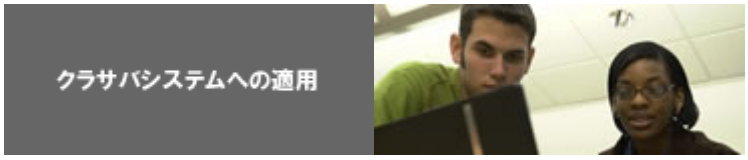
HP IceWall SSO

HP IceWall技術レポート:HP IceWall SSOを利用した Webシステム以外の認証特集(1)



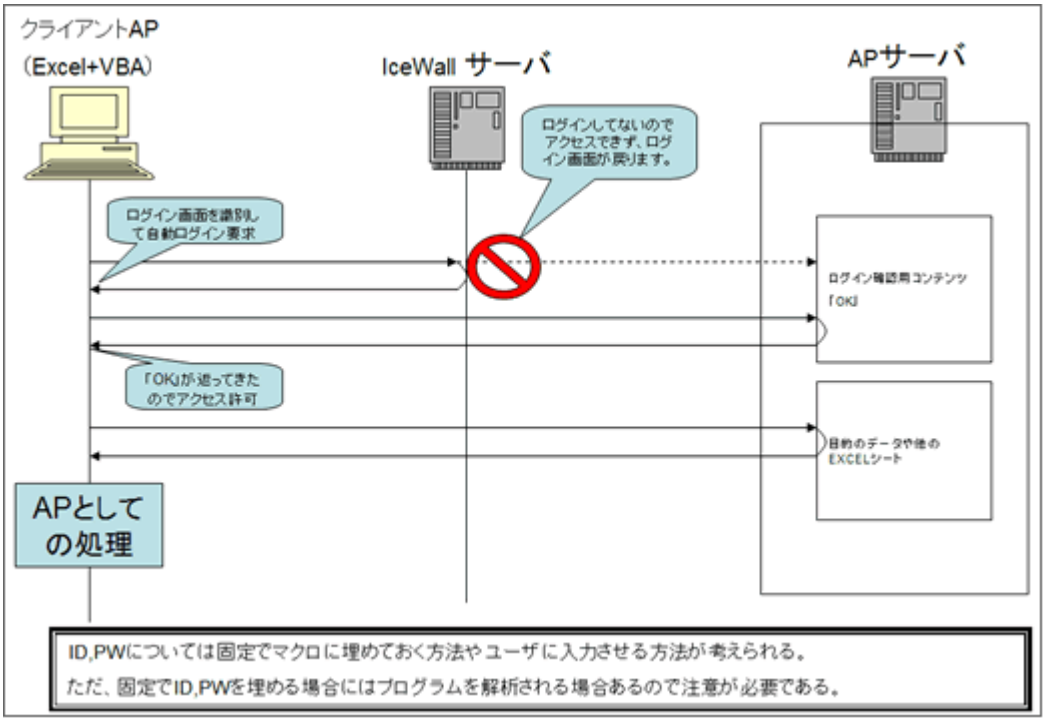
» EXCEL + VBAによるクライアントアプリケーション構築
» VCやVBで独自プロトコルのアプリケーション構築、認証認可はIceWall
»

HP IceWall SSOってWebの認証以外でも利用できるの？・・・もちろんYesです。
IceWall SSOと言えばWebの認証システムだけと思われているお客様も多いはずです。本来の目的としてはその通りなのですが、「せっかく導入したHP IceWall SSOをもっと有効に利用できませんか？」と言うお客様のご要望にお応えして、日本ヒューレット・パッカードと弊社とで様々な可能性も含めてお客様へ提案してきた結果、IceWallを利用してWeb以外のシステムの認証機能として活用して頂いているという客様も少しずつ増えています。

以下に、その事例としてHP IceWall SSOを利用したアプリケーションモデルを2つほど紹介させていただきます。

EXCEL + VBAによるクライアントアプリケーション構築

【例1】
EXCELをクライアントアプリケーションとして開発したいけど、データはHP IceWall SSOで守られた安全な場所に置いておきたい。



この例ではEXCELとVBAを組み合わせたアプリケーションにより、HP IceWall SSOで守られた場所にあるデータを使った処理を実現しています。

問題となる部分は、

・VBAを利用してどのようにHTTPリクエストを行うか？

というところと、

・実際にログイン処理はどうすればいいのか？

といったところでしょう。

HTTPリクエストについては「WebBrowserコントロール」を利用します。このコントロールを利用すれば、以下のように記述するだけでブラウザのようにコンテンツへアクセスすることができます。

```
Sheet1.WebBrowser1.Navigate2 "http://sever1/fw/dfw/BK1/fw/ok.sh", 4, "", "", ""
```

ログイン確認のためのコンテンツ(ok.sh)にアクセスしていますが、初めてアクセスした場合には実際に戻ってくるコンテンツはHP IceWall SSOのログイン画面です。ここでログインするための処理が必要になります。何らかのコンテンツを受信した場合、WebBrowserコントロールの「DocumentCompleteアクション」で通知されます。

