char,
LowercaseRequirement: char,
UppercaseRequirement: char,
NumericRequirement: char,
SymbolRequirement: char,
ValidLowercaseCharacters: char[],
ValidUppercaseCharacters: char[],
ValidSymbols: char[],
}
```

Response Body Details

- **FirstCharacterRequirement:** The first character of the password must be:
  - **C:** Characters (alpha) only
  - **N:** Numeric permitted, in addition to alpha characters
  - **A:** Any character permitted
- **LowercaseRequirement:** Lowercase character requirements:
- **UppercaseRequirement:** Uppercase character requirements:
- **NumericRequirement:** Numeric requirements:
- **SymbolRequirement:** Symbol requirements:
  - **N:** Not permitted
  - **P:** Permitted, not required
  - **R:** Required

Response Codes

200 - Request successful. Password Rules in the response body.

For more information, please see "Common Response Codes" on page 13.

Platforms

Quick Navigation

- GET Platforms
- GET Platforms/{id}
- GET EntityTypes/{id}/Platforms

For more information on related topics, please see "Entity Types" on page 155

GET Platforms

Purpose

Returns a list of Platforms for Managed Systems.

Required Permissions

None

Request Body

None

Response Body

Content-Type: application/json

```
[
{
PlatformID : int,
Name : string,
ShortName : string,
PortFlag : bool,
DefaultPort: int, // can be null
SupportsElevationFlag : bool,
DomainNameFlag: bool,
AutoManagementFlag: bool,
DSSAutoManagementFlag: bool,
ManageableFlag: bool,
DSSFlag: bool,
LoginAccountFlag : bool,
DefaultSessionType: string // can be null
}
…
]
```

Response Body Details

- **PlatformID:** Platform ID.
- **Name:** Platform name.
- **ShortName:** Platform short name.
- **PortFlag:** True if the platform supports setting a port, otherwise false.
- **DefaultPort:** The default port used when no port is given for Managed Systems of this platform.
- **DomainNameFlag:** True if the platform supports setting a Domain Name on a Functional Account of this platform, otherwise false.
- **SupportsElevationFlag:** True if the platform supports elevation, otherwise false.
- **AutoManagementFlag:** True if the platform supports password auto-management, otherwise false.
- **DSSAutoManagementFlag:** True if the platform supports DSS key auto-management, otherwise false.
- **ManageableFlag:** True if Functional Accounts can be created for the platform, otherwise false.
- **DSSFlag:** True if the platform supports DSS Keys, otherwise false.
- **LoginAccountFlag:** True if the platform supports SSH Login Accounts, otherwise false.
- **DefaultSessionType:** The default type of session for the platform (RDP, SSH, or null).

Response Codes

200 – Request successful. Platforms in response body.

For more information, please see "Common Response Codes" on page 13.

GET Platforms/{id}

Purpose

Returns a Platform by ID for Managed Systems.

Required Permissions

None

URL Parameters

id: ID of the Platform.

Request Body

None

Response Body

Content-Type: application/json

```
{
PlatformID : int,
Name : string,
ShortName : string,
PortFlag : bool,
DefaultPort: int, // can be null
SupportsElevationFlag : bool,
DomainNameFlag: bool,
AutoManagementFlag: bool,
DSSAutoManagementFlag: bool,
ManageableFlag: bool,
DSSFlag: bool,
LoginAccountFlag : bool,
DefaultSessionType: string // can be null
}
```

Response Body Details

- **PlatformID:** Platform ID.
- **Name:** Platform name.
- **ShortName:** Platform short name.
- **PortFlag:** True if the platform supports setting a port, otherwise false.
- **DefaultPort:** The default port used when no port is given for Managed Systems of this platform.
- **DomainNameFlag:** True if the platform supports setting a Domain Name on a Functional Account of this platform, otherwise false.
- **SupportsElevationFlag:** True if the platform supports elevation, otherwise false.
- **AutoManagementFlag:** True if the platform supports password auto-management, otherwise false.
- **DSSAutoManagementFlag:** True if the platform supports DSS key auto-management, otherwise false.
- **ManageableFlag:** True if Functional Accounts can be created for the platform, otherwise false.
- **DSSFlag:** True if the platform supports DSS Keys, otherwise false.
- **LoginAccountFlag:** True if the platform supports SSH Login Accounts, otherwise false.
- **DefaultSessionType:** The default type of session for the platform (RDP, SSH, or null).

Response Codes

200 – Request successful. Platform in response body.

For more information, please see "Common Response Codes" on page 13.

GET EntityTypes/{id}/Platforms

Purpose

Returns a list of Platforms by Entity Type ID.

Required Permissions

None

URL Parameters

id: ID of the Entity Type.

Request Body

None

Response Body

Content-Type: application/json

```
{
PlatformID : int,
Name : string,
ShortName : string,
PortFlag : bool,
DefaultPort: int, // can be null
SupportsElevationFlag : bool,
DomainNameFlag: bool,
AutoManagementFlag: bool,
DSSAutoManagementFlag: bool,
ManageableFlag: bool,
DSSFlag: bool,
LoginAccountFlag : bool,
DefaultSessionType: string // can be null
}
```

Response Body Details

- **PlatformID:** Platform ID.
- **Name:** Platform name.
- **ShortName:** Platform short name.
- **PortFlag:** True if the platform supports setting a port, otherwise false.
- **DefaultPort:** The default port used when no port is given for Managed Systems of this platform.
- **DomainNameFlag:** True if the platform supports setting a Domain Name on a Functional Account of this platform, otherwise false.
- **SupportsElevationFlag:** True if the platform supports elevation, otherwise false.
- **AutoManagementFlag:** True if the platform supports password auto-management, otherwise false.
- **DSSAutoManagementFlag:** True if the platform supports DSS key auto-management, otherwise false.
- **ManageableFlag:** True if Functional Accounts can be created for the platform, otherwise false.
- **DSSFlag:** True if the platform supports DSS Keys, otherwise false.
- **LoginAccountFlag:** True if the platform supports SSH Login Accounts, otherwise false.
- **DefaultSessionType:** The default type of session for the platform (RDP, SSH, or null).

Response Codes

200 – Request successful. Platform in response body.

For more information, please see "Common Response Codes" on page 13.

Quick Rules

Quick Rules are a specialized Smart Rule for building a list of known Managed Accounts by ID. Smart Rules are considered Quick Rules when they contain at least one **Managed Account Fields - Quick Group ID** filter. Quick Rules can also be accessed via the SmartRules API endpoint.

Quick Navigation

- POST QuickRules
- GET QuickRules
- GET QuickRules/{id}
- GET QuickRules?title={title}
- GET Organizations/{orgID}/QuickRules?title={title}
- DELETE QuickRules/{id}
- DELETE QuickRules?title={title}
- DELETE Organizations/{orgID}/QuickRules?title={title}

For more information on related topics, please see:

- "GET QuickRules/{quickRuleID}/ManagedAccounts" on page 198
- "PUT QuickRules/{quickRuleID}/ManagedAccounts" on page 200
- "POST QuickRules/{quickRuleID}/ManagedAccounts/{accountID}" on page 203
- "DELETE QuickRules/{quickRuleID}/ManagedAccounts/{accountID}" on page 205

POST QuickRules

Purpose

Creates a new Quick Rule with the Managed Accounts referenced by ID, containing a single filter of type **Managed Account Fields - Quick Group ID** and a single action of type **Show as Smart Group**.

Required Permissions

Smart Rule Management - Managed Account (Read/Write).

Request Body

Content-Type: application/json

```
{
AccountIDs: [ int, …],
Title: string,
Category: string,
Description: string
}
```

Request Body Details

- **AccountIDs:** (required) A list of Managed Account IDs to add to the Quick Rule.
- **Title:** (required) The title/name of the new Quick Rule. Must be unique across all Quick Rules and all Smart Rules.
- **Category:** (optional, default: Quick Rules) The category in which to place the Quick Rule.
- **Description:** (optional, default: <value of Title>) The Quick Rule description.

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool
}
```

Response Codes

201 – Request successful. Quick Rule in the response body.

