I2 REDCap Azure Deployment Instructions

This repository contains the assets used to build the I2 REDCap Azure environments. The repository is forked from the original REDCap Azure Repo found here: https://github.com/microsoft/azure-redcap-paas

We made several modifications to the original repository to handle our unique deployment scenario. More information about the changes made can be found in `changes.md`.

How-To Deploy a REDCap Environment

There are a few initial steps that must be completed before deploying the main REDCap resources to Azure. In order to to secure the website we will need to provision an SSL Certificate and request a couple of DNS entries. We will also need an App Registration in Azure Active Directory to enable WUSTL Key authentication.

Through out this guide there will be placeholders included in the text. The two most common placeholders are: {env} and {hostname}. The placeholders should be replace with the appropriate value. For example, the dev deployment would use `dev` in place of {env} and `redcapdev.wustl.edu` in place of {hostname}. Other values for placeholders will be explained in the context they are used.

Azure AD App Registration

Each deployment will need an App Registration to be created in Azure Active Directory if one does not already exist. If there is an existing registration, ensure that the correct redirect URI is assigned to the the application. An example of what the URI should be is shown below.

We need to file a Service Now ticket to request these entries. Follow these steps to submit the request:

1. Open https://wustl.service-now.com/sp?id=sc_home
2. Click General Service Request
3. Include this text in the description box.

```
I need to have an App Registration created in Azure AD to allow for Azure
AD/WUSTLKey authentication for a new REDCap deployment. Below are the
details for the registration:

App Name: i2-redcap-{env}
Platform: Web
Redirect URIs:
    https://{hostname}/.auth/login/aad/callback
    https://{hostname}/redcap/.auth/login/aad/callback
Tokens: ID tokens enabled
```

4. Click order now

This ticket will eventually get routed to the Identity Management team for processing. I was told that you can email them directly after submitting the ticket to help speed the process up and ensure they have all of the information they need. The current contacts are: mulchekp@wustl.edu and paul.malawy@wustl.edu.

These registrations will be used during app service creation. So, they will need to be in place before proceeding with the main deployment step. However, we can continue with some additional configuration as described in the next section.

Pre-deployment configuration

We start by creating a resource group, static IP address, and certificate for the deployment. You can use the `scripts/pre-config.sh` script to create he resource group, IP address, and certificate. For example, to pre-configure QA you would issue this command:

```
./scripts/pre-config.sh qa redcapqa.wustl.edu
```

Once created, you will need to login the the Azure portal and link the new certificate to the REDCap key vault. You will also need to copy the domain verification information for the certificate. To do so, follow these steps:

1. Open https://portal.azure.com
2. Select the newly created resource group
3. Select the certificate you just created
4. Click `Certificate Configuration` from the menu on the left
5. Click `Step 1: Store`
6. Select `i2-redcap-keys` from the Key Vault list
7. Exit the Store configuration (X on the upper right)
8. Click `Step 2: Domain Verification`
9. Copy the `Domain Verification Token`
    - *NOTE:* you will need this token when you create a Service Now ticket in the `Additional Details` form field discussed following section.

DNS Entry Creation

Now that we have an IP Address and Certificate created, we can request the necessary DNS entries. We need to have two DNS entries created for each REDCap environment/deployment. One entry is a standard `A` record that will point the hostname to the IP Address we just created.

The second entry is a `TXT` record that will be used to verify ownership of the domain. This is necessary to complete the certificate creation process.

We need to file a Service Now ticket to request these entries. Follow these steps to submit the request:

1. Open https://wustl.service-now.com/sp?id=sc_home
2. Search for `DNS` in the service catalog search box
3. Click `IP Address DHCP and DNS Management`
4. Fill out the form with the following information:

- Date needed: {Select an appropriate date}
- IP address of the device: See below
- Domain name: {hostname}
- Additional details:

```
 Please create the following two DNS entries:

 {hostname}  - A - {ipAddress}

 {hostname} - TXT - {domainVerificationToken}

 awverify.{hostname} - TXT - awverify.i2-redcap-{env}-
web.azurewebsites.net

 Thank you!
```

NOTE: Replace the {Placeholders} with the appropriate values. You may need to return to the Azure Portal to get the IP Address and/or verification token if you do not have them available.

