I2's ARM Template for REDCap automated deployment in Azure

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 top to secure the website we will need to provision an SSL Certificate and request a couple of DNS entries.

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 request the DNS entries:

1. Open https://wustl.service-now.com/
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:

    Copy and paste the following text and replace the {Placeholders} with the appropriate values:

    > Please create the following two DNS entries:
    >
    > {hostname} - A - {ipAddress}
    >
    > {hostname} - TXT - {domainVerificationToken}
    >
    > Thank you!

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 that will be used for the deployment.

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

- Deploy the main template `./scripts/init env adminPassword`
- Assign certificate to the app service

Deploy Gateway Template

- Copy SecretId from the imported certificate into the gateway parameters `gateway/gateway-template.parameters.json`
- Deploy the gateway template `./scripts/deploy-gateway.sh env`

- auth settings
    - allow unauth
    - aad

DNS

```
- redcapdev.wustl.edu A 20.98.177.239
    - points to the dev application gateway
- redcapqa.wustl.edu A X.X.X.X
    - points to the qa application gateway
- redcap.wustl.edu A X.X.X.X
    - points to the production application gateway
- Additional TXT verification entires will need to be created for each app
service
to allow the custom domain to be assigned.
```