For more information, please see "Common Response Codes" on page 13.

GET QuickRules

Purpose

Returns a list of Quick Rules to which the current user has at least Read access.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool
},
...
]
```

Response Codes

200 – Request successful. Smart Rules in the response body.

For more information, please see "Common Response Codes" on page 13.

GET QuickRules/{id}

Purpose

Returns a Quick Rule by ID.

Required permissions

Read access to the Quick Rule referenced by ID

URL Parameters

id: ID of the Quick Rule.

Request Body

None

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool
}
```

Response Codes

200 – Request successful. Quick Rule in the response body.

ℹ For more information, please see "Common Response Codes" on page 13.

GET QuickRules?title={title}

Purpose

Returns a Quick Rule by title.

In a multi tenant environment, assumes Global Organization.

Required permissions

Read access to the Quick Rule referenced by title

Query Parameters

title: Title of the Quick Rule.

Request Body

None

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool
}
```

Response Codes

200 – Request successful. Quick Rule in the response body.

ℹ For more information, please see "Common Response Codes" on page 13.

GET Organizations/{orgID}/QuickRules?title={title}

Purpose

Returns a Quick Rule by Organization ID and title.

Only valid in a mult tenant environment.

Required permissions

Read access to the Quick Rule referenced by Organization and title

URL Parameters

orgID: ID of the Organization.

Query Parameters

title: Title of the Quick Rule.

Request Body

None

Response Body

Content-Type: application/json

```
{
SmartRuleID: int,
OrganizationID : string, // can be null
Title: string,
Description: string,
Category: string,
Status: int,
LastProcessedDate: datetime,
IsReadOnly: bool
}
```

Response Codes

200 – Request successful. Quick Rule in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE QuickRules/{id}

Purpose

Deletes a Quick Rule by ID.

Required Permissions

Read/Write access to the Quick Rule referenced by ID

URL Parameters

ID: ID of the Quick Rule

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> For more information, please see "Common Response Codes" on page 13.

DELETE QuickRules?title={title}

Purpose

Deletes a Quick Rule by title.

In a mult tenant environment, assumes Global Organization.

Required Permissions

Read/Write access to the Quick Rule referenced by title

Query Parameters

title: Title of the Quick Rule.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> For more information, please see "Common Response Codes" on page 13.

DELETE Organizations/{orgID}/QuickRules?title={title}

Purpose

Deletes a Quick Rule by Organization ID and title. Only valid in a multi tenant environment.

Required permissions

Read/Write access to the Quick Rule referenced by Organization and title

URL Parameters

orgID: ID of the Organization.

Query Parameters

title: Title of the Quick Rule.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

For more information, please see "Common Response Codes" on page 13.

Replay

Quick Navigation

- POST pbsm/replay
- GET pbsm/replay/{replayId}
- PUT pbsm/replay/{replayId}
- DELETE pbsm/replay/{replayId}

POST pbsm/replay

Purpose

Creates a new replay session for a specified session token. The session token can be discovered using the Sessions endpoints.

Query Parameters

None

Request Body

Content-Type: application/json

```
{
   id: string, // Session Token from query to <base>/Sessions endpoint
   record_key: string, // RecordKey from query to <base>/Sessions endpoint
   protocol: string, // When session Type is 0 this should be RDP or for type 1 SSH
   headless: boolean // Must be set to true
}
```

Response Body

Content-Type: application/json

```
{
   id: string, // ReplayID for this replay session
   token: string, // ReplayID for this replay session
   ticket: string, // Ticket value used internally
}
```

Response Codes

- 200 – Request successful.
- 403 – Access forbidden. Response body contains a message or status code indicating the reason for this forbidden access:
- 404 – Not found. The requested replay session was not found on the server

For more information, please see "Common Response Codes" on page 13.

GET pbsm/replay/{replayId}

Purpose

Displays the replay session details.

URL Parameters

ReplayID: ID of the replay session returned from POST pbsm/replay.

Query Parameters

- **jpeg=(scale):** Requests a jpeg image of the current RDP replay session scaled in size by the given scale.
- **png=(scale):** Requests a png image of the current RDP replay session scaled in size by the given scale.
- **screen=1:** Requests a text representation of the current SSH session.

Request Body

None

Response Body

Content-Type: application/json

```
{
tstamp: int, // Start time of the session in seconds
end: int, // End time of the session in seconds
offset: int, // Current offset of replay session in ms
next: int, // Offset of next activity of replay session in ms
speed: int, // Speed of replay session as a %
eof: boolean, // Set to true when the end of the replay has been reached
duration: int // Duration in ms of the replay session
}
```

Response Codes

- 200 – Request successful.
- 403 – Access forbidden. Response body contains a message or status code indicating the reason for this forbidden access
- 404 – Not found. The requested replay session was not found on the server

For more information, please see "Common Response Codes" on page 13.

PUT pbsm/replay/{replayId}

Purpose

Controls the replay session status.

URL Parameters

ReplayID: ID of the replay session returned from POST pbsm/replay

Query Parameters

None

Request Body

```
{
speed: int, // Sets the replay speed of this session as a %
offset: int, // Sets the offset of the replay cursor for this session in ms
next: int // Requests the next changed frame based on the given % change
}
```

Response Body

Content-Type: application/json

