Aspider Prerequisites

Crawling a Website with Aspider

If you want to crawl a website with Aspider, you need to make sure you have the following points covered.

1. The Aspire Servers must have access to the seed(s) URL (configured in the content source configuration).
    a. Try pinging the website server or requesting the seed URL using curl.
2. Check for any credentials that may be needed to access the sites to be crawled.

Which Authentication Mechanism Should be Used?

There are two types of authentication: cookie-based authentication and HTTP-based authentication.

1. Cookie-based are the ones that redirect you to a login page for you to login.
   You are probably facing a cookie-based authenticated site if the first line of the result is a redirection such as "**HTTP/1.1 302 Found**".

2. HTTP based are the ones that prompt you for credentials in the browser without even displaying a login page.
   You are probably facing with a HTTP based authentication if the first line of the result is "**HTTP/1.1 401 Unauthorized**".

The first thing you should do is execute a curl over the seed URL:

```
$ curl -i http://mysitehost/mysite
```

Cookie-based authentication

If your site requires this kind of authentication, then you need to know certain details about it in order to configure Aspider to crawl it.

1. The login form page is where the login form is located, some sites redirect you here if you are not authenticated, in that case, if you execute the curl command from above, it will most likely be the URL specified in the "**Location**" response Header.
2. The login form HTML structure is the path for where to find the HTML form that Aspider needs (in order to fill and send) to authenticate.
    a. For example, if your login form page consists of the following HTML, then your path would be "div > form". You can use Google Chrome inspect mode to generate this CSS Selector for you.

```
<html>
<head>....</head>
<body>
    <div ....>
    <form method="post" action="...." ...
       ...
    </form>
    </div>
</body>
</html>
```

    b. Also identify the **name** attribute of the **user** and **password** input elements.

SSL Support

Aspider supports the following versions of SSL:

- TLSv1
- TLSv1.1
- TLSv1.2

Note: *SSLV2 and SSLv3 are not supported.*

HTTP-based authentication

If your site requires this type of authentication, then you need to determine which authentication scheme to use.

If you executed the curl command from above, you can determine the authentication scheme by looking at the "**WWW-Authenticate**" headers.

Aspider supports the following schemes:

- Basic

- Digest
- NTLM
- Negotiate/Kerberos

Some sites have two "**WWW-Authenticate**" headers in their response. The first one corresponds to the preferred schema and the second one is for fail-over. This is done because some browsers don't support the preferred authentication scheme. For Aspider, you can use either one of the mechanisms if it are a supported schema (mentioned above).

Some schemas require a **realm** to work. If you see a **realm** inside of the response headers, then use that in the configuration.

Basic/Digest/NTLM

Try your credentials with curl:

```
$ curl -I --<basic/digest/ntlm> -u <username> <the-seed-url>
```

The command will prompt you for the password. If the response header displays as shown below you have the correct credentials.

```
HTTP/1.1 200 OK
Content-Length: <XXX>
Content-Type: text/html
Last-Modified: <XXXXXXXXXXXXXXXXX>
```

Negotiate/Kerberos

If you want to use the Negotiate/Kerberos authentication scheme, then you need to find the "Key Distribution Center" (KDC). This is a service that supplies session tickets and temporary session keys to users and computers within an Active Directory domain. If you don't know your KDC address, do as follows:

- **On Windows CMD or Powershell**

```
> nltest /dsgetdc:<domain.name>
```

The KDC address will appear in the first line as "DC: *<the KDC address>*"

- **On Linux bash**

```
$ cat /etc/krb5.conf
```

Look for something like:

```
    [realms]
        TESTDOM.LAN = {
                kdc = DC1.TESTDOM.LAN
                admin_server = DC1.TESTDOM.LAN
        }
```