# Table of Contents

**1 F5 APM Integration** ..... **1**

**2 F5 Big-IP Access Policy Manager (APM) Integration Notes** ..... **2**

**3 RADIUS Integration** ..... **3**
3.1 Test the RADIUS authentication ..... 5

**4 Logon page Customisation** ..... **6**
4.1 Removing the Automatic TURing image ..... 7

**5 Testing** ..... **9**

**6 F5 Firepass Integration** ..... **10**
6.1 Introduction ..... 10
6.2 Prerequisites ..... 10
6.3 Architecture ..... 10
6.4 Installation ..... 10
6.5 F5 Networks FirePass VPN Configuration ..... 12
6.6 Test the RADIUS authentication ..... 13
6.7 Modifying the FirePass login page for PINsafe TURing image ..... 13
6.8 Verifying Installation ..... 14
6.9 Troubleshooting ..... 14
6.10 Additional Information ..... 14

**7 F5 SAM Integration** ..... **15**

**8 F5 Secure Access Manager (SAM) Integration Notes** ..... **16**

**9 RADIUS Integration** ..... **17**
9.1 Test the RADIUS authentication ..... 19

**10 Log-in page Customisation** ..... **20**

**11 Testing** ..... **21**

# 1 F5 APM Integration

# 2 F5 Big-IP Access Policy Manager (APM) Integration Notes

This article describes how to integrate the F5 Big-IP Access Policy Manager with Swivel. The article covers two aspects:

- the integration of the two servers so that the F5 uses Swivel as its RADIUS server
- the modification of the F5 login page to include the TURing image or other Swivel elements as required.

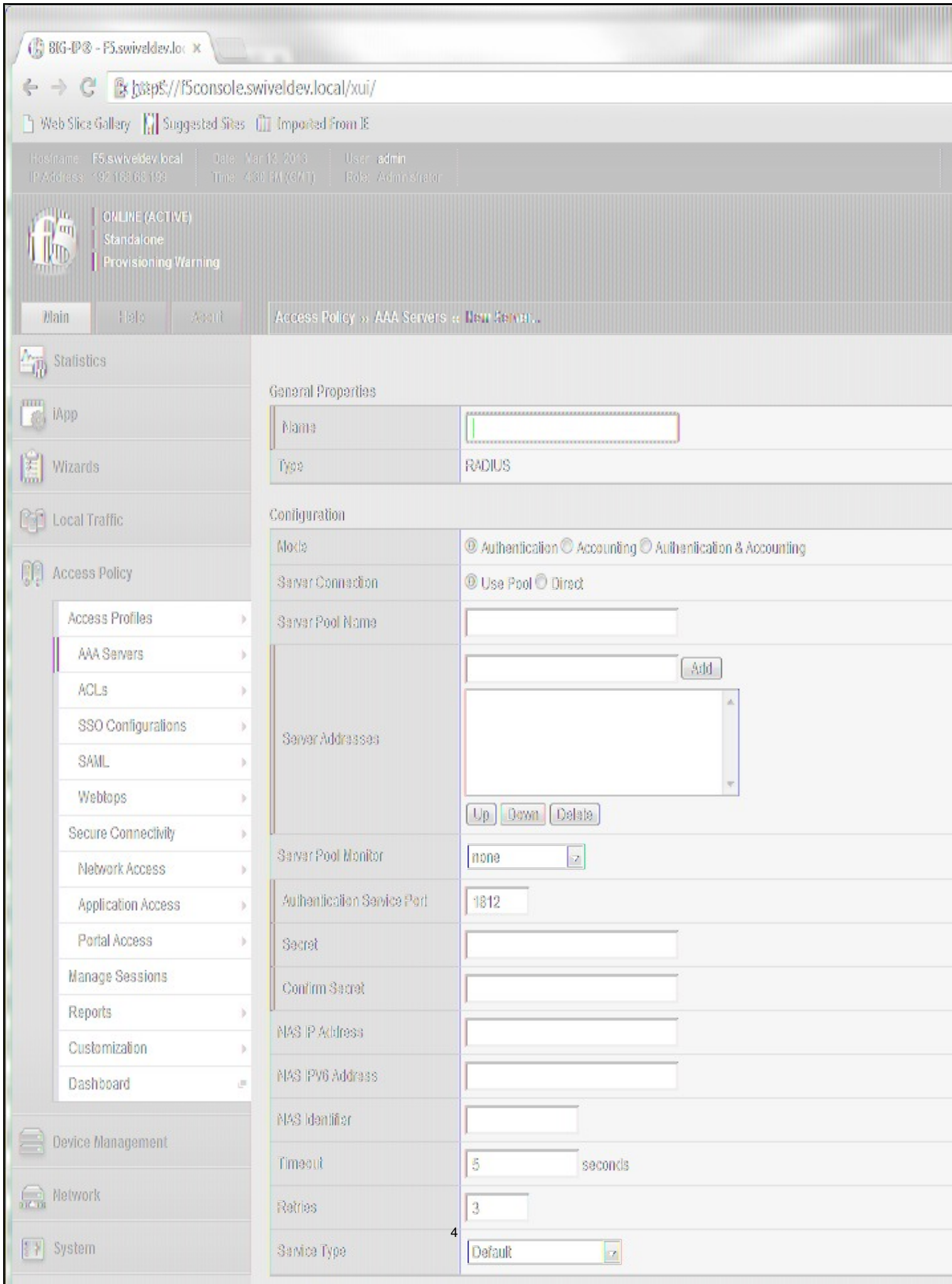
# 3 RADIUS Integration

To use Swivel with F5 Big-IP you need to enable the Radius Server on Swivel. (On the RADIUS->Server page)

A NAS Entry then need to be created that includes the F5 server IP address/hostname and a shared secret.