```
{
tstamp: int, // Start time of the session in seconds
end: int, // End time of the session in seconds
offset: int, // Current offset of replay session in ms
next: int, // Offset of next activity of replay session in ms
speed: int, // Speed of replay session as a %
eof: boolean, // Set to true when the end of the replay has been reached
duration: int // Duration in ms of the replay session
}
```

Response Codes

- 200 – Request successful.
- 403 – Access forbidden. Response body contains a message or status code indicating the reason for this forbidden access:
- 404 – Not found. The requested replay session was not found on the server

For more information, please see "Common Response Codes" on page 13.

DELETE pbsm/replay/{replayId}

Purpose

Terminates the replay session.

URL Parameters

ReplayID: ID of the replay session returned from POST pbsm/replay

Query Parameters

None

Request Body

None

Response Codes

- 200 – Request successful.
- 403 – Access forbidden. Response body contains a message or status code indicating the reason for this forbidden access:
- 404 – Not found. The requested replay session was not found on the server

For more information, please see "Common Response Codes" on page 13.

Requests

Quick Navigation

- GET Requests
- POST Requests
- POST Aliases/{aliasId}/Requests
- PUT Requests/{id}/Checkin
- PUT Requests/{id}/Approve
- PUT Requests/{id}/Deny
- PUT Requests/{id}/RotateOnCheckin

For more information on related topics, please see "Credentials" on page 132

GET Requests

Purpose

Lists requests for the current user.

Query Parameters

- **status:** (optional, default: all) Status of requests to return.
  - **all:** Both active and pending requests.
  - **active:** Requests that have been approved (including auto-approved).
  - **pending:** Requests that have not yet been approved.
- **queue:** (optional, default: req): Type of request queue to return.
  - **req:** Requestor queue, returns requests available to the user as a Requestor.
  - **app:** Approver queue, returns requests for an Approver or Requestor/Approver that have either been approved by the user (active) or have not yet been approved (pending).

Request Body

None

Response Body

Content-Type: application/json

```
[
{
RequestID: int,
SystemID: int,
SystemName: string,
AccountID: int,
AccountName: string,
DomainName: string,
AliasID: int,
ApplicationID: int,
RequestReleaseDate: date-formatted string,
ApprovedDate: date-formatted string,
ExpiresDate: date-formatted string,
Status: string,
AccessType: string
},
...
]
```

Response Codes

- 200 – Request successful. Requests in the response body.
- 403 – Access forbidden. Response body contains a message or status code indicating the reason for this forbidden access:
    - 4032 – Requestor Only API or account. Only Requestors can access this API or account.
    - 4033 – Approver Only API or account. Only Approvers can access this API or account.

For more information, please see "Common Response Codes" on page 13.

POST Requests

Purpose

Creates a new release request.

Required Roles

- Requestor or Requestor/Approver Role to Managed Account referenced by ID

For information on ISA role access, please see "ISA Requests" on page 164

Request Body

Content-Type: application/json

```
{
AccessType: string,
SystemID: int,
AccountID: int,
ApplicationID: int, // can be null
DurationMinutes : int,
Reason : string,
AccessPolicyScheduleID : int, // can be null
ConflictOption : string,
TicketSystemID : int,
TicketNumber : string,
RotateOnCheckin: bool
}
```

Request Body Details

- **AccessType:** (optional, default: View) The type of access requested (View, RDP, SSH, App).
  - **View: View Password** access.
  - **RDP:** RDP access (corresponds to POST Sesssions SessionType **RDP** or **rdpfile**).
  - **SSH:** SSH access (corresponds to POST Sesssions SessionType **SSH**).
  - **App:** Application access (corresponds to POST Sesssions SessionType **App** or **appfile**).
- **SystemID:** (required) ID of the Managed System to request.
- **AccountID:** (required) ID of the Managed Account to request.
- **ApplicationID:** (required when **AccessType**=**App**): ID of the Application for an Application-based request.
- **DurationMinutes:** (required: 1-525600) The request duration (in minutes).
- **Reason:** (optional) The reason for the request.
- **AccessPolicyScheduleID:** (optional) The Schedule ID of an Access Policy to use for the request. If omitted, automatically selects the best schedule.
- **ConflictOption:** (optional) The conflict resolution option to use if an existing request is found for the same user, system, and account (reuse, renew). If omitted and a conflicting request is found, returns a 409 code (see below).
  - **reuse:** Returns an existing, approved request ID for the same user/system/account/access type (if one exists). If the request does not already exist, creates a new request using the request body details.
  - **renew:** Cancels any existing approved requests for the same user/system/account and creates a new request using the request body details.
- **TicketSystemID:** ID of the ticket system. If omitted, then default ticket system will be used.
- **TicketNumber:** Number of associated ticket. Can be required if ticket system is marked as required in the global options.
- **RotateOnCheckin:** (optional, default: true) True to rotate the credentials on check-in/expiry, otherwise false. This property can only be used if the Access Policy (either auto-selected or given in **AccessPolicyScheduleID**) supports it.

For more information, please see the *Allow API Rotation Override* Access Policy setting under *View* access.

***Note:** In reference to **RotateOnCheckin**, If the Managed Account given in **AccountID** does not rotate the credentials after check-in/expiry, this setting is ignored.*

Response Body

```
{
RequestID: int
}
```

Response Codes

- 200 – Existing request is being reused. Existing request ID in the response body.
- 201 – Request successful. Request ID in the response body.
- 403 – User does not have permissions to request the indicated account or the account does not have API access enabled. Response body contains a status code indicating the reason for this forbidden access:
  - 4031 – User does not have permission to request the account or the account is not valid for the system.
  - 4032 – Requestor Only API or account. Only Requestors can access this API or account.
  - 4033 – Approver Only API or account. Only Approvers can access this API or account.
  - 4035 - Not enough approvers configured to approve a request.
- 409 – Conflicting request exists. This user or another user has already requested a password for the specified account within the next <durationMinutes> window.

For more information, please see "Common Response Codes" on page 13.

POST Aliases/{aliasId}/Requests

Purpose

Creates a new release request using an Alias.

Required Roles

Requestor or Requestor/Approver Role to Managed Account referenced by the Alias

URL Parameters

aliasId: ID of the Managed Account Alias.

Request Body

Content-Type: application/json

```
{
AccessType: string,
DurationMinutes : int,
Reason : string,
AccessPolicyScheduleID : int, // can be null
ConflictOption : string,
TicketSystemID : int,
TicketNumber : string,
RotateOnCheckin: bool
}
```

Request Body Details

- **AccessType:** (optional, default: View) The type of access requested (View, RDP, SSH, App).
  - **View:** View Password access.
  - **RDP:** RDP access (corresponds to POST Sesssions SessionType **RDP** or **rdpfile**).
  - **SSH:** SSH access (corresponds to POST Sesssions SessionType **SSH**).
- **DurationMinutes:** (required: 1-525600): The request duration (in minutes).
- **Reason:** (optional) The reason for the request.
- **AccessPolicyScheduleID:** (optional) The Schedule ID of an Access Policy to use for the request. If omitted, automatically selects the best schedule.
- **ConflictOption:** (optional) The conflict resolution option to use if an existing request is found for the same user, system, and account (reuse, renew). If omitted and a conflicting request is found, returns a 409 (see below).
  - **reuse:** Return an existing, approved request ID for the same user/system/account/access type (if one exists). If the request does not already exist, creates a new request using the request body details.
  - **renew:** Cancel any existing approved requests for the same user/system/account and create a new request using the request body details.
- **TicketSystemID:** ID of the ticket system. If omitted then default ticket system will be used.
- **TicketNumber:** Number of associated ticket. Can be required if ticket system is marked as required in the global options.
- **RotateOnCheckin:** (optional, default: true) True to rotate the credentials on check-in/expiry, otherwise false. This property can only be used if the Access Policy (either auto-selected or given in **AccessPolicyScheduleID**) supports it. If the Managed Account given in AccountID does not rotate the credentials after check-in/expiry, this setting is ignored.

> i For more information, please see the *Allow API Rotation Override* Access Policy setting under *View* access.

Response Body

