Bravura Security Fabric

Secure Browser

Hitachi ID Bravura Privilege offers a "Secure browser" disclosure that allows you to record brokered access to websites. This feature is similar to the web app privileged sign-on disclosure with the added session monitoring security feature.

This document shows you how to configure and use the secure browser disclosure.

Terminology

Web application administrator User who has access to website disclosure configuration pre-defined request (PDR)s. This person will need to be a member of the PAM_TEAM_ADMINS user class, and will need to be configured by a product administrator.

Secure browser A web app disclosure configured with session monitoring.

1 Business requirement

Organizations need to broker access to hundreds and thousands of websites. Users need access from any browser without having to provide administrator credentials. When accessing these websites, session monitoring can enhance the organization's ability to audit access.

Users with session monitoring have access to view in-progress and completed sessions. Access to download recorded packages can also be configured.

2 Solution

The secure browser disclosure allows organizations to broker access to hundreds and thousands of web applications, and record the session.

The access settings for web applications are configured in a json file that is loaded when the web application is created in the product. This secure browser disclosure launches a new pop-up and provided automatic login to the website without need to enter administrator credentials for the managed account using the configuration file. Sessions are recorded at launch.

3 Requirements

3.1 System requirements

- Windows 10 Version 1809 is the minimum required version

4 Deployment

The following components are available in *Hitachi ID Bravura Pattern: Privileged Access Edition*, and can be configured as a product administrator or through pre-defined request (PDR)s:

- Scenario.pam_webapp_management
- Scenario.pam_webapp_social (optional)
- Configuration file (json) - See Configuration file

The following access disclosures are available:

- Secure browser (securebrowser)

5 Installing the client

To install the client:

1. Unzip the package provided in C:\Program Files\Hitachi ID\IDM Suite\<instance>\addon\securebrowser.
2. Run the SessmonBrowser.UWP_*_x86_x64_arm.msixbundle file, where * is the version number.
3. Click the Install button.

Troubleshooting

- Failed to install with the following or similar error:

  ```
  App installation failed with error message: error 0x80073D02: Unable to
  install because the following apps need to be closed
  Microsoft.DesktopAppInstaller_1.0.30251.0_x64__8wekyb3d8bbwe. (0x80073d02)
  ```

  - Verify if dependencies are being downloaded and installed correctly.
    - Check app updates can be downloaded and installed via the Microsoft store.
    - Check DNS.
  - Manually install dependencies in Dependencies/x64 either by double clicking or through PowerShell:
    - Microsoft.NET.Native.Framework.2.2.appx
    - Microsoft.NET.Native.Runtime.2.2.appx
    - Microsoft.UI.Xaml.2.4.appx
    - Microsoft.VCLibs.x64.14.00.appx

5.1 Uninstalling the client

To uninstall via the Start menu:

1. In the Start menu, find **Hitachi ID Secure Browser**.
2. Right-click then click **Uninstall**.

To uninstall through Apps & features:

1. Open "Apps & features".
2. Search for **Hitachi ID Secure Browser**
3. Uninstall.

6 Use case: Configuring a website disclosure

Web application administrators (web app admins) can create, update and delete website disclosure configurations for broking access to specific websites.

Additional requirements

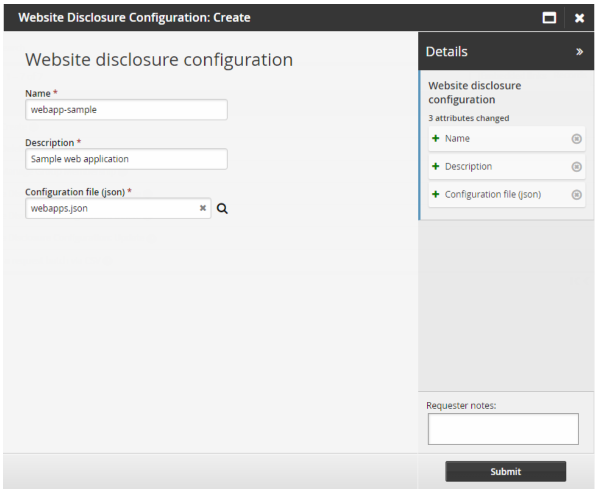
This use case assumes that:

- A web app admin has been added to the PAM_TEAM_ADMINS user class and thus has the privilege to configure website disclosure configurations.
- A configuration file is available and complete.

