### 1.認証リクエストの選択

ツリーのサイトの一覧、もしくはインフォメーションの履歴の一覧の中から、ログインを行っているリクエストを右クリックし「Flag as Context」から「【コンテキスト名】: Form-based Auth Login Request」を実行します（図19①）。

### 2.認証情報の設定

「セッション・プロパティ」ウインドウのコンテキストの「認証」が開きます（図19②）。

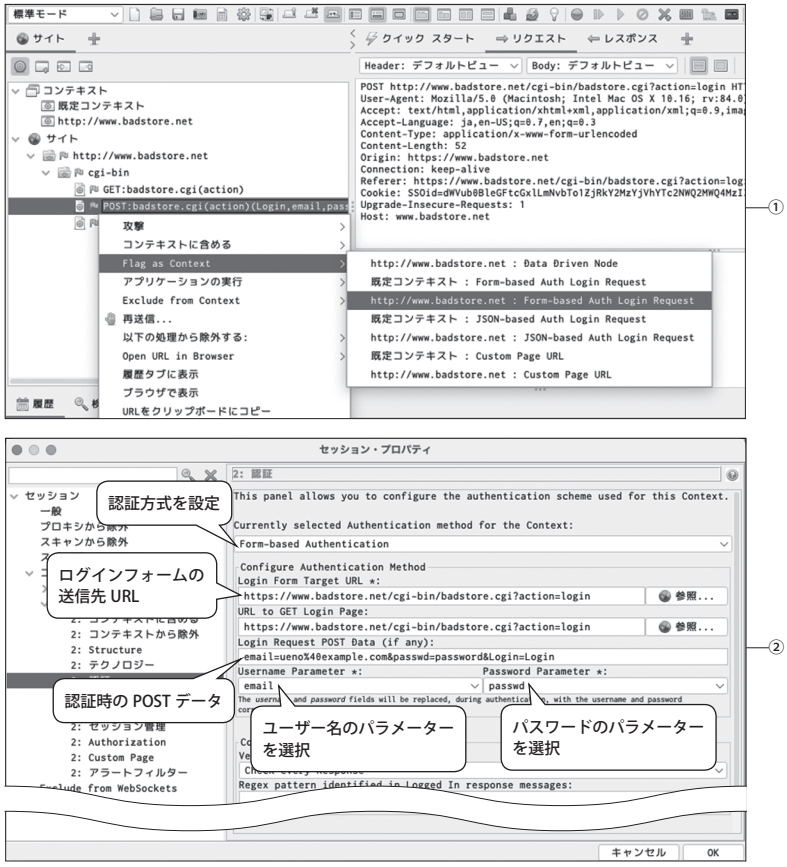


図 19：OWASP ZAP自動ログインの設定１

通常は「Currently selected Authentication method for the Context」に自動的に適切な認証の方式が選択されていますが、異なる認証方式を利用する場合には下記から選択します（ここでは「Form-based Authentication」として説明します）。

- Form-based Authentication（ユーザー名／パスワードによるフォームベース認証）

- HTTP/NTLM認証（BASIC認証、DIGEST認証、NTLM認証）
- マニュアル認証（自動ログインを設定しない）
- Script-based Authentication（あらかじめ作成したスクリプトによる認証）
- JSON-based Authentication（ユーザー名／パスワードを使用してログインURLにJSONオブジェクトを送信するタイプの認証）

通常は「Login Form Target URL」に自動的にログインフォームのURL、「Login Request POST Data」にログインリクエストを送信する際のPOSTデータが入っていますが、異なる場合には編集してください。

自動ログインで使用するユーザー名のパラメーターを「Username Parameter」から選択し、パスワードのパラメーターを「Password Parameter」から選択してください。

BadStoreの場合、ユーザー名／パスワードは「email」と「passwd」になります。「Login Request POST Data」は下記のようになります。

```
email=ueno%40example.com&passwd=password&Login=Login
```

### 3.ログイン成否の判定基準設定

ログイン成功を示すレスポンスの文字列の正規表現を設定します。ログイン中のときにだけ必ず出る文字列、もしくはログアウト中のときにだけ必ず出る文字列を調べて設定します（BadStoreには存在しません）（図20）。


