Progress®
Kemp® LoadMaster®

Technical Note LoadMaster Duo Integration Guide

Generated on: 18 December 2024

Copyright

Visit the following page online to see Progress Software Corporation's current Product Documentation Copyright Notice/Trademark Legend: Product Documentation Copyright Notice & Trademarks | Progress

Table of Contents

Chapter 1: LoadMaster Duo Two Factor Authentication 4

Add an Application to Duo 6

Install Duo Auth Proxy on Linux 6

Create an Application in Duo 7

Configure Duo Auth Proxy and Start 9

Configure the LoadMaster 12

Create the Duo Image Set 14

Add the Image Name to the Manifest 17

Modify lm_initial_dfa.html 18

Chapter 2: References 39

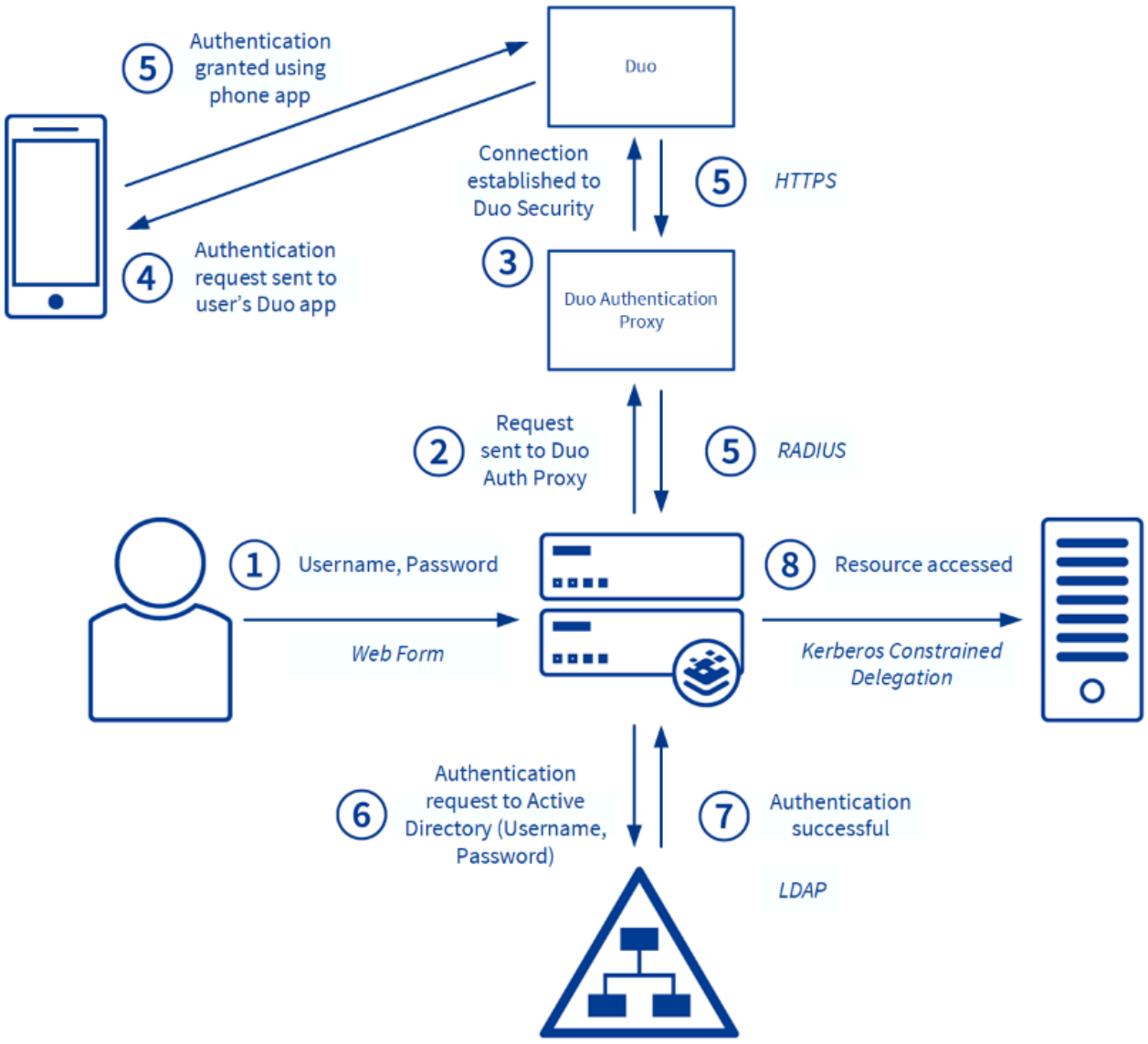
1

LoadMaster Duo Two Factor Authentication

LoadMaster Duo Two Factor Authentication

In this guide we will configure Duo Push along with username and password validation through Active Directory to implement a two factor authentication for our sample application. In this example, the authentication used with the application server is Kerberos Constrained Delegation (KCD).

Once configured the user flow for accessing the application will look like this:



The flow is outlined below:

1. The user provides their username and password.
2. The LoadMaster queries the Duo RADIUS proxy for the user.
3. A connection to Duo Security is established.
4. An authentication request is sent to the user's Duo app.
5. The user validates the request on the app.
6. The LoadMaster performs pre-authentication using the supplied username and password with the Local Domain Controller.
7. Authentication is successful.
8. The resource is accessed through Kerberos Constrained Delegation.

Related Links

- Add an Application to Duo
- Create the Duo Image Set

Add an Application to Duo

Add an Application to Duo

Add an Application to Duo

First you must log in to the Duo Admin Panel and navigate to **Applications > Protect an application**. Click **Radius**. Then, follow the remaining steps in the sub-sections below.

Note: RADIUS is the only required application that should be listed in the Duo Admin Panel.