The associated configuration then needs to be created on the F5 server.

This is done on the Access-Policy->AAA-Servers screen.


BIG-IP® - F5.swiveldev.lo
https://f5console.swiveldev.local/xui/
Web Slice Gallery
Suggested Sites
Imported From IE
Hostname: F5.swiveldev.local
IP Address: 192.168.66.199
Date: Mar 13, 2013
Time: 4:30 PM (GMT)
User: admin
Role: Administrator
ONLINE (ACTIVE)
Standalone
Provisioning Warning
Main
Help
Acout
Access Policy » AAA Servers » New Server...
Statistics
iApp
Wizards
Local Traffic
Access Policy
Access Profiles
AAA Servers
ACLs
SSO Configurations
SAML
Webtops
Secure Connectivity
Network Access
Application Access
Portal Access
Manage Sessions
Reports
Customization
Dashboard
Device Management
Network
System
General Properties
Name
Type
RADIUS
Configuration
Mode
Authentication
Accounting
Authentication & Accounting
Server Connection
Use Pool
Direct
Server Pool Name
Server Addresses
Add
Up
Down
Delete
Server Pool Monitor
none
Authentication Service Port
1812
Secret
Confirm Secret
NAS IP Address
NAS IPV6 Address
NAS Identifier
Timeout
5
seconds
Retries
3
Service Type
Default

Once this entry has been created it can be used when defining Access Profiles.

It is important to remember the name of the profile created as this will be required for the customisation.

## 3.1 Test the RADIUS authentication

At this stage it should be possible to authenticate by SMS, hardware Token, Mobile Phone Client and Taskbar to verify that the RADIUS authentication is working for users. Browse to the SSL VPN login page, and enter Username and if being used, the password. From the Swivel Administration console select User Administration and the required user then View Strings, and select an appropriate authentication string or OTC for the user. At the SSL VPN login enter the required OTC. Check the Swivel logs for a RADIUS success or rejected message. If no RADIUS message is seen, check that the Swivel RADIUS server is started and that the correct ports are being used.

# 4 Logon page Customisation

Once you have configured your access policy, you need to modify the logon page. You can edit it from the management console as follows:

From the Main menu, select Access Policy, then Customization. From the View dropdown, select Advanced Customization. From the folder tree, select Customization Settings -> Access Profiles -> [Your access profile] -> Access Policy -> Logon Pages -> Logon Page -> logon.inc.

Search for the line

```
function OnLoad()
```

Insert the following immediately before it:

For TURing image:

```
// **** PINsafe Customisation Start ****

// Change this to match the PINsafe image URL.
var imageUrl = "https://<your_swivel_server>:8443/proxy/SCImage?username=";

function ShowTuring() {
  var usernameField = document.getElementById("input_1");
  if (usernameField && usernameField.value && usernameField.value != "") {
    var img = document.getElementById("turing_img");
    if (img) {
      img.style.display = "";
      img.src = imageUrl + usernameField.value + "&random=" + Math.floor(Math.random()*10000);
    }
  }
}
// **** PINsafe Customisation End ****
```

Or for PinPad:

```
// **** PINsafe Customisation Start ****

// Change this to match the PINsafe image URL.
var imageUrl = "https://<your_swivel_server>:8443/proxy/SCPinPad?username=";

function ShowPinPad() {
  var usernameField = document.getElementById("input_1");
  if (usernameField && usernameField.value && usernameField.value != "") {
    var padno = Math.floor(Math.random() * 100000);
    for (var i=0; i<10; i++) {
      var img = document.getElementById("pinpad" + i);
      if (img) {
        var url = imageUrl + usernameField.value + "&padno=" + padno + ":" + i;
        img.src = url;
      }
    }
  }
}

function InsertPinPad() {
  var footerCell = document.getElementById("credentials_table_footer");
  if (footerCell) {
    var footerRow = footerCell.parentNode;
    var formTable = footerRow.parentNode;
    var pinpadRow = document.createElement("tr");
    pinpadRow.setAttribute("id", "turing_row");
    var pinpadCell = document.createElement("td");
    pinpadCell.setAttribute("colspan", "2");
    pinpadCell.setAttribute("align", "center");
    var pinpadTable = document.createElement("table");
    pinpadTable.style.height = "225px";
    pinpadTable.style.width = "150px";
    var row, cell, img;
    for (var r=1; r<=9; r+=3) {
      row = document.createElement("tr");
      for (var c=r; c<r+3; c++) {
        cell = document.createElement("td");
        cell.setAttribute("align", "center");
        img = document.createElement("img");
        img.src = "images/blank.png";
        img.setAttribute("id", "pinpad" + c);
        img.setAttribute("onclick", "AddDigit(" + c + ")");
        cell.appendChild(img);
        row.appendChild(cell);
      }
      pinpadTable.appendChild(row);
    }
    row = document.createElement("tr");
    cell = document.createElement("td");
    cell.setAttribute("align", "center");
    img = document.createElement("img");
    img.src = "images/refresh.png";
    img.setAttribute("onclick", "ShowPinPad()");
    cell.appendChild(img);
    row.appendChild(cell);
    cell = document.createElement("td");
    cell.setAttribute("align", "center");
    img = document.createElement("img");
    img.src = "images/blank.png";
    img.setAttribute("id", "pinpad0");
    img.setAttribute("onclick", "AddDigit(0)");
    cell.appendChild(img);
    row.appendChild(cell);
    cell = document.createElement("td");
    cell.setAttribute("align", "center");
    img = document.createElement("img");
    img.src = "images/clear.png";
    img.setAttribute("onclick", "ClearOtc()");
    cell.appendChild(img);
```

```
    row.appendChild(cell);
    pinpadTable.appendChild(row);
    pinpadCell.appendChild(pinpadTable);
    pinpadRow.appendChild(pinpadCell);
    formTable.insertBefore(pinpadRow, footerRow);
  }
}

// Check that the following field is correct. If PINsafe is the ONLY form of authentication,
// or is the first authentication, it will be "input_2".
// If it is the second authentication, it will be "input_3".
var otcFieldId = "input_2";

function AddDigit(digit) {
  var otcField = document.getElementById(otcFieldId);
  if (otcField) {
    otcField.value += digit;
  }
}

function ClearOtc() {
  var otcField = document.getElementById(otcFieldId);
  if (otcField) {
    otcField.value = "";
  }
}

// **** PINsafe Customisation End ****
```