```
{
RequestID: int
}
```

Response Codes

- 200 – Existing request is being reused. Existing request ID in the response body.
- 201 – Request successful. Request ID in the response body.
- 403 – User does not have permissions to request the indicated alias or the account referenced by the alias does not have API access enabled. Response body contains a status code indicating the reason for this forbidden access:
  - 4031 – User does not have permission to request the account or the account is not valid for the system.
  - 4032 – Requestor Only API or account. Only Requestors can access this API or account.
  - 4033 – Approver Only API or account. Only Approvers can access this API or account.
  - 4035 - Not enough approvers configured to approve a request.
- 409 – Conflicting request exists. This user or another user has already requested a password for the specified account within the next <durationMinutes> window.

For more information, please see "Common Response Codes" on page 13.

PUT Requests/{id}/Checkin

Alternate URI (deprecated)

PUT Requests/Release/{id}

Purpose

Checks-in/releases a request before it has expired.

Required Roles

Requestor Role to Managed Account referenced by the request

URL Parameters

id: ID of the Request to check-in/release.

Request Body

Content-Type: application/json

```
{
Reason : string
}
```

Request Body Details

Reason: (optional) A reason or comment why the request is being released.

Response Body

None

Response Codes

- 204 – Request successful. No content in body.
- 403 – User does not have permissions to release the indicated request or the associated account does not have API access enabled. Message or status code in response body:
    - 4031 – User does not have permission to release a password.
    - 4034 – Request is not yet approved.

For more information, please see "Common Response Codes" on page 13.

PUT Requests/{id}/Approve

Purpose

Approves a pending request.

Required Roles

Approver or Requestor/Approver Role to Managed Account referenced by the request

URL Parameters

id: ID of the Request to approve.

Request Body

Content-Type: application/json

```
{
Reason : string
}
```

Request Body Details

Reason: (optional) A reason or comment why the request is being approved.

Response Body

None

Response Codes

- 204 – Request successful. No content in body.
- 403 – User does not have permissions to approve the indicated request or the associated account does not have API access enabled. Message or status code in response body:
  - 4033 – Approver only - User cannot approve his or her own request.
  - 4036 – Request has been approved already.

For more information, please see "Common Response Codes" on page 13.

PUT Requests/{id}/Deny

Purpose

Denies/cancels an active or pending request.

Required Roles

Approver or Requestor/Approver Role to Managed Account referenced by the request

URL Parameters

id: ID of the Request to deny/cancel.

Request Body

Content-Type: application/json

```
{
Reason : string
}
```

Request Body Details

Reason: (optional) A reason or comment why the request is being denied/cancelled

Response Body

None

Response Codes

- 204 – Request successful. No content in body.
- 403 – User does not have permissions to deny the indicated request or the associated account does not have API access enabled. Message or status code in response body:
- 4033 – Approver only - User cannot deny his or her own request.

For more information, please see "Common Response Codes" on page 13.

PUT Requests/{id}/RotateOnCheckin

Purpose

Updates a request to rotate the credentials on check-in/expiry.

Note: *If POST Requests RotateOnCheckin=false, this will update the request to true. If POST Requests RotateOnCheckin=true, the request will not be modified.*

Requirements

- Current User must be the owner of the request.
- Request must not be cancelled or expired.

URL Parameters

id: ID of the Request to update.

Request Body

None

Response Body

None

Response Codes

204 – Request successful. No content in body.

ℹ For more information, please see "Common Response Codes" on page 13.

Request Termination

Quick Navigation

- POST ManagedAccounts/{managedAccountID}/Requests/Terminate
- POST ManagedSystems/{managedSystemID}/Requests/Terminate
- POST Users/{userID}/Requests/Terminate

POST ManagedAccounts/{managedAccountID}/Requests/Terminate

Purpose

Terminates all active Requests by Managed Account ID.

Required Permissions

Password Safe API Global Quarantine (Read/Write)

URL Parameters

managedAccountID: ID of the Managed Account

Request Body

Content-Type: application/json

```
{
Reason : string
}
```

Request Body Details

Reason: (optional) A reason or comment why the requests are being terminated.

Response Body

None

Response Codes

204 – Termination successful. No content in body.

For more information, please see "Common Response Codes" on page 13.

POST ManagedSystems/{managedSystemID}/Requests/Terminate

Purpose

Terminates all active Requests by Managed System ID.

Required Permissions

Password Safe API Global Quarantine (Read/Write)

URL Parameters

managedSystemID: ID of the Managed System

Request Body

Content-Type: application/json

```
{
Reason : string
}
```

Request Body Details

Reason: (optional) A reason or comment why the requests are being terminated.

Response Body

None

Response Codes

204 – Termination successful. No content in body.

For more information, please see "Common Response Codes" on page 13.

POST Users/{userID}/Requests/Terminate

Purpose

Terminates all active Requests by Requestor User ID.

Required Permissions

Password Safe API Global Quarantine (Read/Write)

URL Parameters

userID: ID of the Requestor User

Request Body

Content-Type: application/json

```
{
Reason : string
}
```

Request Body Details

Reason: (optional) A reason or comment why the requests are being terminated.

Response Body

None

Response Codes

204 – Termination successful. No content in body.

For more information, please see "Common Response Codes" on page 13.

Request Sets

Request Sets are a grouping of requests to the same system and account with different access types (i.e. View and RDP). Requests in a request set are also accessible individually via GET Requests.

Quick Navigation

- GET RequestSets
- POST RequestSets

GET RequestSets

Purpose

Lists request sets for the current user.

Query Parameters

status: (optional, default: all) Status of request sets to return (all, active, pending).

Request Body

None

Response Body

Content-Type: application/json

```
[
{
RequestSetID: string,
Requests:
    [
    {
    RequestID: int,
    RequestorName: string,
    SystemID: int,
    SystemName: string,
    AccountID: int,
    AccountName: string,
    DomainName: string,
    AliasID: int, // can be null
    RequestReleaseDate: date-formatted string,
    ApprovedDate: date-formatted string,
    CanceledDate: date-formatted string,
    ExpiresDate: date-formatted string,
    Status: string,
    AccessType: string,
    ApplicationID: int,
    Reason: string
    },
    ...
    ]
},
...
]
```

Response Codes

- 200 – Request successful. Requests in the response body.
- 403 – Access forbidden. Response body contains a message or status code indicating the reason for this forbidden access:
    - 4032 – Requestor Only API or account. Only Requestors can access this API or account.
    - 4033 – Approver Only API or account. Only Approvers can access this API or account.

For more information, please see "Common Response Codes" on page 13.

POST RequestSets

Purpose

Creates a new release request set.

Required Roles

- Requestor or Requestor/Approver Role to Managed Account referenced by ID
- Information Systems Administrator (ISA) role access

For more information, please see:

- "ISA Requests" on page 164
- "ISA Sessions" on page 166

Request Body

Content-Type: application/json