Related Links

- Install Duo Auth Proxy on Linux
- Create an Application in Duo
- Configure Duo Auth Proxy and Start
- Configure the LoadMaster

Install Duo Auth Proxy on Linux

Install Duo Auth Proxy on Linux

The following Duo guide outlines the steps on installing Duo Authentication Proxy: Authentication Proxy - Reference.

Below is an example configuration using CentOS with Wget installed:

```
yum install gcc make libffi-devel perl zlib-devel

wget https://dl.duosecurity.com/duoauthproxy-latest-src.tgz

tar xzf duoauthproxy-latest-src.tgz

cd duoauthproxy-5.1.1-7484191-src/

make

cd duoauthproxy-build/

./install
```

At this point step through the prompts, for example:

```
In what directory do you wish to install the Duo Authentication Proxy?
```

```
[/opt/duoauthproxy]

Enter the name of a user account under which the Authentication Proxy should be
run. We recommend a non-privileged and locked down account.

Or you can press <Enter> and our default locked down user will be created for
you:

[duo_authproxy_svc]

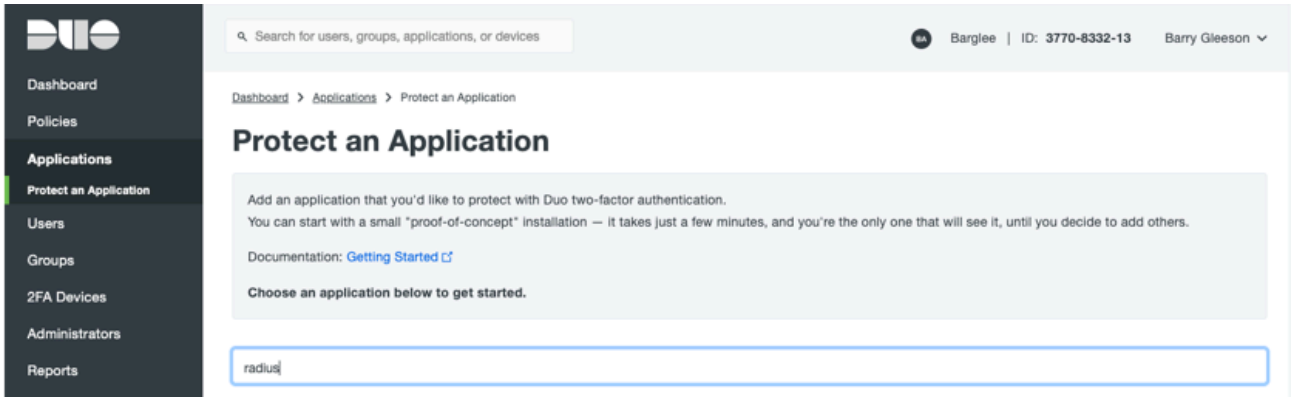
Enter the name of a group under which the Authentication Proxy logs will be
readable. Or press <Enter> and a default group will be created for you:

[duo_authproxy_grp]
```