A few lines below this are the following lines:

```
if( form == null ){
  return;
}
```

Below this, insert the following for TURing:

```
// **** PINsafe Customisation Start ****
var footerCell = document.getElementById("credentials_table_footer");
if (footerCell) {
  var footerRow = footerCell.parentNode;
  var formTable = footerRow.parentNode;
  var turingRow = document.createElement("tr");
  turingRow.setAttribute("id", "turing_row");
  var turingCell = document.createElement("td");
  turingCell.setAttribute("colspan", "2");
  turingCell.setAttribute("align", "center");
  var turingImg = document.createElement("img");
  turingImg.setAttribute("id", "turing_img");
  turingImg.style.display = "none";
  turingCell.appendChild(turingImg);
  var turingBrk = document.createElement("br");
  turingCell.appendChild(turingBrk);
  var turingBtn = document.createElement("input");
  turingBtn.setAttribute("type", "button");
  turingBtn.setAttribute("value", "New Image");
  turingBtn.onclick = ShowTuring;
  turingCell.appendChild(turingBtn);
  turingRow.appendChild(turingCell);
  formTable.insertBefore(turingRow, footerRow);
}
// Optional: to automatically show the TURing after entering the username, include the following lines.
var usernameField = document.getElementById("input_1");
if (usernameField) {
  usernameField.onblur = ShowTuring;
}
// Optional: if the username is pre-populated, use the following line to display the TURing image immediately
ShowTuring();
// **** PINsafe Customisation End ****
```

or this for Pinpad:

```
  // **** PINsafe Customisation Start ****
  InsertPinPad();
  // The next section is optional - use this if you want to show the TURing automatically when the username changes.
  var usernameField = document.getElementById("input_1");
  if (usernameField) {
    usernameField.onblur = ShowPinPad;
  }
  // **** PINsafe Customisation End ****
```

## 4.1 Removing the Automatic TURing image

Remove or comment out the following lines with // at the front

```
// Optional: to automatically show the TURing after entering the username, include the following lines.
var usernameField = document.getElementById("input_1");
if (usernameField) {
  usernameField.onblur = ShowTuring;
}
```

For Pinpad, the penultimate line above will be

```
  usernameField.onblur = ShowPinPad;
```

The final step here is to set the image URL. There are a number of options:

- The simplest option is to use the Swivel Server directly. However, this requires that the Swivel Server is directly accessible from the internet, which is not a recommended solution, as it is a security risk. Also, you will need a commercial SSL certificate on the Swivel server to avoid problems with certificate errors. In this case, simply replace *<your_swivel_server>* above with the external URL of your Swivel Server.
- The second option is to create a virtual server on the F5 Big-IP to act as an anonymous proxy to the Swivel Server. This is suitable if the F5 is your only Swivel integration, as it requires that the F5 is set as the default gateway for your Swivel appliance. Details for this are not provided, as it should be clear from the F5 documentation how to do this. You might also want to create an iRule to restrict access only to the TURing image, as suggested below. In this case, you should replace *<your_swivel_server>* with the external URL of your F5. If you have set up the virtual server with a different service port, you might need to change this as well.

```
when HTTP_REQUEST {
  if { [HTTP::uri] starts_with "/pinsafe/SCImage?" } {
    pool PINsafe_8080
  } else { HTTP::respond 403 }
}
```

- The third option is suitable if you have other Swivel integrations. In this case, you can use the URL of the TURing image on the other integration to deliver the TURing image. For example, if you have an integration with Outlook Web Access, use the following:

```
var imageUrl = "https://<your_swivel_server>/owa/auth/SCImage.aspx?username=";
```

Here, replace *<your_swivel_server>* with the URL of your OWA server.

Another example: if you have a UAG integration, use the following:

```
var imageUrl = "https://<your_swivel_server>/InternalSite/images/customupdate/images.asp?username=";
```

NOTE: If you are using Pinpad, substitute **SCPinPad** for **SCImage** above.

# 5 Testing

Now when the F5 server is accessed via the **pinsafe** access policy the user should see a modified login page with the option to request a TURing image.

The user should enter their username and see a TURing image when the click the TURing button. At this point a Session Start message for the user should show in the PINsafe logs.

If no image shows, check that the URL is correct and ensure that there is no firewalls blocking the request.

Also check that Session Create by Username is enabled on the Swivel server.

The user should then enter their one-time code. The login.

If the log-in fails, check the Swivel log files to see if a RADIUS request was logged. If not then check the settings for the RADIUS on F5 and Swivel to ensure IP Addresses, port numbers and shared secrets all match. Also check that no firewalls are blocking the RADIUS requests.

