Bulk Download From SSO Website With Curl

Abstract

SSOサイトからファイルをダウンロードする手順

Introduction

katanaでリストを作成して, curlでファイルを一括ダウンロードします。
Basic認証の場合ではcurl -u ユーザ名:パスワードでサイトにログインしてファイルを取得することはできますが、SSOの場合は認証エラーになります。config(cookie)を作成してヘッダに含めることで解決します。

Methodologies

Prerequisites

- GitHub - projectdiscovery/katana: A next-generation crawling and spidering framework.

Steps

URLの確認

一般的に以下のようなサイト構造を想定します。

1. ベースURL:
    - `https://www.templatebank.com/`
2. カテゴリ・タグのページ(コンテンツ一覧):
    - `https://www.templatebank.com/category/proposal-templates`
    - `https://www.templatebank.com/category/contract-templates?page=2`
    - `https://www.templatebank.com/tag/e-contract`
3. ダウンロードリンクのあるページ:
    - `https://www.templatebank.com/contents/outsourcing-agreement-contingent-fees`
4. ダウンロードリンク:
    - `https://www.templatebank.com/downloadfile?18102`

リクエストボディの取得

1. ChromeからSSOのページを開き、ログイン処理を行い、SSOを完了します。
2. F12キーを押下して開発者ツールを開き、ダウンロードリンクのあるページ、で更新します。
3. 1つ目のファイルをダウンロードします。
4. `Network`タブを選択します。
5. Name一覧に表示されるリクエスト(基本的に、最初の項目)を右クリックして`Copy-Copy as cURL(bash)`か利用する形式を選択します。

6. 下記のような内容を、`curl.conf`などと保存します。

```
curl 'https://www.templatebank.com/contents/outsourcing-agreement-contingent-fees' \
  -H 'accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8' \
  -H 'accept-language: en-US,en;q=0.5' \
  -H 'cache-control: max-age=0' \
  -H 'cookie: ASP.NET_SessionId=egcpcn1raig0nc35skdrd4ri; MemID=3986621; MFlag=1; Mail2=example%40protonmail%2Ecom; ASPSESSIONIDQAQQTACC=EGHHMAOAPMAJLOFKPJBAKGAK' \
  -H 'priority: u=0, i' \
  -H 'referer: https://www.templatebank.com/contents/outsourcing-agreement-contingent-fees' \
  -H 'sec-ch-ua: "Chromium";v="128", "Not;A=Brand";v="24", "Brave";v="128"' \
  -H 'sec-ch-ua-mobile: ?0' \
  -H 'sec-ch-ua-platform: "Windows"' \
  -H 'sec-fetch-dest: document' \
  -H 'sec-fetch-mode: navigate' \
  -H 'sec-fetch-site: same-origin' \
  -H 'sec-fetch-user: ?1' \
  -H 'sec-gpc: 1' \
  -H 'upgrade-insecure-requests: 1' \
  -H 'user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.0.0 Safari/537.36'
```

configの編集・実行

`curl.conf`をRFC7230に準拠した形式へ編集します。

```
# Remove -i for dry-run
sed -i \
-e 's/^\s*-H\s*//' \
-e 's/ \\\//g' \
-e 's/: /=/' \
-e "s/'//g" \
-e 's/"//g' \
-e 's/\\\s*$//' \
-e 's/\s*$//' \
-e 's/=\(.*\)/ = "\1"/' \
# -e 's/"/\\\"/g' \
curl.conf2
```

- cookie, (referer, user-agent) を残す

編集後

```
cookie = "ASP.NET_SessionId=egcpcn1raig0nc35skdrd4ri; MemID=3986621; MFlag=1; Mail2=example%40protonmail%2Ecom; ASPSESSIONIDQAQQTACC=EGHHMAOAPMAJLOFKPJBAKGAK"
user-agent = "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/128.0.0.0 Safari/537.36"
```

作成したconfを使い、ドメイン一覧を取得します。

```
set base_url "https://www.templatebank.com"
katana -u $base_url -d 8 -H curl.conf -jc -retry 3 -delay 1 -p 50 -c 50 -
store-response
```