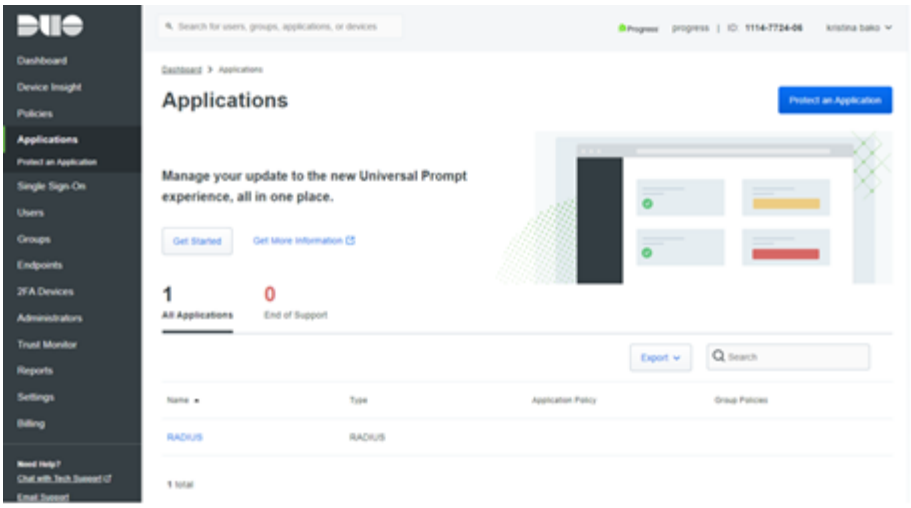
Create an Application in Duo

Create an Application in Duo

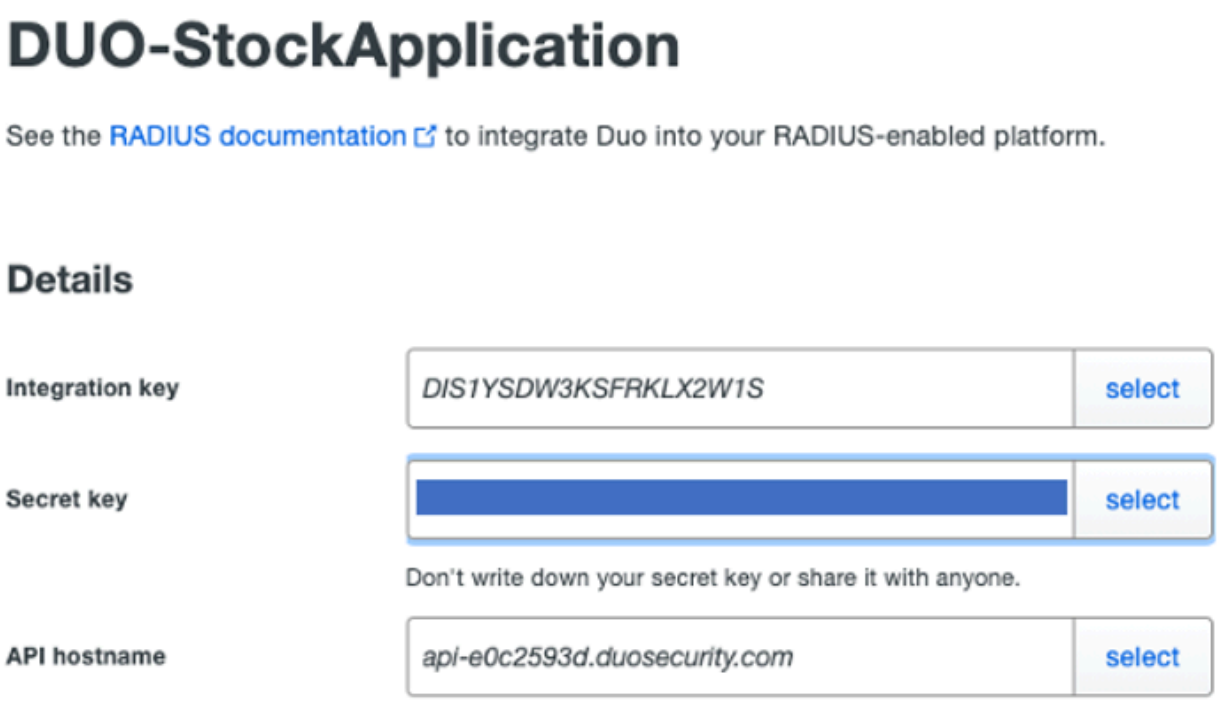
To create an application in Duo, follow these steps:



1. Log in to Duo and go to **Dashboard > Applications > Protect an Application** and search for **radius**.



2. Set the app Name and Users that can access the app.



3. Copy the values for:
   - Integration key
   - Secret key
   - API hostname

Configure Duo Auth Proxy and Start

Configure Duo Auth Proxy and Start

To configure the Duo auth proxy, you must modify the **authproxy.cfg** file. For example:

```
vi /opt/duoauthproxy/conf/authproxy.cfg
```

Below is an example configuration containing the details copied above in addition to the interface address of the LoadMaster which is the RADIUS Client (**radius_ip_1**) and the RADIUS shared secret to use (**radius_secret_1**).

The configuration file should contain the following:

```
[main]

debug=true

test_connectivity_on_startup=true

[duo_only_client]

[radius_server_auto]

ikey=DIS1YSDW3KSFRKLX2W1S

skey=eEqgB1BUP7dfasdasdasdhAPDZOSwLvLp

api_host=api-e0c2593d.duosecurity.com

radius_ip=10.1.151.61

radius_secret_1=Hummingbird

client=duo_only_client

port=1812
```

In the example above:

- The **radius_ip** is the LoadMaster's IP address (or the shared IP address in a High Availability (HA) configuration).
- The **radius_secret_1** is a chosen password. This is required to be configured on the LoadMaster as the **RADIUS Shared Secret** in the SSO Domain configuration of the domain **LDAPDUO** which is mentioned below.

Related Links

- Add a Firewall Rule to Allow Inbound RADIUS

- Start Duo Auth Proxy

Add a Firewall Rule to Allow Inbound RADIUS

Add a Firewall Rule to Allow Inbound RADIUS

This may vary across Linux OSS:

```
firewall-cmd --add-service=radius --permanent

sudo firewall-cmd --reload
```

Start Duo Auth Proxy

Start Duo Auth Proxy

Whenever changes are made the configuration you must stop and start the process.

```
[root@localhost log]# /opt/duoauthproxy/bin/authproxyctl start

Running The Duo Authentication Proxy Connectivity Tool. This may take several
minutes...

[info] Testing section 'main' with configuration:

[info] {'debug': 'True', test_connectivity_on_startup': 'true'}

[info] There are no configuration problems

[info] ------------------------------

[info] Testing section 'duo_only_client' with configuration:

[info] {'debug': 'True'}

[info] There are no configuration problems

[info] ------------------------------

[info] Testing section 'radius_server_auto' with configuration:

[info] {'api_host': 'api-e0c2593d.duosecurity.com',

'client': 'duo_only_client',

'ikey': 'DIS1YSDW3KSFRKLX2W1S',

'port': '1812',
```

```
'radius_ip_1': '10.1.151.61',

'radius_secret_1': '*****',

'skey': '*****[40]'}

[info] There are no configuration problems

[info] ----------------------------

[info] Testing section 'main' with configuration:

[info] {‘debug’: ‘True’, 'test_connectivity_on_startup': 'true'}

[info] There are no connectivity problems with the section.

[info] ----------------------------

[info] Testing section 'duo_only_client' with configuration:

[info] {‘debug’: ‘True’}

[info] No testing to be done for section.

[info] ----------------------------

[info] Testing section 'radius_server_auto' with configuration:

[info] {'api_host': 'api-e0c2593d.duosecurity.com',

'client': 'duo_only_client',

'ikey': 'DIS1YSDW3KSFRKLX2W1S',

'port': '1812',

'radius_ip_1': '10.1.151.61',

'radius_secret_1': '*****',

'skey': '*****[40]'}

[info] The RADIUS Server has no connectivity problems.

[info] ----------------------------

[info] SUMMARY

[info] No issues detected
```

```
The results have also been logged in /opt/duoauthproxy/log/
connectivity_tool.log

Checking updates for Duo Authentication Proxy...

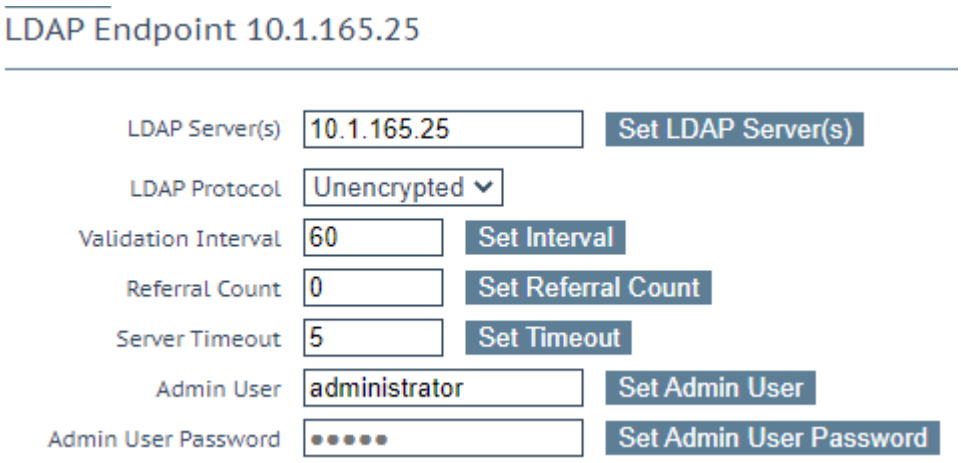
[info] No updates detected. Your Duo Authentication Proxy is up to date.
```