If RADIUS attempts are being logged but authentication is failing, check that the session was started correctly and that there is no password associated with the account etc.

# 6 F5 Firepass Integration

## 6.1 Introduction

This document outlines the steps required to integrate the F5 Networks FirePass SSL VPN with the Swivel PINsafe authentication server.

FirePass VPN appliances are able to use external RADIUS servers for providing authentication. The PINsafe server provides RADIUS authentication, thus the FirePass VPN can be configured to use the PINsafe server for authentication via RADIUS.

PINsafe users can use either PINsafe?s Single Channel ( TURing, PATtern) or Dual Channel (SMS, Swivlet applet) methods to retrieve Security Strings, which are applied against the user?s PIN to extract a One-Time Code (OTC) which represents the password for an authentication request.

With Dual Channel methods, the user already holds one or more Security Strings on their mobile device (and can request more at any time) so with the FirePass VPN configured to use the matching PINsafe server for RADIUS authentication, no further integration is required.

However with Single Channel methods, the user must be presented with a TURing or PATTern image upon login (representing a single time-limited Security String), so they can extract their OTC. The Authentication configuration section below describes how to achieve the RADIUS configuration. Single Channel requires access to the PINsafe server by a Public IP address.

## 6.2 Prerequisites

### 6.2.1 Baseline

The FirePass VPN appliance tested was FirePass 600. (http://www.f5.com/products/FirePass/FP600.html)

The PINsafe server used was PINsafe v3.1. However, no changes have been made to PINsafe since then which would render the integration invalid.

The primary web browser used for testing was Internet Explorer 6.0.2900.2180.xpsp_sp2_gdr.050301-1519.

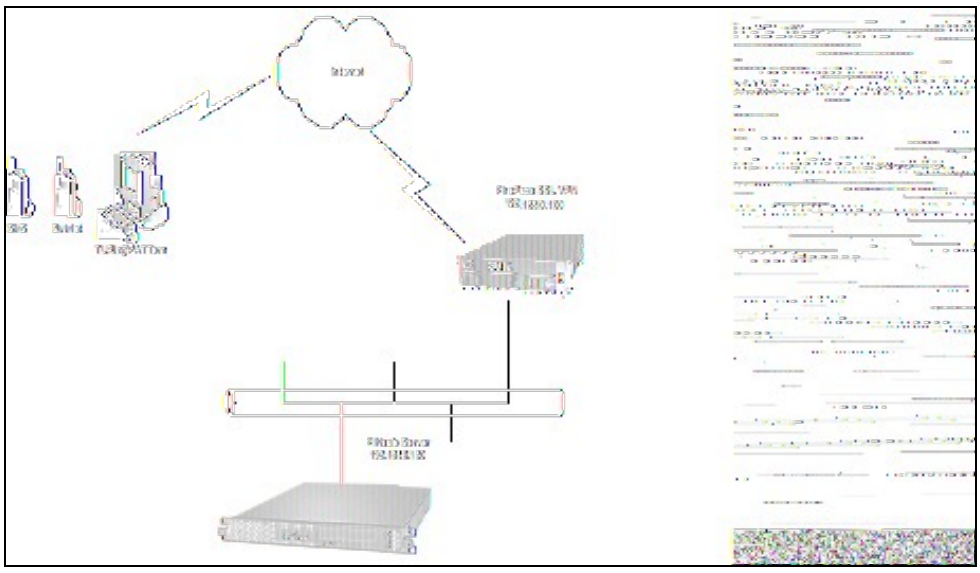
## 6.3 Architecture

The user connects to the FirePass VPN using a web browser, pointing to the appropriate login URL for the VPN in question.

The FirePass VPN is configured to use a PINsafe server for RADIUS authentication.

Users are stored and maintained in the PINsafe server.

*Figure 1. The following diagram shows the configuration used and is typical. This example is used throughout this document*:



## 6.4 Installation

### 6.4.1 PINsafe Configuration

Configuration of the PINsafe server for RADIUS authentication with the FirePass VPN consists of three steps:

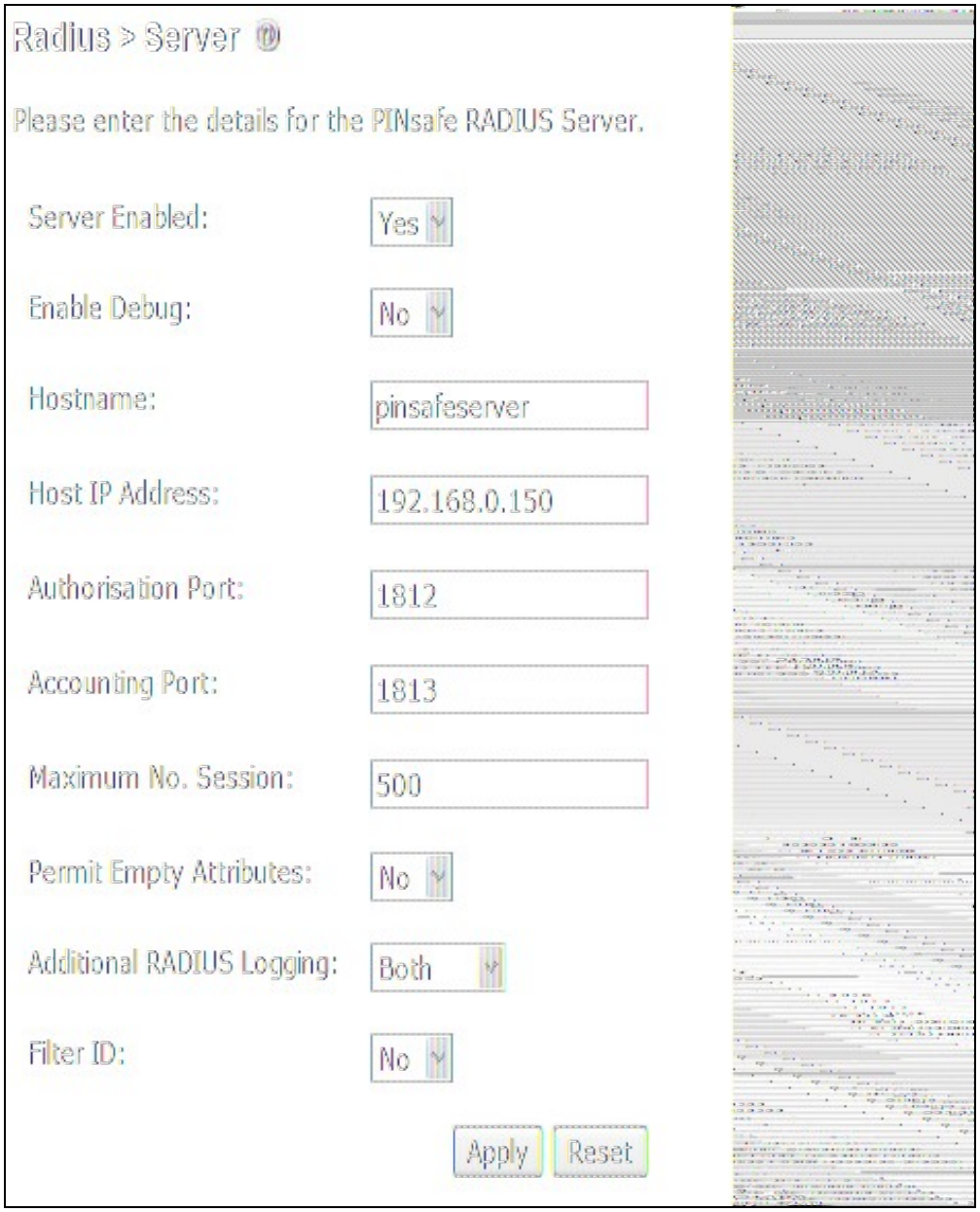
1. Configure PINsafe RADIUS settings.
2. Set up the NAS (Network Access Server), which in this case is the FirePass VPN.
3. Configure the PINsafe server to allow TURing/PATTern session creation with a username.

NOTE ? This document assumes that the PINsafe server has been configured to use a specific user repository and populated with users. Please refer to the PINsafe Administration Guide for detailed instructions.

**1. Configuring PINsafe RADIUS settings**

Configure the RADIUS settings using the RADIUS configuration page in the PINsafe Administration console. In our example (see diagram above) the RADIUS Mode is set to ?RADIUS Server? and the HOST IP (the PINsafe server) is set to 192.168.0.150.

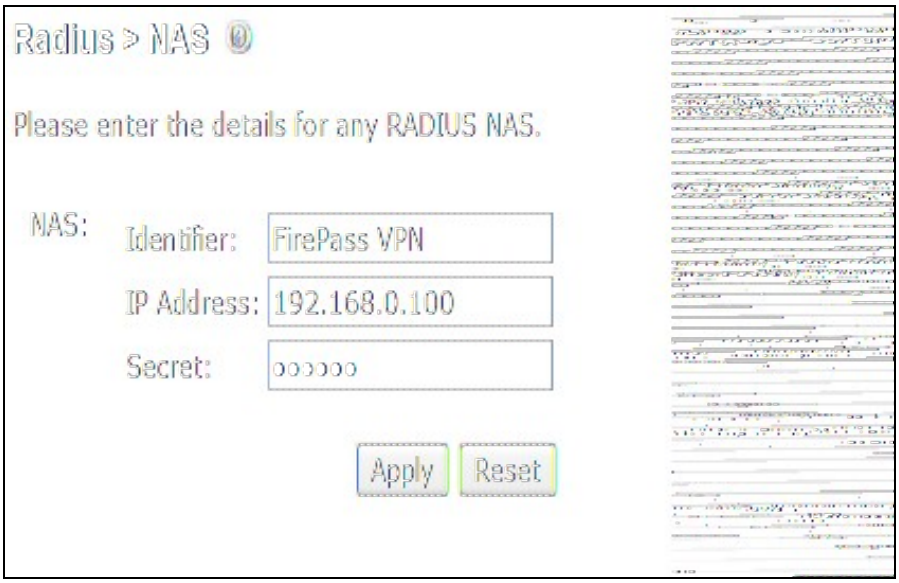
*Figure 2. PINsafe RADIUS configuration page.*




**2. Setting up the NAS**

Set up the NAS using the Network Access Servers page in the PINsafe Administration console. In our example (see Figure 3), the meaningful name ?FirePass VPN? has been assigned so it can be identified if you have more that one NAS configured. The IP address has been set to the IP of the VPN appliance, and the NAS secret assigned that will be used on both the PINsafe server and VPN RADIUS configuration.