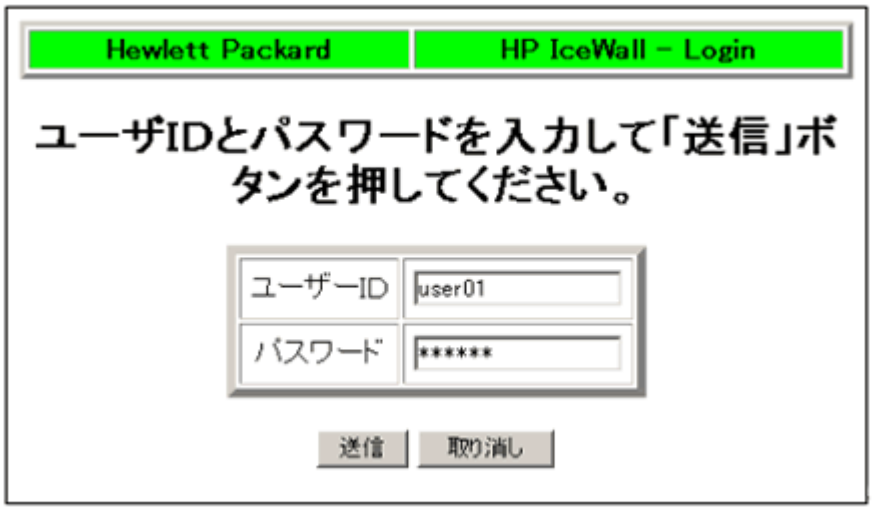
通知されたらドキュメントの内容を見てログイン画面であることを確認しログイン要求を行います。判断部分とログイン処理部分は下のような形で実現できます。

```
＜判断部分＞
If WebBrowser1.Document.Forms.Item(0).Item(0).Name <> "ACCOUNTUID" Then
login=0      'ログイン画面ではありません
Else
login=1      'ログイン画面です
End if
```

この場合は、単純にログイン画面に含まれる項目名「ACCOUNTUID」があるかどうかによってログイン画面を判断しています。

```
＜ログイン処理部分＞
WebBrowser1.Document.Forms.Item(0).Item(0).Value = "user01"
WebBrowser1.Document.Forms.Item(0).Item(1).Value = "user01"
WebBrowser1.Document.Forms.Item(0).Item(4).Click
```

この場合は固定のID(user01)、固定のPW(user01)をIDフィールド、PWフィールドに設定して送信ボタンをクリックしています。この処理でHP IceWall SSOにログインして確認のためのコンテンツ(ok.sh)が取得でき、IceWalにログインした状態となります。(WebBrowserコントロールを見えるようにしていると下のようになっています)



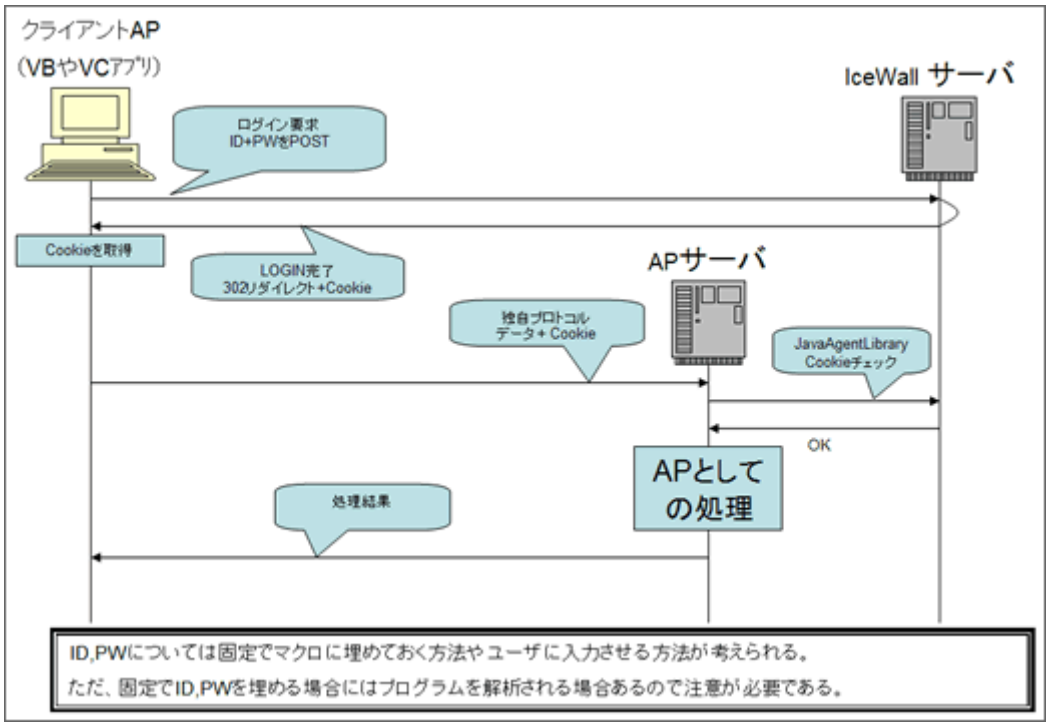
この後、目的のデータやEXCELシートにアクセスするのですが、セッション情報(Cookie)はWebBrowserコントロールが保持しており、このEXCELシートからHTTPリクエストで要求を行う場合には自動的に送ってくれます。それによりHP ceWall SSOの認可を通過できるわけです。この状態になれば、データへのアクセスは、EXCELからWebサーバ上のファイルへ普通にアクセスする形で行えば良いだけです。以下に例を記述しておきます。

```
Workbooks.Open Filename:=" http://sever1/fw/dfw/BK1/data/data.xls"
```