図20：OWASP ZAP自動ログインの設定2

①レスポンスメッセージからログイン中のときにだけ必ず出る文字列を選択します。たとえば、ログイン状態のときに必ずログアウト用のリンクが表示されるのであれば、レスポンスメッセージ内の「logout.php」や「ログアウト」などの文字列を選択します。

②それを右クリックし「Flag as Context」から「【コンテキスト名】: Authentication Logged-in indicatort」を実行すると「セッション・プロパティ」の「認証」の「Regex pattern

identified in Logged In response messages」に下記のように追加されます。

```
\Qlogout.php\E
```

ログイン中のときにだけ必ず出る文字列が存在しない場合には、ログアウト中（ログインしていない状態）のときにだけ必ず出る文字列を調べて設定します。その場合には、右クリックメニューの「Flag as Context」から「【コンテキスト名】: Authentication Logged-out indicatort」を実行すると「セッション・プロパティ」の「認証」の「Regex pattern identified in Logged Out response messages」に追加されます。

OWASP ZAP では認証状態の確認をレスポンスメッセージ中の文字列以外での設定も可能です。「認証」の設定内の「Configure Authentication Verification」の「Verification Strategy」には下記の選択肢があります（図21）。

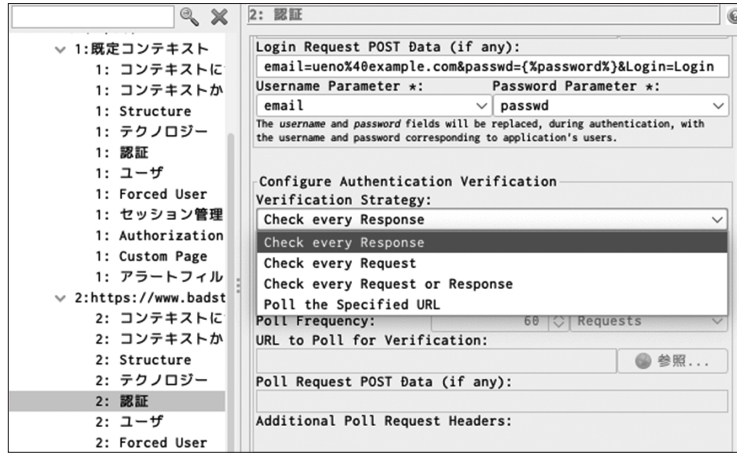


図21：OWASP ZAP自動ログインの設定３

- **Check every Response**
  - すべてのレスポンスメッセージの内容を確認
  - 完全なHTMLページを返す従来型のWebアプリケーションはこのパターンが良い
- **Check every Request**
  - すべてのリクエストメッセージの内容を確認
  - JWT（JSON Web Token）のようなセッションの状態をクライアント側で維持するようなWebアプリケーションはこのパターンが良い
- **Check every Request or Response**
  - すべてのリクエストメッセージとレスポンスメッセージの内容を確認

- **Poll the Specified URL**
  - ― 指定されたURLに対するレスポンスメッセージを指定された時間の間隔で定期的に確認
  - ― ユーザーがログインまたはログアウトしているかどうかを検出するために、確実に使用できるURLが少なくとも1つあるアプリケーションはこのパターンが良い

BadStoreの場合、ログイン中もしくはログアウト中を明示的に示す文字列が存在しないため、この項目に設定する文字列はありません。bsheader.cgi や My Accountページなどのログイン中のメッセージは認証状態を元にしておらず、Cookieの内容を表示しているため認証の成否を判定する基準にはなりません。この場合、認証の成否を基準に判定する脆弱性などの検証を正確に行うことはできないことがあります。

#### 4. 自動ログインユーザーの設定

自動ログインに使用するユーザー名とパスワードを「セッション・プロパティ」の「ユーザ」から設定します。

元のリクエストのPOSTデータに入っていたパラメーターはあらかじめ追加されていますので、使用する場合は「有効」にチェックを入れます。必要に応じてクレデンシャルを追加することもできます（図22）。