```
{
AccessTypes: [ string, … ],
SystemID: int,
AccountID: int,
DurationMinutes : int,
Reason : string,
TicketSystemID : int,
TicketNumber : string
}
```

Request Body Details

- **AccessTypes:** (at least two are required) A list of the types of access requested (View, RDP, SSH, App).
- **SystemID:** (required) ID of the Managed System to request.
- **AccountID:** (required) ID of the Managed Account to request.
- **DurationMinutes:** (required) The request duration (in minutes).
- **Reason:** (optional) The reason for the request.
- **TicketSystemID:** ID of the ticket system. If omitted then default ticket system will be used.
- **TicketNumber:** Number of associated ticket. Can be required if ticket system is marked as required in the global options.

Response Body

Content-Type: application/json

```
{
RequestSetID: string,
Requests:
    [
    {
    RequestID: int,
    SystemID: int,
    SystemName: string,
    AccountID: int,
    AccountName: string,
    DomainName: string,
    AliasID: int,
    RequestReleaseDate: date-formatted string,
    ApprovedDate: date-formatted string,
    ExpiresDate: date-formatted string,
    Status: string,
    AccessType: string
    },
    ...
    ]
}
```

Response Codes

- 201 – Request successful. Request Set in the response body.
- 403 – User does not have permissions to perform a request for the indicated account or the account does not have API access enabled. Response body contains a status code indicating the reason for this forbidden access:
  - 4031 – User does not have permission to request the account or the account is not valid for the system.
  - 4032 – Requestor Only API or account. Only Requestors can access this API or account.
  - 4033 – Approver Only API or account. Only Approvers can access this API or account.
  - 4035 - Not enough approvers configured to approve a request.
- 409 – Conflicting request exists. Another user has already requested a password for the specified account within the next <durationMinutes> window.

For more information, please see "Common Response Codes" on page 13.

Roles

(i.e. Requestor, Approver, Credentials Manager, etc.)

GET Roles

Purpose

Returns a list of Password Safe Roles.

Required Permissions

Password Safe Role Management (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
RoleID : int,
Name : string
},
…
]
```

Response Codes

200 – Request successful. Roles in the response body.

For more information, please see "Common Response Codes" on page 13.

User Group Roles

Quick Navigation

- GET UserGroups/{userGroupId}/SmartRules/{smartRuleId}/Roles
- POST UserGroups/{userGroupId}/SmartRules/{smartRuleId}/Roles
- DELETE UserGroups/{userGroupId}/SmartRules/{smartRuleId}/Roles

GET UserGroups/{userGroupId}/SmartRules/{smartRuleId}/Roles

Purpose

Returns a list of Roles for the User Group and Smart Rule referenced by ID.

Required Permissions

- User Accounts Management (Read)
- Password Safe Role Management (Read)

URL Parameters

- **userGroupId:** ID of the User Group.
- **smartRuleId:** ID of the Smart Rule.

Request Body

None

Response Body

Content-Type: application/json

```
[
{
RoleID : int,
Name : string
},
…
]
```

Response Codes

200 – Request successful. Roles in the response body.

For more information, please see "Common Response Codes" on page 13.

POST UserGroups/{userGroupId}/SmartRules/{smartRuleId}/Roles

Purpose

Sets Password Safe Roles for the User Group and Smart Rule referenced by ID.

Required Permissions

- User Accounts Management (Read/Write)
- Password Safe Role Management (Read/Write)

URL Parameters

- **userGroupId:** ID of the User Group.
- **smartRuleId:** ID of the Smart Rule.

Request Body

Content-Type: application/json

```
{
Roles :
    [
    {
    RoleID : int
    },
    …
    ],
AccessPolicyID : int
}
```

Request Body Details

- **Roles:** (required) Zero or more roles to set on the UserGroup-SmartRule.
- **AccessPolicyID:** The Access Policy ID to set on the UserGroup-SmartRule. Required when the Requestor or Requestor/Approver role is set.

Response Body

None

Response Codes

204 – Request successful. No content in body.

For more information, please see "Common Response Codes" on page 13.

DELETE UserGroups/{userGroupId}/SmartRules/{smartRuleId}/Roles

Purpose

Deletes all Password Safe Roles for the User Group and Smart Rule referenced by ID.

Required Permissions

- User Accounts Management (Read/Write)
- Password SafeRole Management (Read/Write)

URL Parameters

- **userGroupId:** ID of the User Group.
- **smartRuleId:** ID of the Smart Rule.

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

ℹ For more information, please see "Common Response Codes" on page 13.

Sessions

Quick Navigation

- GET Sessions
- GET Sessions/{id}
- POST Requests/{requestID}/Sessions

GET Sessions

Purpose

Returns a list of Sessions.

Required Permissions

A member of the Administrators group, or ISA or Auditor Role to at least one Asset Smart Rule.

Query Parameters (optional)

- **status:** Session status - A single value or comma-delimited list of values:
  - **0:** Not Started
  - **1:** In Progress
  - **2:** Completed
  - **5:** Locked
  - **7:** Terminated (deprecated)
  - **8:** Logged Off
  - **9:** Disconnected (RDP only)
- **userID:** ID of the User that requested the session

Request Body

None

Response Body

Content-Type: application/json

```
[
   {
      SessionID : int,
      UserID : int,
      NodeID : string,
      Status : int,
      ArchiveStatus : int,
      Protocol : int,
      StartTime : date,
      EndTime : date,
      Duration : int,
      AssetName : string,
      ManagedSystemID : int, // can be null
      ManagedAccountID : int,
      ManagedAccountName : string,
      RecordKey : string,
      Token : string
   },
…
]
```

Response Codes

200 – Request successful. Sessions in the response body.

> For more information, please see "Common Response Codes" on page 13.

GET Sessions/{id}

Purpose

Returns a Session by ID.

Required Permissions

A member of the Administrators group, or ISA or Auditor Role to at least one Asset Smart Rule.

URL Parameters

id: ID of the session.

Request Body

None

Response Body

Content-Type: application/json

```
{
SessionID : int,
UserID : int,
NodeID : string,
Status : int,
ArchiveStatus : int,
Protocol : int,
StartTime : date,
EndTime : date,
Duration : int,
AssetName : string,
ManagedSystemID : int,
ManagedAccountID : int,
ManagedAccountName : string,
RecordKey : string,
Token : string
}
```

Response Codes

200 – Request successful. Sessions in the response body.

For more information, please see "Common Response Codes" on page 13.

POST Requests/{requestID}/Sessions

Purpose

Create a new session for the given release.

Requirements

Must be the owner of the requestID.

URL Parameters

requestID: ID of the Request for which to create a session.

Request Body

Content- Type: application/json

```
{
SessionType : string,
NodeID : string
}
```

Request Body Details

- **SessionType:** (required) The type of session to create (SSH or sshticket, RDP or rdpticket, rdpfile, app, or appfile).
- **NodeID:** (optional) ID of the node that should be used to establish the Session. If **NodeID** is not given or if the Remote Session Proxy feature is disabled, uses the local node.