 *Figure 3. Extract from PINsafe NAS setup page*




**3. Configure the PINsafe server to allow TURing/PATTern session creation with a username.**

The PINsafe server must be configured to allow a Single Channel session to be created by accessing a specific URL on the PINsafe server. The following URL would create a start a session and return the image for the user ?test?:

For a Swivel hardware or virtual appliance http://Swivel_IP:8443/proxy/SCImage?username=test

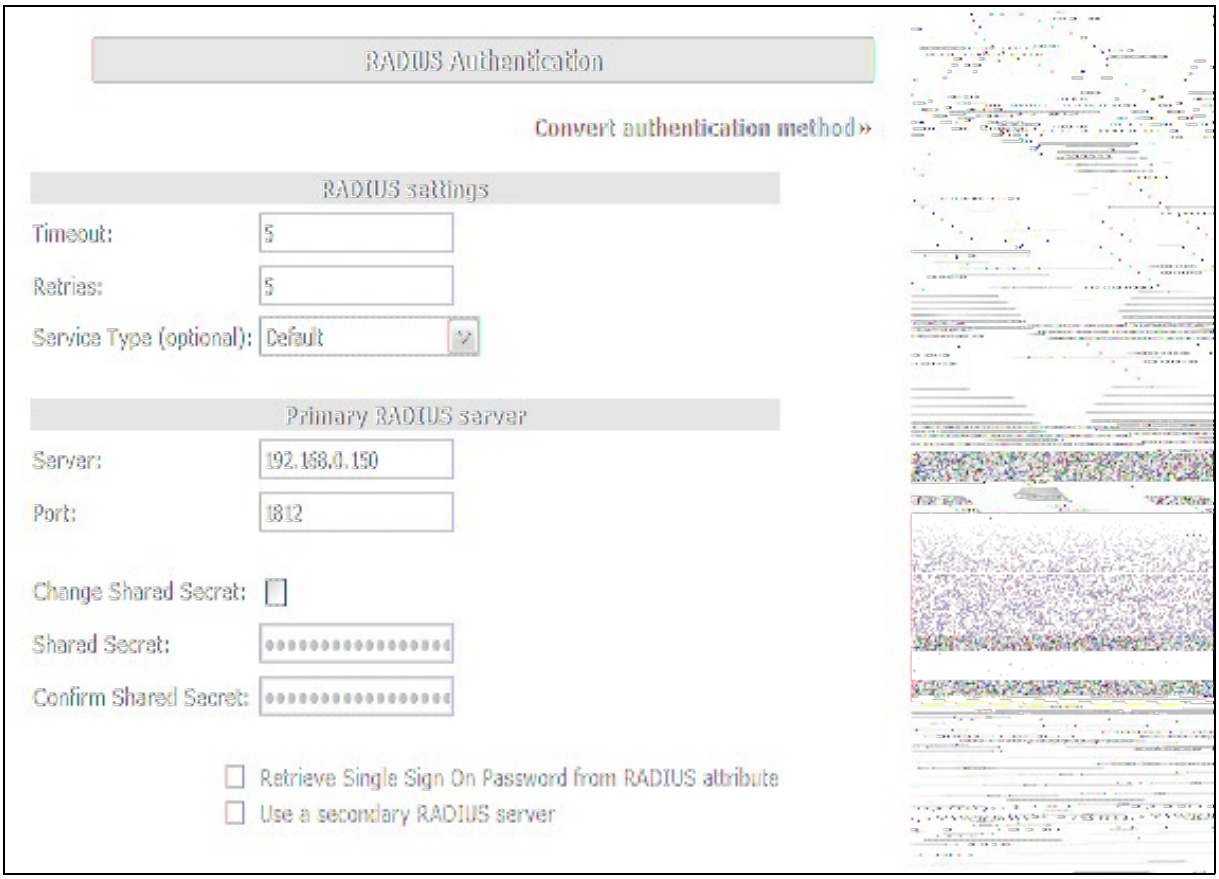
For a software only install see Software Only Installation

## 6.5 F5 Networks FirePass VPN Configuration

The RADIUS FirePass configuration is found under Users, Groups, Master Groups, Radius_Users, and then the Authentication tab. The Primary RADIUS server was set to the IP address of the PINsafe server followed by the authorization port (see Figure 5). The shared secret entered was the same secret entered in the PINsafe NAS entry (see Figure 3).

If you want to configure a secondary PINsafe RADIUS server for failover you would add the details of the server in the ?Secondary RADIUS server? section on this page. If you are utilizing the High Availability PINsafe solution, failover/redundancy is managed by that solution, thus you would only enter the Primary RADIUS server address.

*Figure 4. Extract from FirePass RADIUS Authentication setup page*




## 6.6 Test the RADIUS authentication

At this stage it should be possible to authenticate by SMS, hardware Token, Mobile Phone Client and Taskbar to verify that the RADIUS authentication is working for users. Browse to the SSL VPN login page, and enter Username and if being used, the password. From the Swivel Administration console select User Administration and the required user then View Strings, and select an appropriate authentication string or OTC for the user. At the SSL VPN login enter the required OTC. Check the Swivel logs for a RADIUS success or rejected message. If no RADIUS message is seen, check that the Swivel RADIUS server is started and that the correct ports are being used.

## 6.7 Modifying the FirePass login page for PINsafe TURing image

The PINsafe sends Security Strings to users via SMS, Swivlet applet (Dual Channel) or through a TURing image (Single Channel) accessed by public IP address from the PINsafe server. The user extracts their One Time Code (OTC) from the Security String and enters it into the VPN sign-in. If the user has been assigned a PINsafe server static password, they must enter the password plus their OTC. For example, if the user?s PINsafe static password was ?foobar? and their OTC were 7452, they would enter ?foobar7452? at the login prompt.

If the PINsafe user were configured to use Dual Channel (SMS or applet), they should have a security string ready on their mobile device. No modification to the FirePass login page would be required. For Single Channel users, we need some way of presenting a TURing image on the FirePass VPN?s login page. This can be achieved through configuration of the FirePass login screen via WebDAV.