Create a website disclosure

To create a website disclosure:

1. Log in to *Bravura Security Fabric* as a web app admin.
2. Select **Manage Resources**.
3. Select **Website Disclosure Configuration: Create**.
4. Enter a **Name**, **Description**, and select a **Configuration file**.



Website Disclosure Configuration: Create

Website disclosure configuration

Name *: webapp-sample

Description *: Sample web application

Configuration file (json) *: webapps.json

5. Click **Submit**.

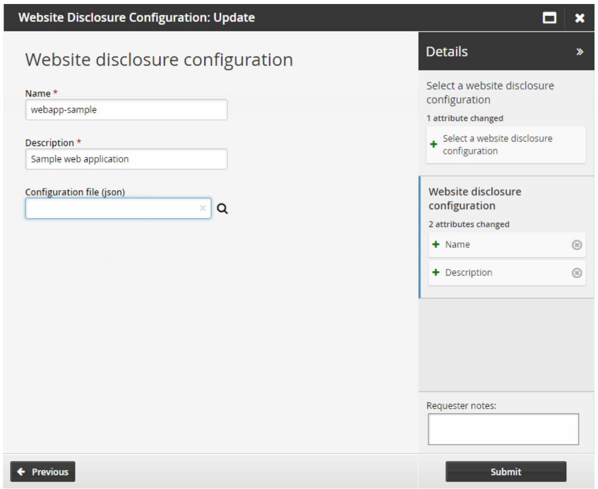
The request should be automatically approved.

Update a website disclosure configuration

To update a website disclosure:

1. Log in to *Bravura Security Fabric* as a web app admin.
2. Select **Manage Resources**.
3. Select **Website Disclosure Configuration: Update**.
4. Select the website disclosure configuration you want to update.
5. Click **Next**.
6. Update the **Name**, **Description**, and/or **Configuration file**.

> **Note:** Each website disclosure configuration is uniquely identified by Hitachi Bravura Privilege, so changing the name will not modify which website disclosure configuration is used.



Website Disclosure Configuration: Update

Website disclosure configuration

Name * webapp-sample

Description * Sample web application

Configuration file (json)

Details

Select a website disclosure configuration

1 attribute changed

+ Select a website disclosure configuration

Website disclosure configuration

2 attributes changed

+ Name

+ Description

Requester notes:

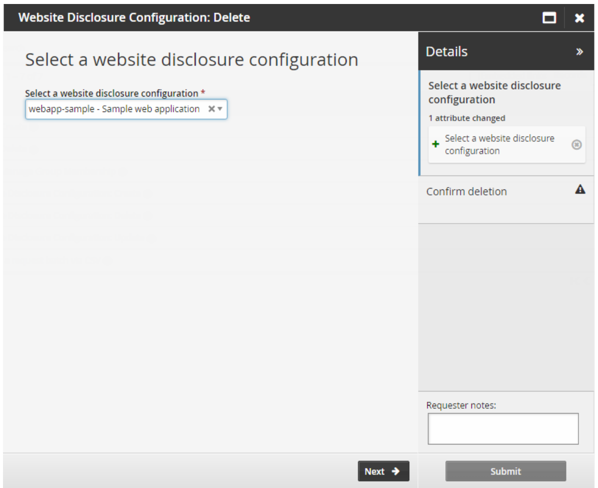
7. Click **Submit**.

The request should be automatically approved.