Configure the LoadMaster

Configure the LoadMaster

Create an SSO domain using LDAP and RADIUS. To do this, follow the steps below:

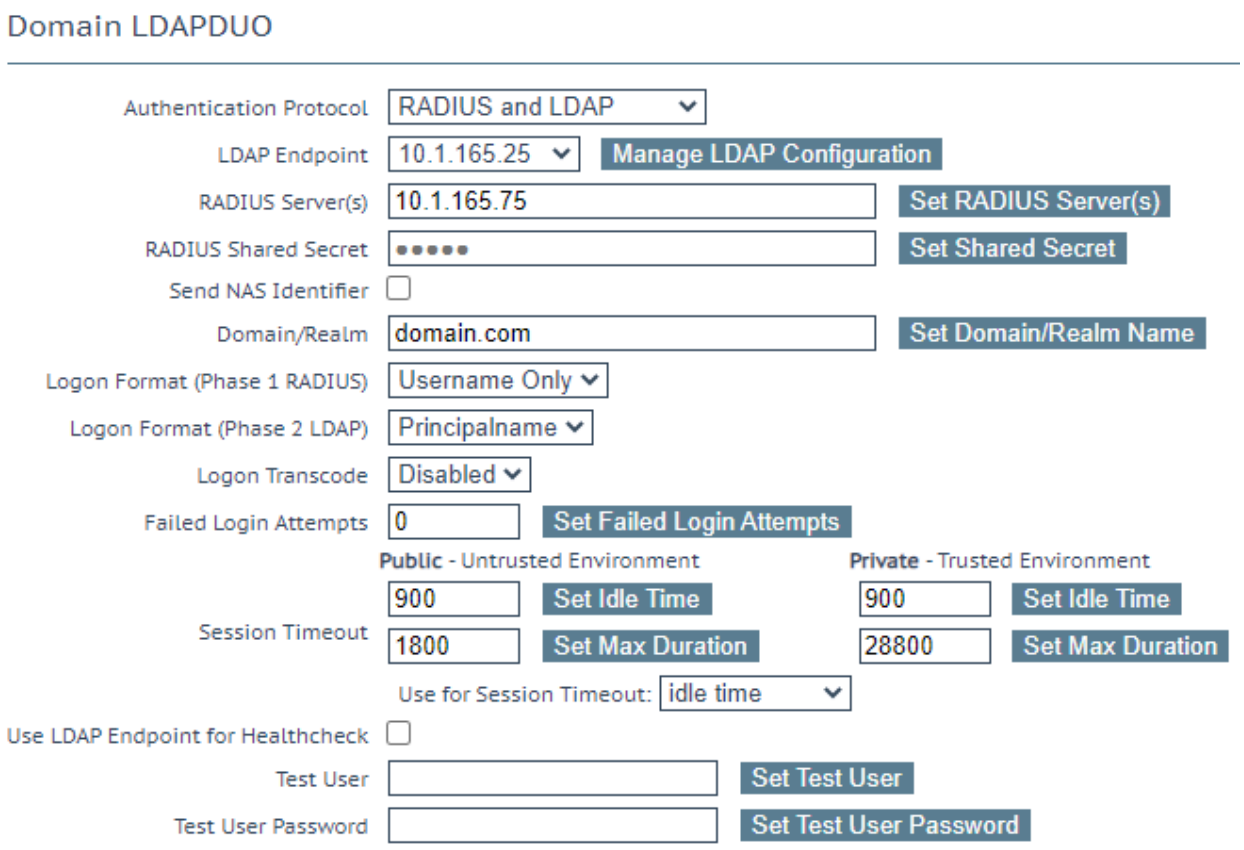
1. In the LoadMaster User Interface (UI), go to **Certificates & Security > LDAP Configuration**.
2. Specify the name of the LDAP endpoint configuration and click **Add**.



3. Configure the settings of the LDAP endpoint as needed.
4. In the LoadMaster UI, go to **Virtual Services > Manage SSO**.



5. Enter the name of the SSO configuration in the text box under **Add new Client Side Configuration** and click **Add**.