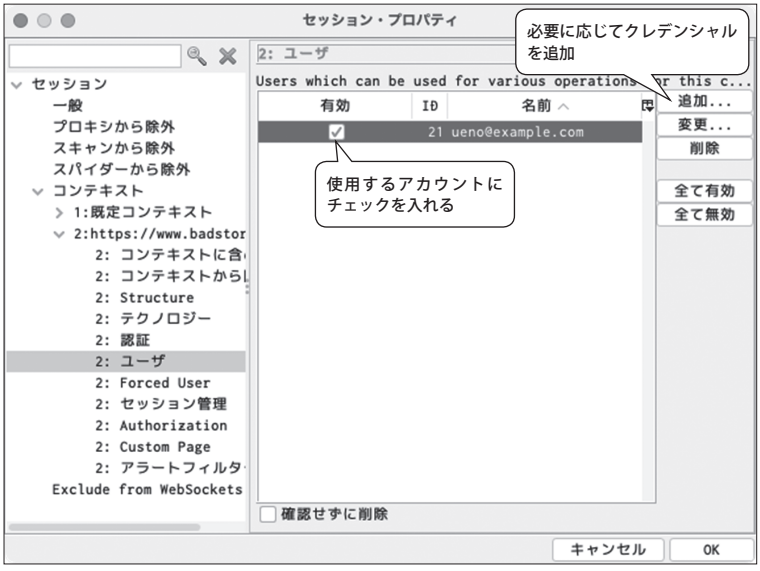


図22：OWASP ZAP自動ログインの設定4

## 手順４：自動クロールによる診断対象の記録

OWASP ZAPの自動クロール機能「スパイダー」を使うことで手動クロールでは見逃していたURLを自動で発見します。スパイダーは必要がなければ実行しなくても構いません。

スパイダーを実行する前には必ずコンテキストにアクセスが禁止されているURLや、実行しない方がよい機能が除外されていることを確認しておきましょう。

スパイダーを実行するにはツリーのサイトから自動クロールを開始するURLを選択して、右クリックメニューの「攻撃」→「スパイダー」を実行します（図23）。

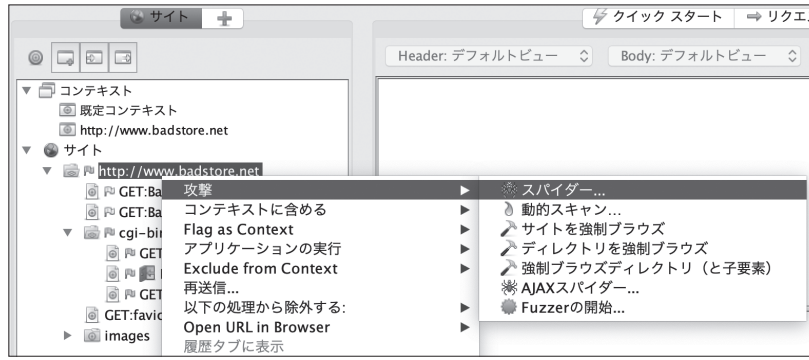


図23：OWASP ZAPスパイダーの実行

「スパイダー」ウインドウが開きますので、使用するコンテキストとユーザーを指定します（図24）。

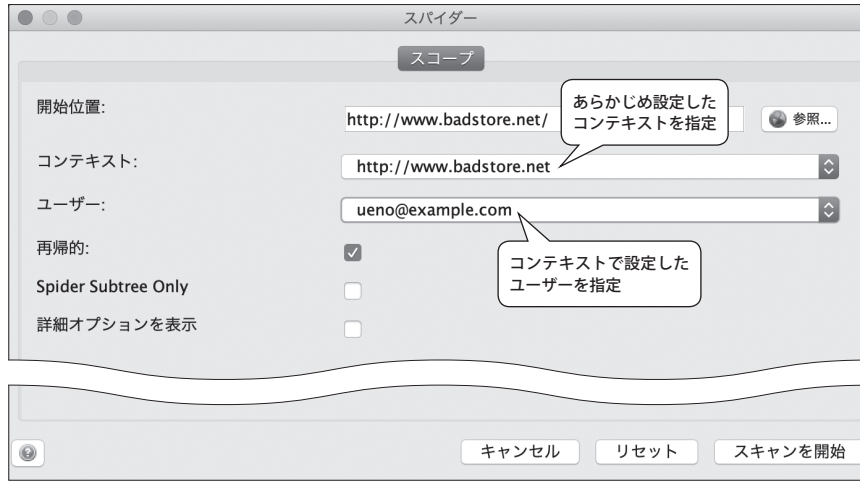


図24：OWASP ZAPスパイダー

「スキャンを開始」をクリックすると自動クロールが開始します。

インフォメーションに「スパイダー」タブが表示され、自動クロールが実行されていきます。このとき同時に静的スキャンも実行されます。進行状況が100%になれば自動クロールは終了です（図25）。

