

Resolve URL API

Background

As part of the issue <https://github.com/vector-im/riot-web/issues/3015> it would require to have the way to resolve the drag & drop link from the other browser window. It can't be handled by client and downloaded locally because of CORS restrictions.

Scenario

As user I want to have the way to drag & drop link under the different domains to chat room

When user logged in and open the chat room

And drag & drop from the different browser(tab) image(file, url) to content view of chat room

Then user should see the preview of dropped url

And user should see the url to download the resource

When user pressed by the url to download the resource

Then user should see save file to the local file system

Proposal

Add new endpoint to resolve url.

Example:

=> *POST*

/_matrix/media/v1/resolve_url

Content-Type: application/json

{ "url": "<media-url>" }

<=

Content-Type: application/json

```
{ "content_uri": "mxc://<server-name>/<media-id>", "msgtype": "<msgtype>" }
```

Expected behaviour by using `resolve_url`:

- should have authentication middleware before to process the request
- should run validation before to process url
- should download resource to media repository by passed url
- should respond with matrix url (`content_uri`) and message type to preview the url in case if `m.image`, `m.video`, `m.audio`, `m.file` resource (`m.file` is not previewable from the client side, it would be downloadable resource by `content_uri`).
- should respond with errors (briefly describing the issue happened on resolving the url).

Validation rules:

- should be restricted by max upload size from the config
- should be restricted by url blacklist (in case we have url pattern match we should skip resolve and upload downloadable resource and respond with error)

List of errors and response codes in case failed validation:

- 403 - failed authentication by user `access_token` from header or query param
- 404 - Missing url parameter that should be passed as body of request
- 403 - URL blocked by url pattern blacklist entry
- 404 - Not found resource to resolve `<URL>`
- 400 - Can't resolve `<URL>` with missing content length information
- 413 - Upload request body is too large for `<URL>`

-