Response Body (SSH or sshticket)

Content-Type: application/json

```
{
ID : string,
Ticket : string,
Host : string,
Port : string,
TicketAtHost : string,
Link : string,
Command : string,
SessionID : int,
NodeID : string
}
```

Response Body (RDP or rdpticket)

Content-Type: application/json

```
{
ID : string,
Ticket : string,
Host : string,
Port : string,
SessionID : int,
NodeID : string
}
```

Response Body (rdpfile)

RDP File as an attachment.

Response Body (app – when the target system is Unix- or ssh-based)

Content-Type: application/json

```
{
ID : string,
Ticket : string,
Host : string,
Port : string,
TicketAtHost : string,
Link : string,
Command : string,
SessionID : int,
NodeID : string
}
```

Response Body (app – when the target system is Windows- or rdp-based)

Content-Type: application/json

```
{
ID : string,
Ticket : string,
Host : string,
Port : string,
SessionID : int,
NodeID : string
}
```

Response Body (appfile)

RDP File as an attachment.

Response Codes

- 201– Request successful. Session details or RDP file in the response body.
- 403 – Access forbidden. Response body contains a message or status code indicating the reason for this forbidden access:
- 4034 – Request is not yet approved.

> For more information, please see "Common Response Codes" on page 13.

POST Sessions/Admin

Purpose

Create a new admin session.

Required Roles

Password Safe Admin Session (Read/Write)

Request Body

Content-Type: application/json

```
{
SessionType : string,
HostName : string,
Port : int, // can be null
DomainName : string,
UserName : string,
Password : string,
Reason : string,
Resolution : string,
RDPAdminSwitch : bool,
SmartSizing : bool,
NodeID : string
}
```

Request Body Details

- **SessionType:** (required) The type of session to create (**SSH** or **sshticket**, **RDP** or **rdpticket**, or **rdpfile**)
- **HostName:** (required) Name or IP of the target host.
- **Port:** (optional, default: <configured default port>) Port to use for the connection.
- **DomainName:** (optional) The domain name of the user.
- **UserName:** (required) The username.
- **Password:** (required) The password.
- **Reason:** (optional) The reason for the session.
- **Resolution (RDP-only):** (optional, default: <configured default resolution>) The default resolution (i.e **1024x768** or **Maximized**).
- **RDPAdminSwitch (RDP-only):** (optional, default: false) True to use the RDP **/admin** switch, otherwise false.
- **SmartSizing (RDP-only):** (optional, default: false) True to use RDP **Smart Sizing**, otherwise false. Applies only when **SessionType=rdpfile**.
- **NodeID:** (optional) ID of the node that should be used to establish the Session. If **NodeID** is not given or if the Remote Session Proxy feature is disabled, uses the local node.

Response Body (SSH or sshticket)

Content-Type: application/json

```
{
ID : string,
Ticket : string,
Host : string,
Port : string,
TicketAtHost : string,
Link : string,
Command : string,
SessionID : int,
NodeID : string
}
```

Response Body (RDP or rdpticket)

Content-Type: application/json

```
{
ID : string,
Ticket : string,
Host : string,
Port : string,
SessionID : int,
NodeID : string
}
```

Response Body (rdpfile)

RDP File as an attachment.

Response Codes

201 – Request successful. Session details or RDP file in the response body.

403 – Access forbidden. Response body contains a message or status code indicating the reason for this forbidden access.

For more information, please see "Common Response Codes" on page 13.

Session Locking

Quick Navigation

- POST Sessions/{sessionID}/Lock
- POST ManagedAccounts/{managedAccountID}/Sessions/Lock
- POST ManagedSystems/{managedSystemID}/Sessions/Lock

POST Sessions/{sessionID}/Lock

Purpose

Locks an active Session.

Required Permissions

One of:

- Password Safe API Global Quarantine (Read/Write), or
- Password Safe Active Session Reviewer Role, ISA Role, or a member of BeyondInsight Administrators group.

URL Parameters

sessionID: ID of the session.

Request Body

None

Response Body

None

Response Codes

204 – Lock successful. No content in body.

i For more information, please see "Common Response Codes" on page 13.

POST ManagedAccounts/{managedAccountID}/Sessions/Lock

Purpose

Locks all active Sessions by Managed Account ID.

Required Permissions

Password Safe API Global Quarantine (Read/Write)

URL Parameters

managedAccountID: ID of the Managed Account

Request Body

None

Response Body

None

Response Codes

204 – Lock successful. No content in body.

> For more information, please see "Common Response Codes" on page 13.

POST ManagedSystems/{managedSystemID}/Sessions/Lock

Purpose

Locks all active Sessions by Managed System ID.

Required Permissions

Password Safe API Global Quarantine (Read/Write)

URL Parameters

managedSystemID: ID of the Managed System

Request Body

None

Response Body

None

Response Codes

204 – Lock successful. No content in body.

> For more information, please see "Common Response Codes" on page 13.

Session Termination

Quick Navigation

- POST Sessions/{sessionID}/Terminate
- POST ManagedAccounts/{managedAccountID}/Sessions/Terminate
- POST ManagedSystems/{managedSystemID}/Sessions/Terminate

POST Sessions/{sessionID}/Terminate

Purpose

Terminates an active Session.

Required Permissions

One of:

- Password Safe API Global Quarantine (Read/Write), or
- Password Safe Active Session Reviewer Role, ISA Role, or a member of BeyondInsight Administrators group.

URL Parameters

sessionID: ID of the Session to terminate.

Request Body

None

Response Body

None

Response Codes

204 – Termination successful. No content in body.

For more information, please see "Common Response Codes" on page 13.

POST ManagedAccounts/{managedAccountID}/Sessions/Terminate

Purpose

Terminates all active Sessions by Managed Account ID.

Required Permissions

Password Safe API Global Quarantine (Read/Write)

URL Parameters

managedAccountID: ID of the Managed Account.

Request Body

None

Response Body

None

Response Codes

204 – Termination successful. No content in body.

ℹ For more information, please see "Common Response Codes" on page 13.

POST ManagedSystems/{managedSystemID}/Sessions/Terminate

Purpose

Terminates all active Sessions by Managed System ID.

Required Permissions

Password Safe API Global Quarantine (Read/Write)

URL Parameters

managedSystemID: ID of the Managed System.

Request Body

None

Response Body

None

Response Codes

- 204 – Termination successful. No content in body.

ℹ For more information, please see "Common Response Codes" on page 13.

Synced Accounts

Synced Accounts are Managed Accounts subscribed/synced to another Managed Account.

Quick Navigation

- GET ManagedAccounts/{id}/SyncedAccounts
- POST ManagedAccounts/{id}/SyncedAccounts/{syncedAccountID}
- DELETE ManagedAccounts/{id}/SyncedAccounts
- DELETE ManagedAccounts/{id}/SyncedAccounts/{syncedAccountID}

GET ManagedAccounts/{id}/SyncedAccounts

Purpose

Returns a list of subscribed/synced Managed Accounts by Managed Account ID.

Required Permissions

Password Safe Account Management (Read)

URL Parameters

id: ID of the parent Managed Account.

Request Body

None

Response Body

Content-Type: application/json

```
 [
 {
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
},
...
]
```

Response Body Details

- **DomainName:** The domain name for a domain-type account.
- **AccountName:** The name of the account.
- **DistinguishedName:** The distinguished name of an LDAP Managed Account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:** True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.
- **MaxConcurrentRequests:** (0-999, 0 is unlimited) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.

