**SBIT Documentation**

Update, polish, and reorganize later

Before using SBIT, you must create an account with Amazon Web Services. You can do thing by going to <https://aws.amazon.com/> and clicking on ‘Sign-Up.’ Once you have an account there are a few tasks that you must complete before running the SBIT script. These are listed below.

**Create an IAM User:**

Before execution of SBIT, users should create a separate IAM user with administrative access:

1. Log in to your AWS console.
2. Scroll down to the section titled **Security, Identity & Compliance** and click on the **IAM** link.
3. In the Navigation Pane on the left, click **Groups**.
4. Click the **Create New Group** Button.
5. Give the group a meaningful name (for example, ‘Administrators’). Click Next.
6. In the search/filter box, type ‘AdministratorAccess’ and tick the box next to **AdministratorAccess**. Click Next.
7. Click **Create Group**.
8. In the Navigation Pane, click **Users**.
9. Click **Add User**.
10. Give the user a name.
11. Under **Select AWS Access Type**, tick both boxes.
12. Give the user a custom password.
13. Untick **Require password reset**. Click Next.
14. Click **Add user to group**.
15. Tick the box next to the group you created in Steps 3-7. Click Next.
16. Verify the information and click **Create User**.
17. Make a note of the user’s Access Key and Secret Key. These are needed in order to use SBIT. Keep these stored in a safe place; if the secret key is lost or otherwise compromised, your account may be vulnerable. At the very least, you will have to generate a new user.
18. Click Close.

**Create a Key Pair:**

1. Log into your AWS Console.
2. Find the **Compute** section and click the **EC2** link.
3. In the Navigation Pane, under **Network and Security**, click **Key Pairs**.
4. Click **Create Key Pair**.
5. Give the key pair a name. The name has little impact but should be meaningful.
6. Be sure to download the key file as you CANNOT download it later. This will be the last time you can download the key file. The key file must be stored in a secure location. If the key file is lost, you must regenerate a new key and must use the AWS Console to reconfigure all instances to use the new key.
7. The name of this key pair will need to be entered when running SBIT.

\*Note: SBIT will use this one key pair to secure access to all instances. Again, all instances will use the same key file to secure access.

**Configuring AWS CLI:**

After a separate IAM account has been created and enabled for programmatic access, the Access and Secret Access Keys must be configured in the AWS CLI.

1. Install Python 3.6.4 onto the computer from which SBIT will be run.
2. Follow Amazon’s online documentation to install the AWS CLI.
3. Install Boto3 (the Python SDK for AWS).
4. Once these three things are installed, open a **Command Prompt**. This can be done by pressing the Windows key and typing **cmd**.
5. When the Command Prompt opens, type **aws configure** and press enter.
6. Enter the Access Key, Secret Key, AWS Region, and Output Format when prompted.
7. When finished, you can close the Command Prompt.

\*Note: The AWS Region determines where the instances created by SBIT will be placed. This should generally be the nearest AWS region to your on-site location. The output type determines how information is returned to the user and is more or less unimportant to the execution of SBIT.

**Required Information (from user):**

Before execution of SBIT, users must have the following information available. The SBIT script will ask for all of this information:

Domain name:

* If you want the domain name to be accessible by the public, for a web server for example, you should purchase a domain name from any reputable domain name registrars. I can personally use GoDaddy.
* If you do not want the domain to be accessible by the public and the domain will be entirely private, you can choose any domain name that you would like, within the standard domain naming rules.

\*Note: At the current stage of development on SBIT, this domain can be anything. Currently, there are no components that need public DNS records, therefore you don’t need to worry about someone else having your domain name.

NetBIOS Name of the Domain:

* This is a shortened version of the domain name and is required for certain steps in SBIT. This name is typically the domain name in all capital letters without the root domain attached.
* Ex. The ‘example.com’ domain name would usually have ‘EXAMPLE’ as the NetBIOS name.

Key Pair name

* You will need the key pair in order to access instances. You must have this in order to manage your servers.
* The key pair must be created before the script is executed. See Create a Key pair

Number of Domain Controllers: (Currently, the script only supports 2 DCs. The number entered here isn’t used.)

* Default: 2
* Can be anywhere between 2 and 8,
  + Two provides redundancy and availability, enough for most very small businesses
  + Four provides even more redundancy
  + More than 4 is likely overkill for most target businesses, save those with extremely high employee counts
* This number will be split across two availability zones

Number of File Servers: (Currently, the script only supports 2 FSs. The number entered here isn’t used.)

* Default: 2
* Can be anywhere between 2 and 4
* These will be split across two availability zones

File Server Storage: (Currently, this number isn’t validated.)

* The size of the second volume (“extra drives”) which will be added to file servers, specifically for shares
* Can be between 10 (GiB) and 1500 (GiB), no default

Exchange Server Storage:

* This parameter determines the size of the root volume (the size of the C drive). This is needed as the default size of 30GB is not enough to install Exchange.

\*Note: 33 is the minimum number that should be entered here. Much more should be entered if the environment is to be used in actual production. 33GB is only enough for basic testing and should not be used in a production deployment.

Instance Types

* DCs, File, and Exchange servers can be set independently
* Instance type determines resources available to server (memory, cpu, etc.) as well as cost.
* Type r4.large is the lowest recommended size for Exchange servers. I have tested with smaller (cheaper) instances, but the Exchange installation or configuration fails, or problems arise without enough resources.
* Type t2.micro lies within the free tier. For the first year of AWS use, users are granted up to 750 hours of free t2.micro use per month. Charges for other services or licenses may still apply.

Domain Administrator Username and Password (Currently, passwords are NOT secured.)

* Domain Admin user will be added to domain and given domain admin permissions
* Safer/more secure than using default Admin account

\*Note: Without this secondary domain admin user, I would need to halt script execution after the Active Directory was configured and ask the user for the default Windows password of the DC1 domain controller instance. This is because the DC1 default password is the password of the default Domain Administrator account. This secondary user allows me to gather information once, and only once, from the user.

Domain Restore Mode Password (Currently, passwords are NOT secured.)

* Required for Active Directory
* Allows users to put domain in “Restore Mode” in case of server impairment or possible attack.