**To enable WebDAV based customization**

1. Create an HTTP web service on the Device Management : Configuration : Network Configuration : Web Services screen.
2. Select the **Allow insecure access** option on the Device Management : Security : User Access Security screen.
3. Check **Allow WebDAV sandbox customization** on the Device Management : Customization screen and enter a WebDAV password in the text box that appears.

The WebDAV sandbox is accessed via HTTP at the URI /**sandbox** as the user **webdav**. So, for example, if the FirePass controller has been configured using the steps above with a HTTP web service at 192.168.0.99, you would use the URL **http://192.168.0.99/sandbox/**.

Any content can be placed in the sandbox directory. The FirePass controller uses specific files to override or supplement stock system behavior. To add the TURing image to the right of the logon prompt, the **right.inc** file was created and added to the sandbox, with the following content:

```
<script language="JavaScript">


</script>

<input name="btnTuring" type="button" value="OTC Image" class="submitbutton" onclick="ShowTuring()" />

<img id="imgTuring" style="visibility:hidden;" alt="Turing image" />
```

Edit the following line with the correct IP address

**sUrl="http://192.168.0.150:8443/proxy/SCImage?UserName=";**

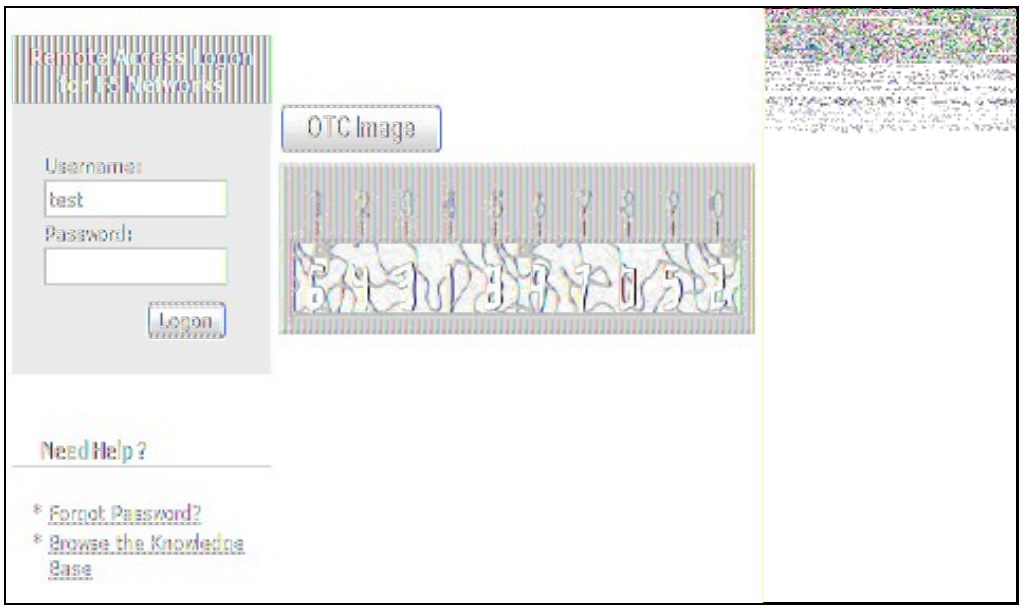
For PINsafe 3.1.3a and later the following line needs to be edited:

**sUrl="http://192.168.0.150:8443/proxy/SCImage?username=";**

To upload the WebDAV pages browse to the sandbox with a web browser using http (not https) and enter the WebDAV username and password.

Once loaded into the sandbox, the login page should contain a new button and the ability to display the TURing image.

*Figure 6. Example of a modified FirePass login page*




## 6.8 Verifying Installation

Navigate to the F5 interface login page. The customisation is visible in the addition of a **One Time Code Image** button. Only when a correct PINsafe one time code is entered should the user be logged in. This can be done either by entering the OTC for a dual channel login, or selecting OTC Image and entering the OTC for a single channel login.

## 6.9 Troubleshooting

Check the PINsafe logs for any failure information.

## 6.10 Additional Information

For assistance in the PINsafe installation and configuration please contact your reseller or email Swivel Secure support at support@swivelsecure.com

# 7 F5 SAM Integration

# 8 F5 Secure Access Manager (SAM) Integration Notes

This article describes how to integrate the F5 Secure Access Manager with Swivel. The article covers two aspects:

- The integration of the two servers so that the F5 uses Swivel as its RADIUS server
- The modification of the F5 login page to include the TURing image or other Swivel elements as required.