- ○ **ChangeFrequencyType:** The change frequency for scheduled password changes:
    - ■ **first:** Changes scheduled for the first day of the month
    - ■ **last:** Changes scheduled for the last day of the month
    - ■ **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
- ○ **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
- ○ **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.

- **ParentAccountID:** If this is a subscribed account (**IsSubscribedAccount**), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

> For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

200 – Request successful. Linked Managed Accounts in the response body.

> For more information, please see "Common Response Codes" on page 13.

POST ManagedAccounts/{id}/SyncedAccounts/{syncedAccountID}

Purpose

Subscribes/syncs a Managed Account to the Managed Account referenced by ID.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

- **id:** ID of the parent Managed Account
- **syncedAccountID:** ID of the synced Managed Account

Request Body

None

Response Body

Content-type: application/json

```
{
ManagedAccountID : int,
ManagedSystemID : int,
DomainName : string,
AccountName : string,
DistinguishedName : string,
PasswordFallbackFlag : bool,
LoginAccountFlag : bool,
Description : string,
PasswordRuleID : int,
ApiEnabled : bool,
ReleaseNotificationEmail : string,
ChangeServicesFlag : bool,
RestartServicesFlag : bool,
ReleaseDuration : int,
MaxReleaseDuration : int,
ISAReleaseDuration : int,
MaxConcurrentRequests : int,

AutoManagementFlag : bool,
    DSSAutoManagementFlag : bool,
    CheckPasswordFlag : bool,
    ResetPasswordOnMismatchFlag : bool,
    ChangePasswordAfterAnyReleaseFlag : bool,
    ChangeFrequencyType : string,
    ChangeFrequencyDays : int,
    ChangeTime : string,

ParentAccountID : int, // can be null
IsSubscribedAccount : bool,
LastChangeDate: datetime, // can be null
NextChangeDate: datetime, // can be null
IsChanging: bool
}
```

Response Body Details

- **AccountName:** The name of the account.
- **PasswordFallbackFlag:** True if failed DSS authentication can fall back to password authentication, otherwise false.
- **LoginAccountFlag:** True if the account should use the Managed System Login Account for SSH sessions, otherwise false.
- **Description:** A description of the account.
- **PasswordRuleID:** ID of the Password Rule assigned to this Managed Account.
- **ApiEnabled:**True if the account can be requested through the API, otherwise false.
- **ReleaseNotificationEmail:** Email address used for notification emails related to this Managed Account.
- **ChangeServicesFlag:** True if services run as this user should be updated with the new password after a password change, otherwise false.
- **RestartServicesFlag:** True if services should be restarted after the run as password is changed (**ChangeServicesFlag**), otherwise false.
- **ReleaseDuration:** (minutes: 1-525600) Default release duration.
- **MaxReleaseDuration:** (minutes: 1-525600) Default maximum release duration.
- **ISAReleaseDuration:** (minutes: 1-525600) Default Information Systems Administrator (ISA) release duration.

- **MaxConcurrentRequests:** (0-999, 0 means unlimited) Maximum number of concurrent password requests for this account.
- **AutoManagementFlag:** True if password auto-management is enabled, otherwise false.
    - **DSSAutoManagementFlag:** True if DSS Key auto-management is enabled, otherwise false.
    - **CheckPasswordFlag:** True to enable password testing, otherwise false.
    - **ChangePasswordAfterAnyReleaseFlag:** True to change passwords on release of a request, otherwise false.
    - **ResetPasswordOnMismatchFlag:** True to queue a password change when scheduled password test fails, otherwise false.
    - **ChangeFrequencyType:** The change frequency for scheduled password changes:
        - **first:** Changes scheduled for the first day of the month
        - **last:** Changes scheduled for the last day of the month
        - **xdays:** Changes scheduled every x days (**ChangeFrequencyDays**)
    - **ChangeFrequencyDays:** (days: 1-999) When **ChangeFrequencyType** is **xdays**, password changes take place this configured number of days.
    - **ChangeTime:** (24hr format: 00:00-23:59) UTC time of day scheduled password changes take place.
- **ParentAccountID:** If this is a subscribed account (**IsSubscribedAccount**), this is the ID of the Parent Managed Account.
- **IsSubscribedAccount:** True if the account is a Synced or Subscribed Account, otherwise false.

For more information, please see *Configure Subscriber Accounts* in the Password Safe Admin Guide at https://www.beyondtrust.com/docs/beyondinsight-password-safe/ps/index.htm.

- **LastChangeDate:** The date and time of the last password change.
- **NextChangeDate:** The date and time of the next scheduled password change.
- **IsChanging:** True if the account credentials are in the process of changing, otherwise false.

Response Codes

- 200 – Account was already synced. Managed Account in the response body.
- 201 – Account was synced successfully. Managed Account in the response body.

For more information, please see "Common Response Codes" on page 13.

DELETE ManagedAccounts/{id}/SyncedAccounts

Purpose

Unsubscribes/unsyncs all Managed Accounts from the parent Managed Account by ID.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

id: ID of the parent Managed Account

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> ℹ For more information, please see "Common Response Codes" on page 13.

DELETE ManagedAccounts/{id}/SyncedAccounts/{syncedAccountID}

Purpose

Unsubscribes/unsyncs a Managed Account from the Managed Account by ID.

Required Permissions

Password Safe Account Management (Read/Write)

URL Parameters

- **id:** ID of the parent Managed Account
- **syncedAccountID:** ID of the synced Managed Account

Request Body

None

Response Body

None

Response Codes

200 – Request successful.

> ℹ For more information, please see "Common Response Codes" on page 13.

Ticket Systems

GET TicketSystems

Purpose

List registered Ticket Systems.

Required Permissions

Ticket System (Read)

Request Body

None

Response Body

Content-Type: application/json

```
[
{
TicketSystemID : int,
IsActive : bool,
TicketSystemName : string,
Description : string,
IsDefaultSystem : bool
},
…
]
```

Response Codes

200 – Request successful. Ticket systems in the response body

For more information, please see "Common Response Codes" on page 13.

Deprecated

The content in this section of the guide has been deprecated and is compatible with earlier versions only.

Quick Navigation

Aliases

[deprecated] GET Aliases/{name}

***Note:** This API has been deprecated and is available for backwards compatibility only. Use GET Aliases?name={name} instead.*

Purpose

Returns a requestable Managed Account Alias by name.

Required Roles

Requestor or Requestor/Approver role for the preferred Managed Account referenced by the Alias.

URL Parameters

name: Name of the Managed Account Alias

Request Body

None

Response Body

Content-Type: application/json

```
{
AliasId: int,
AliasName: string,
SystemId: int,
SystemName: string,
AccountId: int,
AccountName: string,
DomainName: string,
InstanceName: string,
DefaultReleaseDuration: int,
MaximumReleaseDuration: int,
LastChangeDate: datetime,
NextChangeDate: datetime,
IsChanging: bool,
ChangeState: int,

MappedAccounts :
    [
    {
    AliasID: int,
    ManagedSystemID: int,
    ManagedAccountID: int,
    Status: string
    },
    …
    ]
}
```

Response Codes

200 - Request successful. Alias in response body.