Delete a website disclosure configuration

To delete a website disclosure configuration:

1. Log in to *Bravura Security Fabric* as a web app admin.
2. Select **Manage Resources**.
3. Select **Website Disclosure Configuration: Delete**.
4. Select the website disclosure configuration you want to delete.



Website Disclosure Configuration: Delete

Select a website disclosure configuration

Select a website disclosure configuration * webapp-sample - Sample web application

Details

Select a website disclosure configuration

1 attribute changed

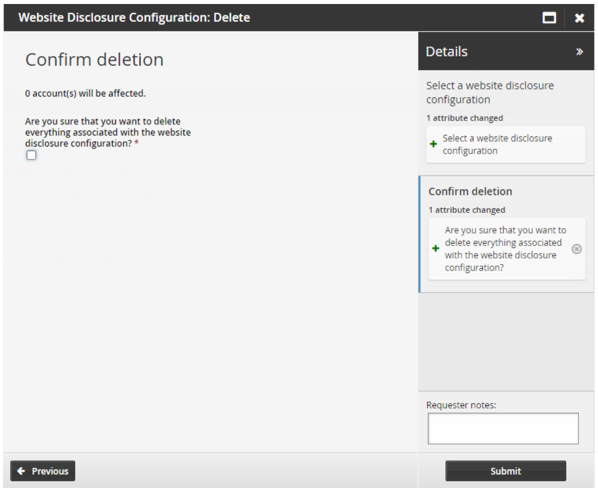
+ Select a website disclosure configuration

Confirm deletion

Requester notes:

5. Click **Next**.
6. Confirm delete.

Note: Number of accounts using this website disclosure will be displayed.



Website Disclosure Configuration: Delete

Confirm deletion

0 account(s) will be affected.

Are you sure that you want to delete everything associated with the website disclosure configuration? *

□

Details
Select a website disclosure configuration
1 attribute changed
+ Select a website disclosure configuration
Confirm deletion
1 attribute changed
+ Are you sure that you want to delete everything associated with the website disclosure configuration?
Requester notes:
Previous
Submit

7. Click **Submit**.

The request should be automatically approved.

7 Use case: Configuring a team account to use secure browser

Once website disclosures are configured, team accounts can select them as a disclosure option. By default, this is only available for team vault accounts.

Additional requirements

This use case assumes that:

- A team has been created and configured
- A vault trustee has been configured for a team
- A team vault has been configured
- A website disclosure configuration is configured and available

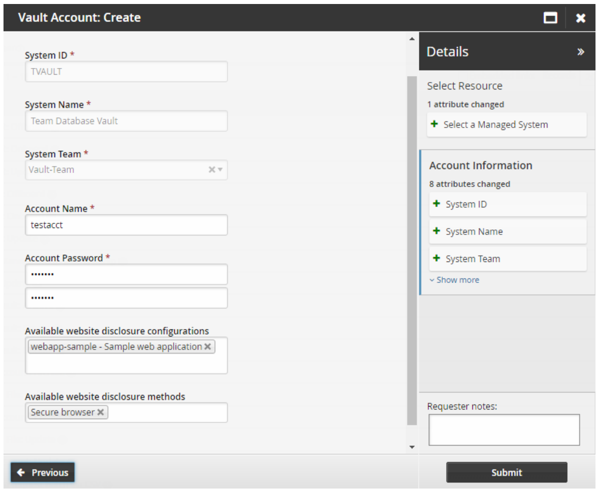
Note: Built-in social website disclosure configurations for GMail, Facebook, and Twitter are available. Install the Scenario.pam_webapp_social component to use these.

Create a vault account to use secure browser

To create a vault account to use secure browser:

1. Log in to *Bravura Security Fabric* as a vault trustee.

2. Click **Manage Resources**.
3. Click **Vault Account: Create**.
4. Select a managed system.
5. Click **Next**.
6. Enter **Account Name**.
7. Enter **Account Password** and confirm.
8. Select one or more **Available website disclosure configurations**.
9. Select **Secure browser** from the **Available website disclosure methods**.