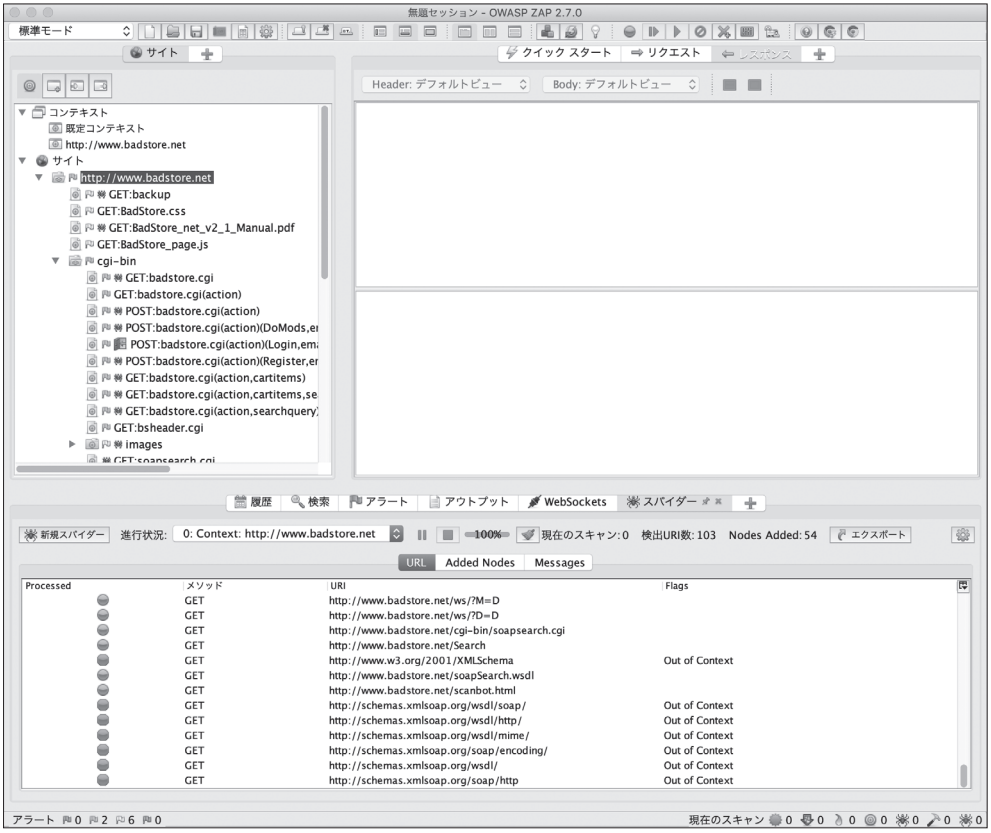


図25：OWASP ZAPスパイダーのスキャン完了

手動クロールやスパイダーでも発見できないURLを検出するには右クリックメニューの「攻撃」から「強制ブラウズ」機能を使うことで、ディレクトリ名を探すこともできます。この機能は辞書ファイルを指定することで、辞書ファイルに載っている名前のディレクトリ名を探していくというものです。

macOS版のOWASP ZAP（Ver. 2.7.0）では「強制ブラウズ」機能が標準では提供されていませんので、使用する場合には「アドオンの管理」→「マーケットプレイス」から「Forced

Browse」を追加してください。


図26：「Forced Browse」の追加

## コラム　異なるアクセス権限のアカウントがある場合

診断対象のWebアプリケーションで使用するアクセス権限が1つではない場合があります。たとえば、一般ユーザー権限と管理者権限が異なる場合や、他にアクセス権限が存在する場合（BadStoreの場合にはサプライヤーという権限が存在する）です。

OWASP ZAPの動的スキャンでは、基本的には1つのアクセス権限で1つのセッションとして管理する方が扱いやすいため、新規にセッションを作成して、別のセッションとして保存しておくことが望ましいでしょう。