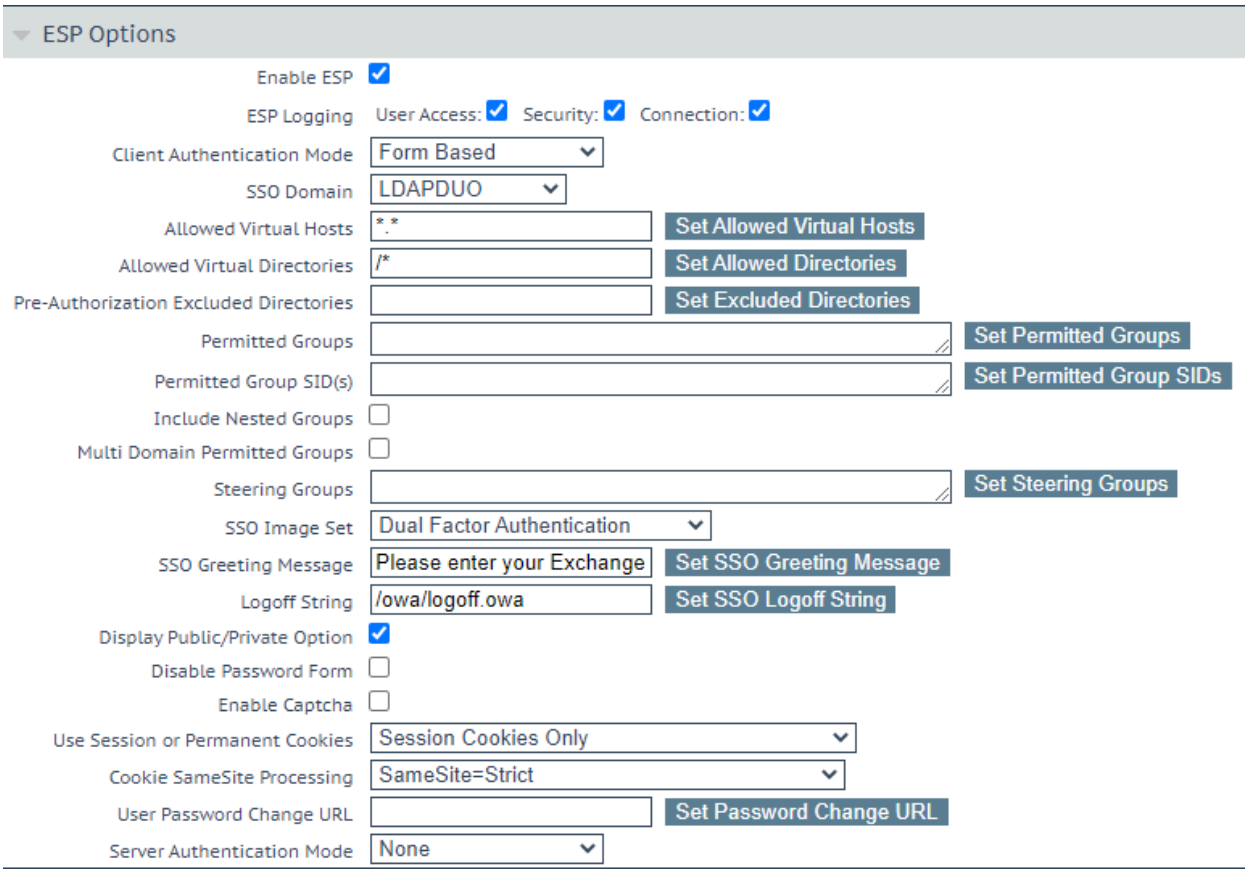


Note: The LDAP server does not need to be the same as the RADIUS server. For example, it can be an LDAP Windows Server that is already used in the domain.

6. Select **RADIUS and LDAP** as the **Authentication Protocol**.
7. Select the relevant **LDAP Endpoint**.
8. Configure the other settings as needed.

Note: The **RADIUS Shared Secret** should be the same as the one configured for **radius_secret_1** as mentioned in the Configure Duo Auth Proxy and Start section.

9. In the LoadMaster UI, go to **Virtual Services > View/Modify Services**.
10. Click **Modify** on the relevant Virtual Service (or add a new one).
11. Expand the **ESP Options** section.



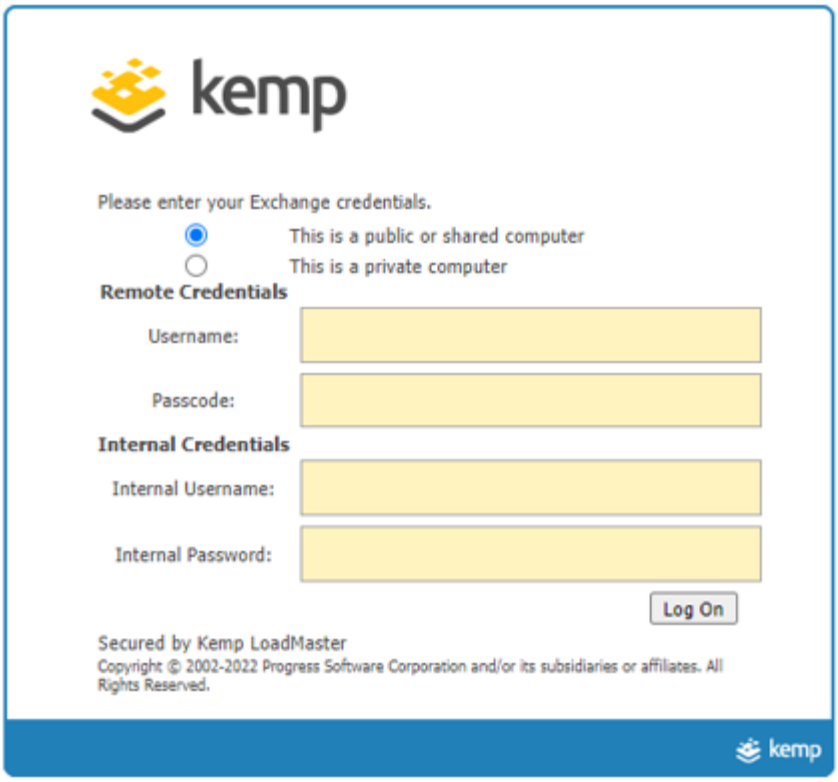
12. Select the **Enable ESP** check box.
13. Select the relevant **SSO Domain**.
14. Configure any other settings as needed.