Vault Account: Create

System ID * TVAULT

System Name * Team Database Vault

System Team * Vault-Team

Account Name * testacct

Account Password * ••••••• •••••••

Available website disclosure configurations: webapp-sample - Sample web application

Available website disclosure methods: Secure browser

Details

Select Resource

1 attribute changed

- Select a Managed System

Account Information

8 attributes changed

- System ID
- System Name
- System Team

Requester notes:

10. Click **Submit**.

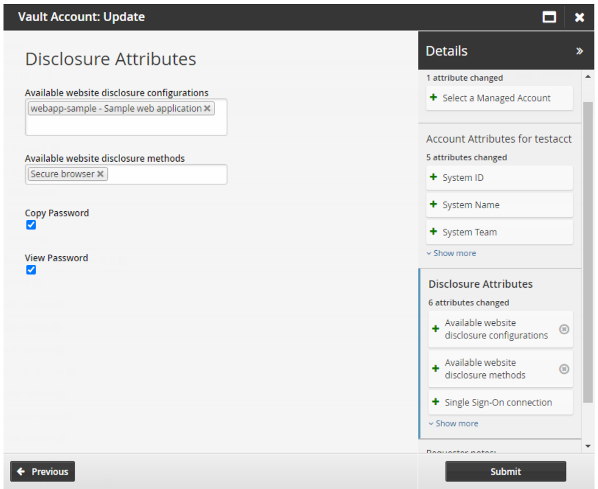
The request should be automatically approved if submitted by the team's vault trustee. Otherwise the appropriate trustee will need to approve the request.

Update a vault account to use different website disclosure configurations

To update a vault account to use secure browser:

1. Log in to *Bravura Security Fabric* as a vault trustee.
2. Click **Manage Resources**.
3. Click **Vault Account: Update**.
4. Select a managed account.
5. Click **Next**.
6. *Optional*: Update the password.
7. Click **Next**.

8. Select/deselect one or more **Available website disclosure configurations**.



Vault Account: Update

Disclosure Attributes

Available website disclosure configurations: webapp-sample - Sample web application

Available website disclosure methods: Secure browser

Copy Password ☑

View Password ☑

9. Click **Submit**.

The request should be automatically approved if submitted by the team's vault trustee. Otherwise the appropriate trustee will need to approve the request.

8 Use case: Disclosing website access to a user

Additional requirements

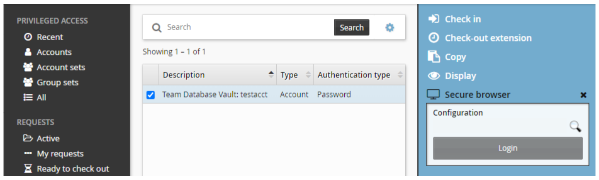
This use case assumes that:

- A team account has been created and configured.
- A vault trustee has been configured for a team.
- A team vault has been configured.
- A website disclosure configuration has been configured and is available.
- A vault account has been configured.

To disclose a website:

1. Log in to *Bravura Security Fabric* as an end user with requester privileges.
2. Select **Privileged access**.
3. Search and select the vault account configured with secure browser disclosure.
4. Submit a request for access. Wait for approval as required through your organization's process.

5. Check out the account access.
6. Click **Secure browser**.

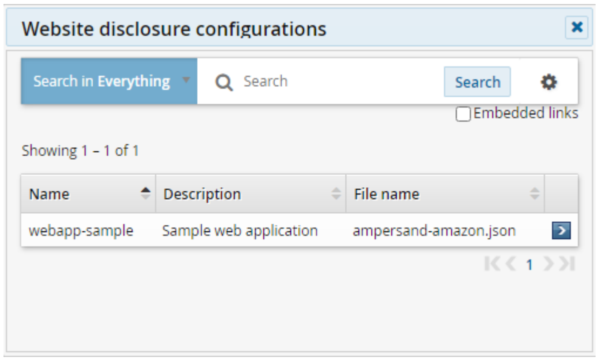


Showing 1 - 1 of 1

	Description	Type	Authentication type
☑	Team Database Vault: testacct	Account	Password