For more information, please see "Common Response Codes" on page 13.

Keystrokes

[deprecated] GET Keystrokes/search/{condition}

***Note:** This API has been deprecated and is available for backwards compatibility only. Use POST Keystrokes/Search instead.*

Purpose

Search for Keystrokes by condition/keyword.

Required Roles

Password Safe Auditor Role, ISA Role, or a member of BeyondInsight Administrators group.

URL Parameters

condition: Keyword to search for.

Response Body

Content-Type: application/json

```
[
{
KeystrokeID: long,
SessionID: int,
TimeMarker: long,
Type: byte,
Data: string
},
...
]
```

Response Codes

200 - Request successful. Keystrokes are in response body.

For more information, please see "Common Response Codes" on page 13.

[deprecated] GET Keystrokes/search/{condition}/{type:int}

***Note:** This API has been deprecated and is available for backwards compatibility only. Use POST Keystrokes/Search instead.*

Purpose

Search for Keystrokes by condition/keyword and type.

Required Roles

Password Safe Auditor Role, ISA Role, or a member of BeyondInsight Administrators group.

URL Parameters

- **condition:** Keyword to search for.
- **type:** Type of keystrokes: 0 - All, 1 - StdIn, 2 - StdOut, 4 - Window Event, 5 - User Event.

Response Body

Content-Type: application/json

```
[
{
KeystrokeID: long,
SessionID: int,
TimeMarker: long,
Type: byte,
Data: string
},
...
]
```

Response Codes

200 - Request successful. Keystrokes are in response body.

> For more information, please see "Common Response Codes" on page 13.

Managed Account Credentials

PUT Workgroups/{workgroupName}/Assets/{assetName}/ManagedSystems/ManagedAccounts/{accountName}/Credentials

> ***Note:*** *This API has been deprecated and is available for backwards compatibility only. Use PUT Credentials?workgroupName={workgroupName}&assetName={assetName}&accountName={accountName} instead.*

Purpose

Updates the credentials for a Managed Account by Workgroup name, Asset name, and Managed Account name, optionally applying the change to the Managed System.

Required Permissions

- Password Safe Account Management (Read/Write) or
- ISA Role or Credentials Manager Role on a Smart Rule referencing the account.

URL Parameters

- **workgroupName:** Name of the Workgroup.
- **assetName:** Name of the Asset.
- **accountName:** Name of the Managed Account for which to set the credentials.

Request Body

Content-Type: application/json

```
{
Password: string,
PublicKey: string,
PrivateKey: string,
Passphrase: string,
UpdateSystem: bool
}
```

Request Body Details

- **Password:** (optional) The new password to set. If not given, generates a new random password.
- **PublicKey:** (required if PrivateKey is given and updateSystem=true) The new public key to set on the host
- **PrivateKey:** The private key to set (provide Passphrase if encrypted)
- **Passphrase:** (optional) The passphrase to use for an encrypted private key
- **UpdateSystem:** (default: true) Whether to update the credentials on the referenced system

Response Body

None

Response Codes

204 - Request Successful. No Response Body.

> ℹ For more information, please see "Common Response Codes" on page 13.

Appendix

Migration from v1 or v2

Any script or application written for v1 or v2 of the API will need some minor modifications to work with v3, namely the Authorization header and URL endpoints.

Authorization Header

In v1 and v2 the authorization header was used solely for the API Application Key. Now it is used to communicate the API Application Key as well as the RunAs username.

***Example:** C#*

```
HttpClient client = new HttpClient();

client.DefaultRequestHeaders.Add("Authorization", "PS-Auth key=C9AB26FB-B322- ED97-A34C-
215CA4D9630C; runas=doe-main\johndoe;");
```

***Example:** Powershell*

```
$headers = @{ Authorization="PS-Auth key=C9AB26FB-B322-ED97-A34C-215CA4D9630C; runas=doe-
main\johndoe;"; };
```

Endpoint Comparison

Note the use of https/SSL and removal of **PasswordSafe** segment in v3:

- v1 base endpoint: http://the-server/BeyondTrust/api/public/v1/PasswordSafe
- v2 base endpoint: http://the-server/BeyondTrust/api/public/v2/PasswordSafe
- v3 base endpoint: https://the-server/BeyondTrust/api/public/v3

Endpoint Mapping

Migration from v1

V1		V3	
Method	***Endpoint***	***Method***	***Endpoint***
GET	/v1/PasswordSafe/GetPublicKey	<deprecated>	
GET	/v1/PasswordSafe/SignIn	<deprecated>	
GET	/v1/PasswordSafe/Signout	POST	/v3/Auth/Signout

V1		V3	
Method	***Endpoint***	***Method***	***Endpoint***
GET	/v1/PasswordSafe/SignAppIn	POST	/v3/Auth/SignAppin
GET	/v1/PasswordSafe/SecureSignAppIn	<deprecated>	
GET	/v1/PasswordSafe/GetManagedAccountsList	GET	/v3/ManagedAccounts
POST	/v1/PasswordSafe/ImmediatePasswordRequest	POST	/v3/Requests
GET	/v1/PasswordSafe/GetPendingRequests	GET	/v3/Requests?status=pending
GET	/v1/PasswordSafe/GetActiveRequests	GET	/v3/Requests?status=active
POST	/v1/PasswordSafe/RetrievePassword	GET	/v3/Credentials/{requestId}
POST	/v1/PasswordSafe/RetrieveSecurePassword	<deprecated>	
POST	/v1/PasswordSafe/ReleasePassword	PUT	/v3/Requests/{requestId}/Checkin

Migration from v2

v2		v3	
Method	***Endpoint***	***Method***	***Endpoint***
GET	/v2/PasswordSafe/GetPublicKey	<deprecated>	
GET	/v2/PasswordSafe/SignIn	<deprecated>	
POST	/v2/PasswordSafe/Signout	POST	/v3/Auth/Signout
POST	/v2/PasswordSafe/SignAppIn	POST	/v3/Auth/SignAppin
POST	/v2/PasswordSafe/SecureSignAppIn	<deprecated>	
GET	/v2/PasswordSafe/GetManagedAccountsList	GET	/v3/ManagedAccounts
POST	/v2/PasswordSafe/ImmediatePasswordRequest	POST	/v3/Requests
GET	/v2/PasswordSafe/GetPendingRequests	GET	/v3/Requests?status=pending
GET	/v2/PasswordSafe/GetActiveRequests	GET	/v3/Requests?status=active
POST	/v2/PasswordSafe/RetrievePassword	GET	/v3/Credentials/{requestId}
POST	/v2/PasswordSafe/RetrieveSecurePassword	<deprecated>	
POST	/v2/PasswordSafe/ReleasePassword	PUT	/v3/Requests/{requestId}/Checkin
GET	/v2/PasswordSafe/GetWorkgroups	GET	/v3/Workgroups
POST	/v2/PasswordSafe/QueueImport	POST	/v3/Imports