Note: Any Real Server can be added to this service now. There are no additional configuration or installation requirements needed on the Real Server.

Create the Duo Image Set

Create the Duo Image Set

The default **Dual Factor Authentication SSO Image Set** contains two authentication fields for authentication:



For Duo, this image set should be edited to one field of user/password, because the second authentication is using the Duo RADIUS proxy for the authentication.



The image prompt has no other functionality other than to indicate that connection to Duo has been established successfully. When the DUO image displays after login, a notification on the mobile device should have been received.

In this example, the custom image form is modified to include just one **Username** and one **Password** input field and an image is added to indicate to the user that they must approve access (on their mobile device) in response to the initiated Duo push.

The image being displayed in this example is **PhoneApprove.png** and must be added to the custom image set manifest directory as described in the following section.

For details on how to modify a custom image set, refer to the following document: Custom Authentication Form Technical Note. The edits to the Dual Factor Authentication – Custom files are described in this document.

Two files must be modified: **MANIFEST** and **lm_initial_dfa.html**.

Related Links

- Add the Image Name to the Manifest
- Modify lm_initial_dfa.html

Add the Image Name to the Manifest

Add the Image Name to the Manifest

Add the image used to the manifest and include in the imageset/Duo directory, for example, **PhoneApprove.png**.

```
# Default manifest of all files that are part of the logon

# screens.

#ident "$Id: DFA.manifest 16042 2018-02-28 10:16:37Z phil $ "

#

# Format is "filename filetype".

# The first file is the initial login screen

# The second file is the logout screen.

#

lm_initial_dfa.html

lm_logout_dfa.html Text/html

lm_sso.js

espblank.gif

espbottom.gif

esptop.gif

favicon.ico

kmgerror.gif

kmgexleft.gif

kmgexlogo.gif

kmgexright.gif

kmgleft.gif

kmgright.gif
```

```
kmgstyle.css

PhoneApprove.png
```

When imagesets are created they exist within an imageset folder. Each specific imageset has its own folder within the imageset folder.

Modify lm_initial_dfa.html

Modify lm_initial_dfa.html

Add JavaScript

Within the **<head>** portion of the page, add the following JavaScript to enable dynamic display of the Duo image and hiding of the button once credentials are entered.

Add:

```
<script type="text/javascript">

function picture(){



var pic = "/lm_auth_proxy?LMimage=PhoneApprove.png"

document.getElementById('bigpic').src = pic.replace('90x90', '225x225');

document.getElementById('bigpic').style.display='block';

}

</script>

<script>

function hidebutton(button){

button.style.visibility = "hidden";

}
```

Edit Username

Replace:

```
<form action="/lm_auth_proxy?LMLogon" method="post" id="logonForm"
```

```
autocomplete="off" onSubmit="return save_usernames_dfa(this.dusername,
this.username, this.pubpriv);">
```

With:

```
<form action="/lm_auth_proxy?LMLogon" method="post" id="logonForm"
autocomplete="off" onSubmit="return save_usernames_dfa(this.dusername,
this.dusername, this.pubpriv);">
```

Remove remotecredsid

Remove the following section because there is no need for separate remote credentials:

```
<tr id="remotecredsid">

<td class="nowrap"><label for="remotecreds"><b>Remote
Credentials</b></label></td>

</tr>
```

Replace dpasscodeid

Replace :

```
<tr id="dpasscodeid">

<td class="nowrap"><label for="dpasscode">Passcode:</label></td>

<td class="txtpad">

<input class="txt" id="dpasscode" type="password" name="dpasscode"
autocomplete=off required maxlength=128 />

</td>

</tr>
```

With :

```
<tr id="dpasscodeid"><td class="txtpad">

<input class="txt" id="dpasscode" value="XXX" type="hidden" name="dpasscode"
autocomplete=off required maxlength=128 />

</td>

</tr>
```

Remove interncredsid

Remove the second username field:

```
<tr id="interncredsid">

<td class="nowrap"><label for="interncreds"><b>Internal
Credentials</b></label></td>

</tr>
```

Remove Displaying of the Internal Username

Remove displaying of the internal username because we will reuse a single username.

Replace:

```
<tr id="userid">

<td class="nowrap"><label for="username">Internal Username:</label></td>

<td class="txtpad">

<input class="txt" id="username" name="username" type="text"/>

</td>

</tr>
```

With:

```
<tr id="userid"><td class="txtpad">

<input class="txt" id="username" name="username" type="text"/>

</td>

</tr>
```

This field is not being used. The first username is used for both RADIUS and LDAP authentication.

Update the Internal Password Field

Update the internal password field to simply call it password.