Configuration

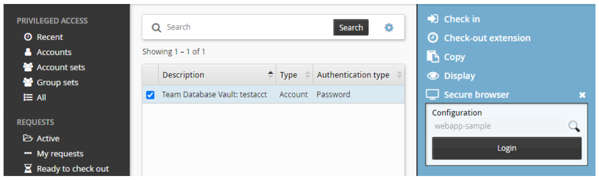
7. Click the search icon 🔍.
8. Select **Configuration file**.



Showing 1 - 1 of 1

Name	Description	File name
webapp-sample	Sample web application	ampersand-amazon.json

9. Click **Login**.



Showing 1 - 1 of 1

	Description	Type	Authentication type
☑	Team Database Vault: testacct	Account	Password

Configuration: webapp-sample

10. Open Hitachi ID Secure Browser.

9 Use case: Auditing recorded sessions

Additional requirements

This use case assumes that:

1. A team has been created and configured.
2. A vault trustee has been configured for a team.
3. A team vault has been configured.
4. A website disclosure configuration is configured and available.
5. A vault account has been configured.
6. The vault account has been checked out with secure browser disclosed access.

> **Note:** Only screenshots are recorded.

To audit recorded sessions:

1. Log in to *Bravura Security Fabric* as an end user with session monitoring privileges.
2. Open the *Session monitor* app.
3. Search for and select the vault account session.
4. Click **View** to view the session.

   > **Note:** Sessions can be in-process, in which case a short delay exists.

5. Click **Request download** to submit a request to download the package.

> **Note:** Searching, viewing, and downloading sessions are separate privileges that can be granted.

To revoke access while viewing an in-progress session:

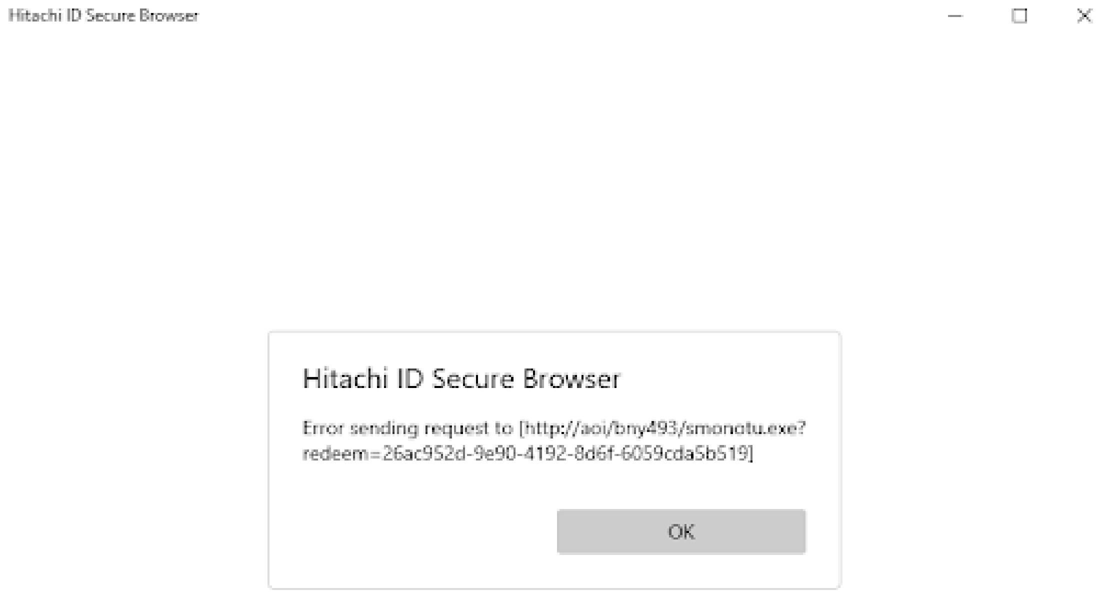
1. Log in to *Bravura Security Fabric* as an end user with session monitoring privileges.
2. Open the *Session monitor* app.
3. Search for and select the vault account session.
4. Click **View**.
5. Click **Check in access**.
6. Confirm the check in.

