LOCKSS Configuration Service REST API

API of the LOCKSS Configuration REST Service

[Contact the developer](#)
[BSD-3-Clause](#)

aus

GET	/aus	Get the configurations of all AUs

Implementation Notes

Get the configuration of all AUs

Response Class (Status 200)

The configuration of all AUs

Model | Example Value

```
[
  {
    "auConfig": {},
    "auId": "string"
  }
]
```

Response Content Type   application/json ⌄

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Unauthorized		
403	Forbidden		
404	Not Found		
500	Internal Server Error		

Try it out!

DELETE	/aus/{auid}	Delete the configuration of an AU

Implementation Notes

Delete the configuration of an AU given the AU identifier

Response Class (Status 200)

The deleted configuration of the specified AU

Model | Example Value

```
{
   "auConfig": {},
   "auId": "string"
}
```

Response Content Type   application/json ∨

Parameters

Parameter	Value	Description	Parameter Type	Data Type
auid	(required)	**The identifier of the AU for which the configuration is\ \ to be deleted**	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
204	No Content		
400	Bad Request		
401	Unauthorized		
403	Forbidden		
404	Not Found		
500	Internal Server Error		

Try it out!

GET   /aus/{auid}   Get the configuration of an AU

Implementation Notes

Get the configuration of an AU given the AU identifier

Response Class (Status 200)

The configuration of the specified AU

Model | Example Value

```
{
    "auConfig": {},
    "auId": "string"
}
```

Response Content Type   application/json ∨

Parameters

Parameter	Value	Description	Parameter Type	Data Type
auid	(required)	**The identifier of the AU for which the configuration is\ \ requested**	path	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Bad Request		
401	Unauthorized		
403	Forbidden		
404	Not Found		
500	Internal Server Error		

Try it out!

| PUT | /aus/{auid} | Store the configuration of an AU |

Implementation Notes
Store the configuration of an AU given the AU identifier

Response Class (Status 200)
The Archival Unit configuration after it is stored

Model | Example Value

```
{
    "auConfig": {},
    "auId": "string"
}
```

Response Content Type application/json ⌄

Parameters

Parameter	Value	Description	Parameter Type	Data Type
auConfiguration	(required)	**The Archival Unit configuration to be stored**	body	Model | Example Value

Parameter content type:

application/json ⌄

Model \| Example Value

```
{
  "auConfig": {},
  "auId": "string"
}
```

Response Messages

HTTP Status Code	Reason	Response Model	Headers
201	Created		
400	Bad Request		
401	Unauthorized		
403	Forbidden		
404	Not Found		
415	Unsupported Media Type		
500	Internal Server Error		

Try it out!

config

GET /config/file/{sectionName}   Get the named configuration file

Implementation Notes
Get the configuration file stored for a given name

Response Class (Status 200)
The named configuration file

Model \| Example Value

```
{
  "description": "string",
  "filename": "string",
```

```
    "inputStream": {},
    "open": true,
    "readable": true,
    "uri": {
      "absolute": true,
      "authority": "string",
      "fragment": "string",
      "host": "string",
      "opaque": true,
      "path": "string",
      "port": 0,
      "query": "string",
      "rawAuthority": "string",
      "rawFragment": "string",
      "rawPath": "string",
      "rawQuery": "string",
      "rawSchemeSpecificPart": "string",
      "rawUserInfo": "string",
      "scheme": "string",
      "schemeSpecificPart": "string",
      "userInfo": "string"
    },
    "url": {
      "authority": "string",
      "content": {},
      "defaultPort": 0,
      "file": "string",
      "host": "string",
      "path": "string",
      "port": 0,
      "protocol": "string",
      "query": "string",
      "ref": "string",
      "userInfo": "string"
    }
  }
```

Response Content Type   multipart/form-data ⌄

Parameters

Parameter	Value	Description	Parameter Type	Data Type
sectionName	(required)	**The name of the section for which the configuration file\ \ is requested**	path	string
Accept	(required)	**The Accept header**	header	string

Parameter	Value	Description	Parameter Type	Data Type
If-Match		The If-Match header	header	string
If-Modified-Since		The If-Match header	header	string
If-None-Match		The If-Match header	header	string
If-Unmodified-Since		The If-Match header	header	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
304	Not Modified		
400	Bad Request		
401	Unauthorized		
403	Forbidden		
404	Not Found		
406	Not Acceptable		
412	Precondition failed		
500	Internal Server Error		

Try it out!