Replace:

```
<tr id="passid">
```

```
<td class="nowrap"><label for="password">Internal Password:</label></td>

<td class="txtpad">

<input class="txt" id="password" type="password" name="password"
autocomplete=off required maxlength=128 />

</td>

</tr>
```

With:

```
<tr id="passid">

<td class="nowrap"><label for="password">Password:</label></td>

<td class="txtpad">

<input class="txt" id="password" type="password" name="password"
autocomplete=off required maxlength=128 />

</td>

</tr>
```

Replace the Final Row

Replace :

```
<tr>

<td class="nowrap"> </td>

<td class="txtpad">

<input type="submit" value="Log On" name="submit" />

</td>

</tr>
```

With :

```
<tr>

<td class="nowrap"> </td>

<td class="txtpad">
```

```
<input type="submit" value="Log On" onclick="hidebutton(this);picture();"
name="submit" />

</td>

</tr>

<tr id="interncredsid">

<td class="nowrap" align="center">

<label for="interncreds"><b><img        max-width:45px; max-height:45px;
id="bigpic" src="bigpic"></tr>
```

Example of the Updated File

A complete example of an updated file is as follows:

```
<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; CHARSET=utf-8">

<title>Kemp Login Screen</title>

<meta content="NOINDEX, NOFOLLOW" name="Robots">

<link rel="shortcut icon" href="/lm_auth_proxy?LMimage=favicon.ico"
type="image/x-icon">

<link href="/lm_auth_proxy?LMimage=kmgstyle.css" type="text/css"
rel="stylesheet">

<style type="text/css">

body

{

font-family:Tahoma,Arial,Helvetica;

font-size:70%;

}

input, button
```

```
{

font-family:Tahoma,Arial,Helvetica;

}

.mid

{

font-size:70%;

}

input, button, label, table

{

font-size:100%;

}

</style>

<script>

var xx_msg10 = "Login Failed - The security service has blocked your request.
Please contact your System Administrator.<br><br>";

var xx_msg11 = "Login Failed - Please make sure that both your remote and
internal credentials are correct, and then try again.<br><br>";

</script>

<script type="text/javascript" src="/lm_auth_proxy?LMimage=lm_sso.js"></script>

<script type="text/javascript">

function picture(){

var pic = "/lm_auth_proxy?LMimage=PhoneApprove.png"

document.getElementById('bigpic').src = pic.replace('90x90', '225x225');

document.getElementById('bigpic').style.display='block';

}
```

```
</script>

<script>

function hidebutton(button){

button.style.visibility = "hidden";

}

</script>

</head>

<body><noscript>

<div id="dvErr">

<table cellpadding="0" cellspacing="0">

<tr>

<td><img src="/lm_auth_proxy?LMimage=kmgerror.gif"><td></tr>

</table>

</div>

</noscript>

<form action="/lm_auth_proxy?LMLogon" method="post" id="logonForm"
autocomplete="off" onSubmit="return save_usernames_dfa(this.dusername,
this.dusername, this.pubpriv);">

<input type="hidden" id="curl" name="curl" value="">

<input type="hidden" id="curlid" name="curlid" value="">

<input type="hidden" id="curlmode" name="curlmode" value="0">

<table align="center" id="tblMain" cellpadding=0 cellspacing=0>

<tr>

<td colspan=3>

<table cellspacing=0 cellpadding=0 class="tblLgn">

<tr><td><img src="/lm_auth_proxy?LMimage=esptop.gif"></td>
```

```
</tr>

</table>

</td>

</tr>

<tr>

<td id="mdLft"> </td>

<td id="mdMid">

<table class="mid">

<tbody>

<tr>

<td class="align">

<table cellpadding="0" cellspacing="0">

<tr><td id="ssomsg"></td></tr>

</table>

</td>

</tr>

<tr id="reset_pass"><td class="align">

<table cellpadding="0" cellspacing="0">

<tr><td>

<span><span><span><span><span><span id="reset_msg"> </span> 

<a id="reset_link" href="#">Click Here</a>

</td></tr>

</table>

</td>

</tr>
```

```
<tr>

<td class="align">

<table cellpadding="0" cellspacing="0">

<tr><td class="wrng" id="badmsg"></td></tr>

</table>

</td>

</tr>

<tr>

<td>

<table cellspacing="0" cellpadding="0">

<colgroup>

<col class="nowrap">

<col class="w100">

<col>

<tbody>

<tr id="nopub"><td class="nowrap">

<input type=radio id=pubr name="pubpriv" value=0 checked="checked">

</td><td class="align"><label for="pubr">This is a public or shared
computer</label></td>

</tr>

<tr id="nopub1"><td class="nowrap">

<input type=radio id=pubp name="pubpriv" value=1>

</td><td class="align"><label for="pubp">This is a private
computer</label></td>

</tr>

<tr id="duserid">
```

```
<td class="nowrap"><label for="dusername">Username:</label></td>

<td class="txtpad">

<input class="txt" id="dusername" name="dusername" type="text"/>

</td>

</tr>

<tr id="dpasscodeid"><td class="txtpad">

<input class="txt" id="dpasscode" value="XXX" type="hidden" name="dpasscode"
autocomplete=off maxlength=128 />

</td>

</tr>



<tr id="userid"><td class="nowrap"><label for="username">Ignore
Username:</label></td>

<td class="txtpad">

<input class="txt" id="username" name="username" type="text"/>

</td>

</tr>

<tr id="passid">

<td class="nowrap"><label for="password">Password:</label></td>

<td class="txtpad">

<input class="txt" id="password" type="password" name="password"
autocomplete=off required maxlength=128 />

</td>

</tr>



<tr>
```

```
<td class="nowrap"> </td>

<td class="txtpad">

<input type="submit" value="Log On" onclick="hidebutton(this);picture();"
name="submit" />

</td>

</tr>

<tr id="interncredsid">

<td class="nowrap" align="center">

<label for="interncreds"><b>

<img src="/lm_auth_proxy?LMimage=wally.png" id="bigpic" max-width:45px max-