10 SSL enforcement

By default, users can only use web app privileged sign-on or secure browser disclosures if accessing *Bravura Security Fabric* using SSL (HTTPS). If the insecure HTTP method is used, the user will see a 'Web App disclosure enforces SSL' message and will not be able to access the web app. This behavior is controlled by the **PAM WEBAPP ENFORCE SSL** system variable.

11 Troubleshooting

Issue: Unable to connect to site



Hitachi ID Secure Browser

Error sending request to [http://aoi/bny493/smonotu.exe?redeem=26ac952d-9e90-4192-8d6f-6059cda5b519]

Fix: Update external address

Method 1:

Update the registry under `HKLM\SOFTWARE\Hitachi ID\IDM Suite\<instance>\`, add new String Value: `serveraddressexternal = <IP>`.

Method 2:

1. Log in to *Bravura Security Fabric* as superuser.
2. Navigate to **Manage the system** → **Maintenance** → **Database replication**.
3. Select node.
4. Update "External adress" to <*IP*>.
5. Restart the Database Service (iddb).

Appendices

A Configuration file

The configuration file for the web app privileged sign-on disclosure requires a JSON file with the following structure:

```
{
    "info": {
        "url": "https://webpage/"
        },
    "username": {
        "order": 1,
        "type": "CSS",
        "value": "input[name='session[username_or_email]']",
        "input_value": "%username%",
        "keypress_event": true,
        "click": false,
        "settle_time_before": 5,
        "settle_time_after": 0,
      "stop_on_fail": false
    },
    "password": {
        "order": 2,
        "type": "CSS",
        "value": "input[name='session[password]']",
        "input_value": "%password%" ,
        "keypress_event": true,
        "click": false,
        "settle_time_before": 0,
        "settle_time_after": 0,
        "stop_on_fail": false
    },
    "submit": {
        "order": 3,
        "type": "CSS",
        "value": "[role='button'][type='submit']",
        "input_value": "" ,
        "keypress_event": false,
        "click": true,
        "settle_time_before": 0,
        "settle_time_after": 0,
        "stop_on_fail": false
    }
}
```

Where:

"Info"["url"] is the url the web app will open the webpage on.

"username" , **"password"**, and **"submit"** action keys with a structure which determines how the browser will interact with the webpage.

You can have as many action keys as needed to interact with the webpage tab

"order" order number the browser extension will act on.

"type" the search method the browser extension will use to find an element that matches "value" on the webpage. The supported types are:

- "XPATH": find an element using XPATH that matches the "value"
- "CSS": find an element using CSS that matches the "value"
- "ID": find an element with "id" that matches the "value"
- "NAME": find the first element with "name" that matches the "value"
- "CLASS": find the first element with the class that matches the "value"

"value" browser extension will use to look the webpage by "type".

"input_value" the value we will look for in the search to identify the input field.

"keypress_event" a flag that is required, but not yet used.

"click" a flag that determines if the browser extension will click on the "input_value" element.

"settle_time_before" the time in seconds the browser extension waits before accessing this element.

"settle_time_after" the time in seconds the browser extension waits after accessing this element.

"stop_on_fail" a flag that stops the continued execution if the browser extension fails to find the element set by "value".