この後は持ってきたデータをどのように利用するかはアプリケーションの処理次第ということになります。新しいアイデアにご利用ください。

VCやVBで独自プロトコルのアプリケーション構築、認証認可はIceWall

【例2】
VCやVBでアプリケーションを開発し独自プロトコルでデータの受け渡しはしたい。でも、認証、認可はHP IceWall SSOに任せたい。



こちらの例ではVCやVBで作ったアプリケーションが、独自のプロトコルで通信することを想定しています。こんな場合でも方法によってはHP IceWall SSOを使った認証、認可システムを構築することができます。
まずクライアントAPがIceWall SSOに対しログイン要求を行い、認証された証であるセッション情報(Cookie)を取得します。ログイン要求やCookieの取得は以下のように行います。

＜ログイン要求＞
POST /fw/dfw HTTP/1.1
Accept-Language: ja
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)
Host: 192.168.50.250
Content-Length: 84
Cache-Control: no-cache
ACCOUNTUID=user01&PASSWORD=user01&HIDEURL=%2Fsys%2Ffw%2Fprintenv&LOGIN=ICEWALL_LOGIN
＜戻りデータ＞
HTTP/1.1 302
Date: Tue, 24 Jun 2003 12:05:10 GMT
Server: Apache/2.0.45 (Unix)
Set-Cookie: IW_INFO=414d5e4754ade7ec3469932287a949c5;
Location: http://192.168.50.250/fw/dfw/sys/fw/printenv
Content-Length: 0
Keep-Alive: timeout=15, max=98
Connection: Keep-Alive
Content-Type: text/html; charset=shift_jis

上の「IW_INFO=414d5e4754ade7ec3469932287a949c5;」内の「414d5e4754ade7ec3469932287a949c5」がセッション情報です。

クライアントAPは、取得したセッション情報をその後のサーバ間通信で付加して送り続けます。サーバ側は受け取ったセッション情報をIceWallサーバに問い合わせ、そのセッション情報が有効であるかをチェックします。そのとき、セッションをチェックするためにJava Agent Libraryを利用します。Java Agent Libraryは指定されたCookie情報を認証サーバが保持しているかを問い合わせるためのライブラリで、利用方法は以下のようになります。(JAVA Agent Library 開発者マニュアル参照)

```
import com.hp.icewall.certrequest.*;

    :

AccessRequest access = new AccessRequest();

access.setSession(session);                    // session は送られてきたデータ
access.setUrl("http://www.xxx.co.jp/");

RequestResult ret = access.doRequest("TESTSERVER:14142", 10, 5, 3);

if ( ret == RequestResult.OK ) {
      String userid = access.getUserid();
      System.out.println("userid=" + userid);
      ArrayList list = access.getDatalist();
      Iterator iterator = list.iterator();
      while (iterator.hasNext()) {
            String userdata = (String)iterator.next();
            System.out.println("userdata=" + userdata);
   }
} else {
      System.out.println("Access NG!!(ret=" + ret + ")");
}
```

認証サーバがセッション情報を持っているのであれば、認証されているということでAPサーバ上のアプリケーションは処理を続けることができ、その結果をクライアントへ返すことで一連の処理が完結します。
このようにHP IceWall SSOを利用することによって、新たな認証の仕組みを作ることなく、一貫した認証基盤と言われる物を構築することができます。

今回の例はほんの一部の例であり、お客様のアイデア次第では他にも様々な可能性があるはずです。もしもお客様が「こんな可能性がありそうかな？」と思われたらご相談ください。きっと良いアイデアを提案できるはずです。

2004.07.13 （株）SCC I&C事業部インターネットセキュリティ部マネージャ 祐徳 弘己 氏

●関連技術レポート

≫ HP IceWall SSOを利用した Webシステム以外の認証特集(1) - クラサバシステムへの適用（本トピックス）
≫ HP IceWall SSOを利用した Webシステム以外の認証特集(2) - MetaFrameを使用したWindowsターミナルサービスへの適用
≫ HP IceWall SSOを利用した Webシステム以外の認証特集(3) - Web化されていないNotesとIceWallの連携