height:45px /></b></label></td>

</tr>









</tbody>

</table>

</td>

</tr>

</tbody>

</table>

<div class="g-recaptcha" id=captchabox><table class="mid tblConn">

<tr>

<td class="tdConn"></tr>

<tr>

<td></tr>
```

```
</table>

</td>

<td id="mdRt"> </td>

</tr>

<tr>

<td colspan=3>

<table cellspacing=0 cellpadding=0 class="tblLgn">

<tr>

<td><img src="/lm_auth_proxy?LMimage=espbottom.gif"></tr></table>

</td>

</tr>

</tbody>

</table>

</form>

</body>

</html>
```

```
<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; CHARSET=utf-8">

<title>Kemp Login Screen</title>

<meta content="NOINDEX, NOFOLLOW" name="Robots">
```

```
<link rel="shortcut icon" href="/lm_auth_proxy?LMimage=favicon.ico"
type="image/x-icon">

<link href="/lm_auth_proxy?LMimage=kmgstyle.css" type="text/css"
rel="stylesheet">

<style type="text/css">

body

{

font-family:Tahoma,Arial,Helvetica;

font-size:70%;

}

input, button

{

font-family:Tahoma,Arial,Helvetica;

}



.mid

{

font-size:70%;

}



input, button, label, table

{

font-size:100%;

}



</style>
```

```
<script>

var xx_msg10 = "Login Failed - The security service has blocked your request.
Please contact your System Administrator.<br><br>";

var xx_msg11 = "Login Failed - Please make sure that both your remote and
internal credentials are correct, and then try again.<br><br>";

</script>







<script type="text/javascript">



function picture(){



var pic = "/lm_auth_proxy?LMimage=PhoneApprove.png"



document.getElementById('bigpic').src = pic.replace('90x90', '225x225');



document.getElementById('bigpic').style.display='block';



}



</script>



<script>



function hidebutton(button){
```

```
button.style.visibility = "hidden";


}


</script>









<script type="text/javascript" src="/lm_auth_proxy?LMimage=lm_sso.js"></script>

</head>

<body><noscript>

<div id="dvErr">

<table cellpadding="0" cellspacing="0">

<tr>

<td><img src="/lm_auth_proxy?LMimage=kmgerror.gif"><td></tr>

</table>

</div>

</noscript>

<form action="/lm_auth_proxy?LMLogon" method="post" id="logonForm"
autocomplete="off" onSubmit="return save_usernames_dfa(this.dusername,
this.dusername, this.pubpriv);">

<input type="hidden" id="curl" name="curl" value="">

<input type="hidden" id="curlid" name="curlid" value="">
```

```
<input type="hidden" id="curlmode" name="curlmode" value="0">

<table align="center" id="tblMain" cellpadding=0 cellspacing=0>

<tr>

<td colspan=3>

<table cellspacing=0 cellpadding=0 class="tblLgn">

<tr><td><img src="/lm_auth_proxy?LMimage=esptop.gif"></td>

</tr>

</table>

</td>

</tr>

<tr>

<td id="mdLft"> </td>

<td id="mdMid">

<table class="mid">

<tbody>

<tr>

<td class="align">

<table cellpadding="0" cellspacing="0">

<tr><td id="ssomsg"></td></tr>

</table>

</td>

</tr>

<tr id="reset_pass"><td class="align">

<table cellpadding="0" cellspacing="0">

<tr><td>
```

```
<span>

<span><span><span><span>

<span id="reset_msg"> </span> 

<a id="reset_link" href="#">Click Here</a>

</td></tr>

</table>

</td>

</tr>

<tr>

<td class="align">

<table cellpadding="0" cellspacing="0">

<tr><td class="wrng" id="badmsg"></td></tr>

</table>

</td>

</tr>

<tr>

<td>

<table cellspacing="0" cellpadding="0">

<colgroup>

<col class="nowrap">

<col class="w100">

<col>

<tbody>

<tr id="nopub"><td class="nowrap">

<input type=radio id=pubr name="pubpriv" value=0 checked="checked">
```

```
</td><td class="align"><label for="pubr">This is a public or shared
computer</label></td>

</tr>

<tr id="nopub1"><td class="nowrap">

<input type=radio id=pubp name="pubpriv" value=1>

</td><td class="align"><label for="pubp">This is a private
computer</label></td>

</tr>









<tr id="duserid">

<td class="nowrap"><label for="dusername">Username:</label></td>

<td class="txtpad">

<input class="txt" id="dusername" name="dusername" type="text"/>

</td>

</tr>



<tr id="dpasscodeid"><td class="txtpad">

<input class="txt" id="dpasscode" value="XXX" type="hidden" name="dpasscode"
autocomplete=off required maxlength=128 />

</td>

</tr>
```

```
<tr id="userid"><td class="txtpad">

<input class="txt" id="username" name="username" type="text"/>

</td>

</tr>

<tr id="passid">

<td class="nowrap"><label for="password">Password:</label></td>

<td class="txtpad">

<input class="txt" id="password" type="password" name="password"
autocomplete=off required maxlength=128 />

</td>

</tr>



<tr>

<td class="nowrap"> </td>

<td class="txtpad">

<input type="submit" value="Log On" onclick="hidebutton(this);picture();"
name="submit" />

</td>

</tr>

<tr id="interncredsid">

<td class="nowrap" align="center"><label for="interncreds"><b><img     max-
width:45px; max-height:45px; id="bigpic" src="bigpic"></tr>
```

```
</tbody>

</table>

</td>

</tr>

</tbody>

</table>

<div class="g-recaptcha" id=captchabox><table class="mid tblConn">

<tr>

<td class="tdConn"></tr>

<tr>

<td></tr>

</table>

</td>

<td id="mdRt"> </td>

</tr>

<tr>

<td colspan=3>

<table cellspacing=0 cellpadding=0 class="tblLgn">

<tr>

<td><img src="/lm_auth_proxy?LMimage=espbottom.gif"></tr></table>

</td>

</tr>

</tbody>

</table>

</form>
```

```
</body>

</html>
```

2

References

References

For further details, refer to the following Duo document: Authentication Proxy - Reference.