A.1 Sample configuration file for web app privileged sign-on (twitter.json)

```
{
    "info": {
        "url": "https://twitter.com/"
        },
    "username": {
        "order": 1,
        "type": "CSS",
        "value": "input[name=’session[username_or_email]’]",
        "input_value": "%username%",
        "keypress_event": true,
        "click": false,
        "settle_time_before": 5,
        "settle_time_after": 0,
        "stop_on_fail": false
    },
    "password": {
        "order": 2,
        "type": "CSS",
        "value": "input[name=’session[password]’]",
        "input_value": "%password%" ,
        "keypress_event": true,
        "click": false,
        "settle_time_before": 0,
        "settle_time_after": 0,
        "stop_on_fail": false
    },
    "submit": {
        "order": 3,
        "type": "CSS",
        "value": "[role=’button’][data-testid=’LoginForm_Login_Button’]",
        "input_value": "" ,
        "keypress_event": false,
        "click": true,
        "settle_time_before": 0,
        "settle_time_after": 0,
        "stop_on_fail": false
    }
}
```

B Utility: pswxdom2webapp

The **pswxdom2webapp** utility converts existing Browser Driver (pswxdom) configuration to JSON that can be used for web apps. The utility locates all Browser Driver access disclosure plugins, including ones created globally, as well as those overridden in the managed system policy (MSP) level. By default, a corresponding web app will be generated for each converted Browser Driver access disclosure. Browser Driver access disclosures used in the MSP-level will not be converted if none of the disclosure attributes values are overridden.

The **pswxdom2webapp** utility can be found in the util directory.

The following Browser Driver disclosure attribute settings are converted:

- checkboxdata
- constfielddata
- formatted username
- passwordfieldids
- settletime
- submitbuttonids
- url
- usernamefieldids

The following Browser Driver disclosure attribute settings are *not* converted:

- denypopups
- formatted title
- height
- uicontrols
- width
- simulatekeypress*

Notes:

- simulatekeypress will be supported, but not in Phase 1. A warning message will be given when pswxdom2webapp finds a Browser Driver disclosure that has this setting enabled. A web app will be created, but running it may cause issues if the website relies on keypresses to log in.
- If a semicolon is used as a submitbuttonids rather than an HTML tag, it will no longer work on web app. This will need to be replaced with an explicit HTML tag.

B.1 Usage

```
pswxdom2webapp.exe [-prefix] [-file] [-force]
```

Table 1: pswxdom2webapp arguments

Argument	Description
-prefix	Prefix the web app's name with the specified value.
-file	Convert Browser Driver disclosure configuration to JSON files. This is added to <*instance*>/webappfiles. This does not commit changes to the database. The JSON files generated can be used to manually create web apps.
-force	Override existing changes made to the database or converted web app files.

Converted global-level Browser Driver disclosure plug-ins will be named: `<disclosure plugin>`. If the -prefix argument is used, they will be named: `<prefix><disclosure-plugin>`. Description of the web app will be: `Webappjson generated for <disclosure plugin>`.

Converted MSP-level Browser Driver disclosure plugins will be named: `<MSP>-<disclosure plugin>`. If the -prefix argument is used, they will be named: `<prefix><MSP>-<disclosure-plugin>`. Description of the web app will be: `Webappjson generated for <MSP> / <disclosure plugin>`.

If a web app was already generated for a specific Browser Driver disclosure using pswxdom2webapp, it will not be generated again unless -force is used.

B.2 Examples

- To convert all browser driver disclosures to web app and adding it to the database, assuming none of the browser driver disclosures are converted previously:

  ```
  pswxdom2webapp.exe
  ```

- To convert all browser driver disclosures to web app and add them to the database, regardless if they have been already converted using pswxdom2webapp :

  ```
  pswxdom2webapp.exe -force
  ```

- To convert all browser driver disclosures to web app, but only create JSON configuration files instead of creating web app entries in the database:

  ```
  pswxdom2webapp.exe -file
  ```

- To convert all browser driver disclosures to web app, but only create JSON configuration files regardless if they have been already converted using pswxdom2webapp:

  ```
  pswxdom2webapp.exe -file -force
  ```