News feed service v1.0

Created Jan 16, 2024 at 02:01 PM

Resource: feeds

List of available feeds

http://{device}/api/mgmt.newsfeeds/1.0/feeds

JSON

```
{
  "items": [ feed ],
  <prop>: any
}
```

Property Name	Type	Description	Notes
feeds	<object>	List of available feeds	
feeds.items	<array of <feed>>		Optional;
feeds.items[items]	<feed>	Provides information about the feed. Also gives the categories and ID's for what news can be posted.	
feeds.<prop>	<any>		Optional;

Links

feeds: get

GET http://{device}/api/mgmt.newsfeeds/1.0/feeds

Response Body
Returns a feeds data object.

Resource: feed

Provides information about the feed. Also gives the categories and ID's for what news can be posted.

http://{device}/api/mgmt.newsfeeds/1.0/feeds/items/{name}

JSON

```
{
  "name": string,
  "categories": [
    {
      "name": string,
      "items": [ summary ],
      <prop>: any
    }
  ]
}
```

Property Name	Type	Description	Notes
feed	<object>	Provides information about the feed. Also gives the categories and ID's for what news can be posted.	
feed.name	<string>	Name of the feed	Optional;
feed.categories	<array of <object>>	Feed categories	Optional;
feed.categories[items]	<object>		
feed.categories[items].name	<string>		Optional;
feed.categories[items].items	<array of <summary>>		Optional;
feed.categories[items].items[items]	<summary>	Describes the feed	
feed.categories[items].<prop>	<any>		Optional;

Links

feed: get

GET http://{device}/api/mgmt.newsfeeds/1.0/feeds/items/{name}

Response Body
Returns a [feed](#) data object.

Resource: summary

Describes the feed

http://{device}/api/mgmt.newsfeeds/1.0/feeds/items/{name}/items/{id}

JSON

```
{
  "id": string,
  "summary": string,
  <prop>: any
}
```

Property Name	Type	Description	Notes
summary	<object>	Describes the feed	
summary.id	<string>	Unique summary ID	Optional;
summary.summary	<string>	Brief user visible summary of the description	Optional;
summary.<prop>	<any>		Optional;

Links

summary: get

GET http://{device}/api/mgmt.newsfeeds/1.0/feeds/items/{name}/items/{id}

Response Body
Returns a [summary](#) data object.

Resource: news

A collection of all news on the system. It supports various query parameters for getting news. New news can be posted here.

http://{device}/api/mgmt.newsfeeds/1.0/news{?
remote_user,severity,audit_id,feed_category,feed_name,source,feed_id,limit,user,offset,start_time,end_time}

JSON

```
{
  "items": [ news_item ],
  <prop>: any
}
```

Property Name	Type	Description	Notes
news	<object>	A collection of all news on the system. It supports various query parameters for getting news. New news can be posted here.	
news.items	<array of <news_item>>		Optional;
news.items[items]	<news_item>	The schema for news items	
news.<prop>	<any>		Optional;

Links

news: get

```
GET http://{device}/api/mgmt.newsfeeds/1.0/news{?
remote_user,severity,audit_id,feed_category,feed_name,source,feed_id,limit,user,offset,start_time,end_time}
```

Response Body
    Returns a <u>news</u> data object.

news: create

```
POST http://{device}/api/mgmt.newsfeeds/1.0/news{?
remote_user,severity,audit_id,feed_category,feed_name,source,feed_id,limit,user,offset,start_time,end_time}
```

Request Body
    Provide a <u>news_item</u> data object.

Response Body
    Returns a <u>news_item</u> data object.

news: set

```
PUT http://{device}/api/mgmt.newsfeeds/1.0/news{?
remote_user,severity,audit_id,feed_category,feed_name,source,feed_id,limit,user,offset,start_time,end_time}
```

Request Body
    Provide a <u>news</u> data object.

Response Body
    Returns a <u>news</u> data object.

Resource: news_item

The schema for news items

```
http://{device}/api/mgmt.newsfeeds/1.0/news/items/{id}
```

JSON

```json
{
  "id": integer,
  "timestamp": timestamp-hp,
  "feed_name": string,
  "feed_category": string,
  "feed_id": string,
  "user": string,
  "remote_user": string,
  "audit_id": string,
  "source": string,
  "severity": string,
  "details": string,
  "resources": news_resources
}
```

Property Name	Type	Description	Notes
news_item	*<object>*	The schema for news items	Required properties: [feed_id, feed_name, feed_category, timestamp, severity, user, remote_user, audit_id, source, details];
news_item.id	*<integer>*	ID of the news item	Read-only; Optional;
news_item.timestamp	*<timestamp-hp>*	Miliseconds since epoch of news feed item	
news_item.feed_name	*<string>*	Name of the feed	
news_item.feed_category	*<string>*	Category of the feed	
news_item.feed_id	*<string>*	Registered feed identification tag	
news_item.user	*<string>*	Username for user driven feeds. Defaults to Blank for nonuser driven feeds."	

news_item.remote_user	*<string>*	Remote username for user driven feeds. This can be the SCC user or the user authenticated through TACACs. Defaults to Blank for nonuser driven feeds."	
news_item.audit_id	*<string>*	An audit itentification ID to tag related resources together. Blank string if none.	
news_item.source	*<string>*	Service sending out the news feed item	
news_item.severity	*<string>*	Severity of the news feed item	Values: low, normal, high, critical;
news_item.details	*<string>*	Detailed description of news feed item	
news_item.resources	*<news_resources>*	Additional detailed information about the news	

Links

news_item: get

GET http://{device}/api/mgmt.newsfeeds/1.0/news/items/{id}

Response Body
  Returns a [news_item](#) data object.

Relations

news_item: feed

Related resource
  [feed](#)

Variables

Related var	Data value for replacement
name	0/feed_name

news_item: summary

Related resource
  [summary](#)

Variables

Related var	Data value for replacement
name	0/feed_name
id	0/feed_id

news_item: instances

Related resource
  [news](#)

Type: news_resource

Contains JSON data supporting the news. It should be self describing, containing information to interpret the JSON data.

JSON

```
{
  "resource_id": string,
  "json_object": {
    <prop>: any
  },
  "object_tag": string
}
```

Property Name	Type	Description	Notes
news_resource	*<object>*	Contains JSON data supporting the news. It should be self describing, containing information to interpret the JSON data.	Required properties: [resource_id, object_tag, json_object];
news_resource.resource_id	*<string>*	ID of the schema containing resource	
news_resource.json_object	*<object>*	Object containing details of the news feed item: contains new object for configuration type, contains event object if based on lumberjack events framework.	
news_resource.json_object.<prop>	*<any>*		Optional;
news_resource.object_tag	*<string>*	Tag for the JSON resource	

Type: news_resources

Additional detailed information about the news

JSON

```
{
  "items": [ news_resource ],
  <prop>: any
}
```

Property Name	Type	Description	Notes
news_resources	*<object>*	Additional detailed information about the news	
news_resources.items	*<array of <news_resource>>*		Optional;
news_resources.items[items]	*<news_resource>*	Contains JSON data supporting the news. It should be self describing, containing information to interpret the JSON data.	
news_resources.<prop>	*<any>*		Optional;