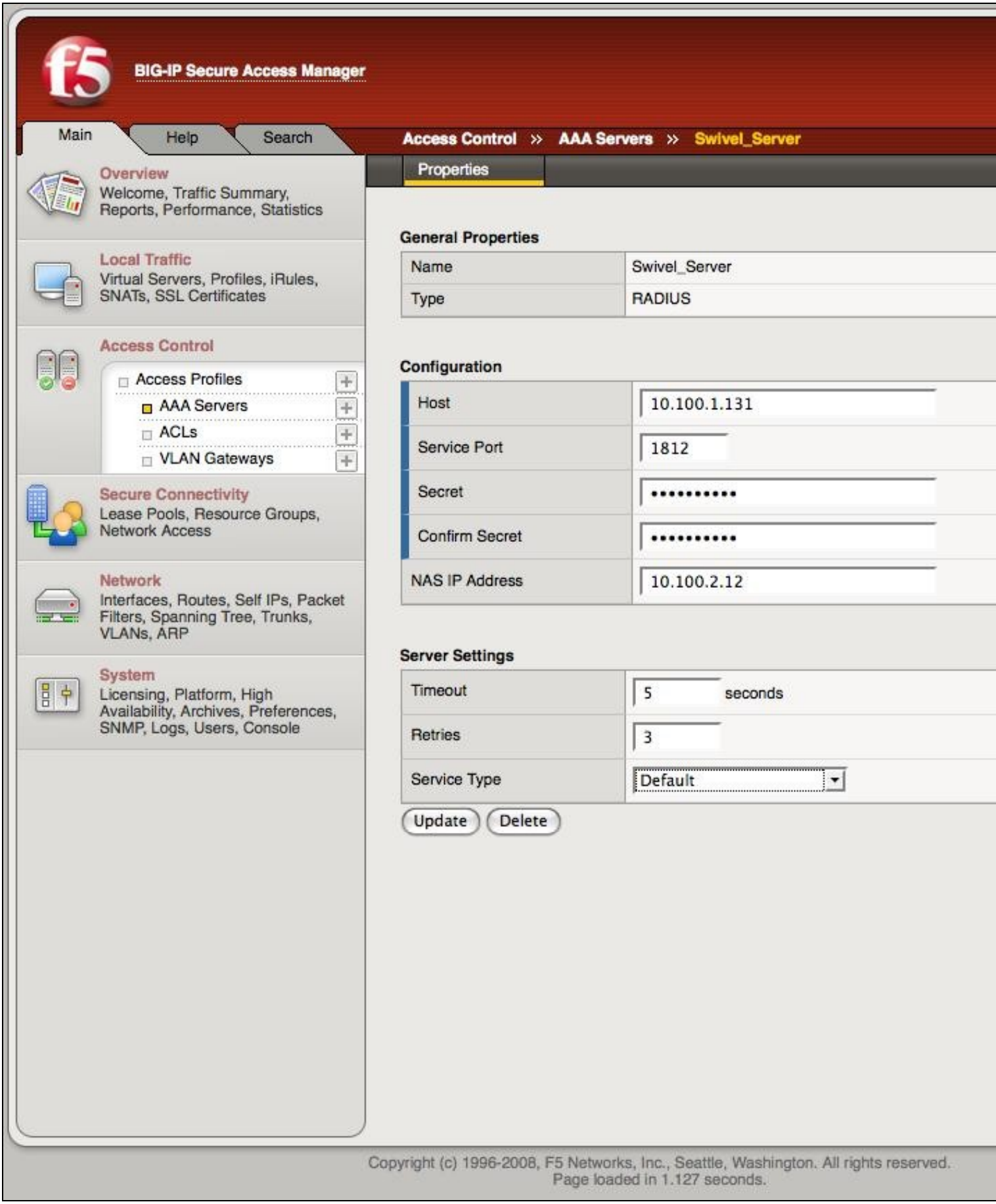
# 9 RADIUS Integration

To use Swivel with F5 SAM you need to enable the Radius Server on PINsafe. (On the RADIUS->Server page)

A NAS Entry then need to be created that includes the F5 server ip address/hostname and a shared secret.

The associated configuration then needs to be created on the F5 server.

This is done on the Access-Control->AAA-Servers screen.




Once this entry has been created it can be used when defining Access Profiles.

It is important to remember the name of the profile created as this will be required for the customisation.

## 9.1 Test the RADIUS authentication

At this stage it should be possible to authenticate by SMS, hardware Token, Mobile Phone Client and Taskbar to verify that the RADIUS authentication is working for users. Browse to the SSL VPN login page, and enter Username and if being used, the password. From the Swivel Administration console select User Administration and the required user then View Strings, and select an appropriate authentication string or OTC for the user. At the SSL VPN login enter the required OTC. Check the Swivel logs for a RADIUS success or rejected message. If no RADIUS message is seen, check that the Swivel RADIUS server is started and that the correct ports are being used.

# 10 Log-in page Customisation

To modify the log-on page to include a TURing image you need secure-shell access (SSH) to the server.

Assuming that the Access Profile is called **pinsafe** the modifications are implemented by editing the file

/config/customization/advanced/logon/**pinsafe**_act_logon_page_ag/logon_en.inc

The steps are as follows.

1. Change directory to the required location

```
cd /config/customization/advanced/logon/pinsafe_act_logon_page_ag
```

2. Take a back-up of the existing file. Note that this example assumes that the Access Policy uses English. If another language is specified then you need to edit the corresponding file, eg log_fr.inc for French.

```
cp log_en.inc tmp_logon_en.inc
```

3. Edit the login file or copy a modified version of the file onto the server.

An example modified script is shown here. The required modifications are between the lines of asterisks. The setting of the sUrl variable needs to correspond to the PINsafe server being used.

4. To register the changes the following commands must be executed.

```
b customization group pinsafe_act_logon_page_ag action update
b profile access pinsafe generation action increment
```

# 11 Testing

Now when the F5 server is accessed via the **pinsafe** access policy the user should see a modified login page with the option to request a TURing image.

The user should enter their username and see a TURing image when the clcik the TURing button. At this point a Session Start message for the user should show in the PINsafe logs.

If no image shows, check that the URL is correct and ensure that there is no firewalls blocking the request.

Also check that Session Create by Username is enabled on the PINsafe server.

The user should then enter their one-time code. The login.

If the log-in fails, check the Swivel log files to see if a RADIUS request was logged. If not then check the settings for the RADIUS on F5 and Swivel to ensure the IP Addresses, port numbers and shared secrets all match. Also check that no firewalls are blocking the RADIUS requests.

If RADIUS attempts are being logged but authentication is failing, check that the session was started correctly and that there is no password associated with the account etc.