対象となるダウンロードリンクを特定します。

```
rg '/Download\?\d+' -g '*.txt' -0 -l |
   parallel -0 -k -j9 'cat {}' |
   pup -p a json{} |
   jq -r '[.. | .href? | select(. != null and test("/Download\\\\?[0-
9]+"))]|unique' |
   tee endp.txt

jq -r '.[]' endp.txt | wc -l
```

ファイルをダウンロードします。
(/Download?を適切に再処理する。)

```
sed -i 's/\/Download?//g' endp.txt
  set base_url "https://www.templatebank.com/Download?"
  jq -r '.[]' endp.txt |
      parallel "curl -vJLO -K curl.conf \"$base_url{}\" --data-raw \"fileId=
{}\""
# --ssl-no-revoke \
# --data-raw 'contentsInfoId=2300&fileId=18102&categoryId=110' \
```

`--data-raw` に組み合わせが必要な場合

それぞれのリストを用いて、総当たり的に試してみる

```
# See previous step for jq '...' :
# rg '/Download\?(\d|\w)+' -g '*.txt' -0 -l | parallel -0 -k -j9 'cat {}' |
pup -p a json{} |
   # jq -r '[.. | .href? | select(. != null and test("/Download\\\\?[0-9a-zA-
Z_-]+"))]|unique' |
   # tee file_ids.txt

# rg '/contents/(\d|\w)+' -g '*.txt' -0 -l | parallel -0 -k -j9 'cat {}' | pup
-p a json{} |
   # jq -r '[.. | .href? | select(. != null and test("/contents/[0-9a-zA-
Z_-]+"))]|unique' |
   # tee contents_ids.txt

# rg '/category/(\d|\w)+' -g '*.txt' -0 -l | parallel -0 -k -j9 'cat {}' | pup
-p a json{} |
#    jq -r '[.. | .href? | select(. != null and test("/category/[0-9a-zA-
Z_-]+"))] | unique' |
#    tee category_ids.txt
```

```
# ファイルからリストを読み込む
file_ids=$(jq -r '.[] | select(contains("/Download"))' file_ids.txt)
category_ids=$(jq -r '.[] | select(contains("/category"))' category_ids.txt)
contents_urls=$(jq -r '.[] | select(contains("/contents/"))' contents_ids.txt)

extract_id_from_url() {
  local url=$1
  echo "$url" | grep -oP '(?<=/)\d+'
}

# 全ての組み合わせを生成
for contents_url in $contents_urls; do
  contents_id=$(extract_id_from_url "$contents_url")
  for file_id in $file_ids; do
    for category_id in $category_ids; do
      # Example of how you might use these IDs in a request
      echo "Contents ID: $contents_id, File ID: $file_id, Category ID:
$category_id"
      # Replace with your request logic
      # curl -X POST --data-raw
"contentsInfoId=$contents_id&fileId=$file_id&categoryId=$category_id" ...
    done
  done
done
```

- ripgrepと違って、JqではEscapeSequenceが使用できないので、文字列全体などの表現に差異があります。

Key Adjustments:

1. `-H 'accept: text/csv'`: Adjusted to match the expected content type (`text/csv`). You can change this depending on the actual file type you are downloading.
2. **Removed** `User-Agent` **and** `Referer`: These headers are optional and removed for simplicity unless they are absolutely required for the download.
3. `--ssl-no-revoke`: Ensures secure SSL connections.
4. `--data-raw`: Used to pass necessary POST data for the file download request.
5. `-vJLO`: Options for verbose output, continuing downloads, and saving to a file.

References

1. Add `-no-clobber` to avoid overwriting files that already exist · Issue #749 · projectdiscovery/katana
2. curl to get a file with correct name when redirected
3. curl - How To Use
4. Logical AND between glob patterns · Issue #1046 · BurntSushi/ripgrep
5. CURL to access a page that requires a login from a different page
6. Authentication - everything curl