PUT   /config/file/{sectionName}   Store the named configuration file

Implementation Notes
Store the configuration file for a given name

Parameters   🛑

Parameter	Value	Description	Parameter Type	Data Type
sectionName	(required)	**The name of the section for which the configuration file\ \ is to be stored**	path	string
configFile	Browse...	file detail	formData	file

Parameter	Value	Description	Parameter Type	Data Type
If-Match		The If-Match header	header	string
If-Modified-Since		The If-Match header	header	string
If-None-Match		The If-Match header	header	string
If-Unmodified-Since		The If-Match header	header	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
200	OK		
201	Created		
400	Bad Request		
401	Unauthorized		
403	Forbidden		
404	Not Found		
412	Precondition failed		
415	Unsupported Media Type		
500	Internal Server Error		

Try it out!

GET /config/lastupdatetime   Get the timestamp when the configuration was last updated

Implementation Notes
Get the timestamp when the configuration was last updated

Response Class (Status 200)
date-time

Response Content Type application/json ▼

Response Messages

HTTP Status Code	Reason	Response Model	Headers

HTTP Status Code	Reason	Response Model	Headers
401	Unauthorized		
403	Forbidden		
404	Not Found		
500	Internal Server Error		

Try it out!

GET **/config/loadedurls**   Get the URLs from which the configuration was loaded

Implementation Notes

Get the URLs from which the configuration was actually\ \ loaded, reflecting any failover to local copies

Response Class (Status 200)

The URLs from which the configuration was loaded

Model | Example Value

```
[
  "string"
]
```

Response Content Type   application/json ⌄

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Unauthorized		
403	Forbidden		
404	Not Found		
500	Internal Server Error		

Try it out!

PUT **/config/reload**   Request a configuration reload

Implementation Notes

Request that the stored configuration is reloaded

Response Messages ⊗

HTTP Status Code	Reason	Response Model	Headers
200	OK		
201	Created		
401	Unauthorized		
403	Forbidden		
404	Not Found		
500	Internal Server Error		

[Try it out!]

GET	/config/url	Get the configuration file for a URL

Implementation Notes
Get the configuration file stored for a given URL

Response Class (Status 200) ⊗
The configuration file for the URL

Model | Example Value

```
{
  "description": "string",
  "filename": "string",
  "inputStream": {},
  "open": true,
  "readable": true,
  "uri": {
    "absolute": true,
    "authority": "string",
    "fragment": "string",
    "host": "string",
    "opaque": true,
    "path": "string",
    "port": 0,
    "query": "string",
    "rawAuthority": "string",
    "rawFragment": "string",
    "rawPath": "string",
    "rawQuery": "string",
    "rawSchemeSpecificPart": "string",
    "rawUserInfo": "string",
    "scheme": "string",
```

```
      "schemeSpecificPart": "string",
      "userInfo": "string"
    },
    "url": {
      "authority": "string",
      "content": {},
      "defaultPort": 0,
      "file": "string",
      "host": "string",
      "path": "string",
      "port": 0,
      "protocol": "string",
      "query": "string",
      "ref": "string",
      "userInfo": "string"
    }
  }
```

Response Content Type   multipart/form-data ⌄

Parameters

Parameter	Value	Description	Parameter Type	Data Type
url	(required)	**The URL for which the configuration is requested**	query	string
Accept	(required)	**The Accept header**	header	string
If-Match		The If-Match header	header	string
If-Modified-Since		The If-Match header	header	string
If-None-Match		The If-Match header	header	string
If-Unmodified-Since		The If-Match header	header	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
304	Not Modified		
400	Bad Request		
401	Unauthorized		

HTTP Status Code	Reason	Response Model	Headers
403	Forbidden		
404	Not Found		
406	Not Acceptable		
412	Precondition failed		
500	Internal Server Error		

Try it out!

config-api-controller : the config API

GET   /status   getStatus

Response Class (Status 200)
OK

Model | Example Value

```
{
   "ready": true,
   "version": "string"
}
```

Response Content Type   application/json ⌄

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Unauthorized		
403	Forbidden		
404	Not Found		

Try it out!

[BASE URL: / , API VERSION: 1.0.0]