5. Verify that the correct values are included in the additional details and click the `Order Now` button.

Completing Certificate Setup

Once the DNS entries have been created, we can return to the Azure Portal and complete the Certificate setup. There are three tasks to complete the setup. First, the domain verification needs to be completed. Then the certification secret needs to be exported. Finally we import the secret into the Key Vault certificates. This final step allows the Application Gateway to use the certificate that has been issued for the hostname we specified during pre-configuration.

Follow these steps to finalize the domain verification.

1. Open https://portal.azure.com
2. Select the appropriate resource group
3. Select the certificate that was created previously
4. Click `Step 2: Domain Verification`
5. Click `Refresh` if necessary to complete the verification process

Now we need to export the certificate private key (aka secret) from the Key Vault.

1. Click `Export Certificate`
2. Click `Open Key Vault Secret`
3. Click on the `Current Version` of the secret
   - *NOTE:* the current version should show an `Activation date` and an `Expiration date`
4. Scroll down and click the `Download as a certificate` button
5. Save the `pfx` file to a local folder

Finally import the `pfx` file into the Key Vault certificates.

1. Open the `Certificates` section of the i2-redcap-keys Key Vault

2. Click `Generate/Import`
3. Select `Import`
4. Certificate Name: `i2-redcap-{env}-gateway-cert`
5. Certificate File: select the `pfx` file you exported
6. Password: {leave blank}
7. Click `Create`

Deploy Main Template and Configure app

Now that all of the necessary configuration is done, we can deploy the main REDCap resources. To do this all we will need to is run the deployment script with a few parameters. The parameters are the {env} and the password to use for admin access to the database.

```
./scripts/deploy-redcap.sh {env} {adminPassword}
```

This script will take several minutes to run. Once it is complete you will be able to browse to the new REDCap deployment using the default URL provided by the app service. It should look like this: https://i2-redcap-{env}-web.azurewebsites.net/index.php

Manual App Configuration

After the application is created we need to do a few manual configuration steps. We will need to add the custom domain to the app service, configure the SSL binding, and enable WUSTL Key authentication. All of these steps can be completed using the Azure Portal.

Add Domain and SSL Binding

1. Open the newly created app service
2. Open the `Custom Domains` view
3. Click `Add custom domain`
4. Enter {hostname}
5. Click `Validate`
6. Click `Add custom domain`
7. Click `TLS/SSL settings`
8. Click `Private Key Certificates (.pfx)`
9. Click `Import Key Vault Certificate`
10. Select the certificate we created for this deployment
11. Click `Custom domains`
12. Click `Add binding` under the `SSL Binding` column for the domain you just added
13. Select the certificate and specify `SNI SSL`
14. Save your changes

References Configure Custom Domain Configure SSL

Configure Authentication

1. Open the `Authentication` view for the app service.

2. Click `Add Identity Provider`
3. Select Microsoft for Identity Provider drop down
4. App registration type: `Provide the details of an existing app registration`
5. Application Id:
6. Client Secret:
7. Authentication: `Allow unauthenticated access`
8. Click `Add`

References Configure AAD Authentication

Deploy Gateway Template

The final step is to deploy the Application Gateway. To do this run the following script and provide the {env} and optionally a certificate name. Assuming you used the `pre-config.sh` script the certificate name should be `i2-redcap-{env}-cert`. If no certificate name is provided to the script it will default to that naming pattern.

```
./scripts/deploy-gateway.sh {env} {certificateName}
```

This script will also take a few minutes to complete. Once the script finishes you should be able to browse to https://{hostname} and log into the new REDCap deployment.