



**DEVICE**  
**AUTHORITY™**  
Trust for every Thing

# KeyScaler 6.3 Installation (Single Tenant) Prerequisites

---

Security Level:	<b>Confidential</b>
Author:	<b>Frode Nilsen</b>
Last Edit Date:	<b>21 January, 2019</b>
<p>© 2018 Device Authority</p> <p><i>This document contains proprietary and confidential information of Device Authority and shall not be reproduced or transferred to other documents, disclosed to others, or used for any purpose other than that for which it is furnished, without the prior written consent of Device Authority. It shall be returned to the respective Device Authority companies upon request.</i></p> <p><i>The trademark and service marks of Device Authority, including the Device Authority mark and logo, are the exclusive property of Device Authority, and may not be used without permission. All other marks mentioned in this material are the property of their respective owners.</i></p>	

## Contents

1	Document Version Control .....	3
2	Assumptions & Constraints.....	4
2.1	Assumptions .....	4
2.2	Constraints.....	4
3	Introduction .....	5
3.1	Document Overview .....	5
3.2	Terms and Definitions.....	5
3.3	Related Documentation.....	5
4	Hardware and Software Requirements .....	6
5	DAE, KMS and CP Installation Prerequisites .....	7
5.1	Overview.....	7
5.2	Install summary .....	8
5.3	Download the installation package from the Device Authority Customer Portal.....	9
5.4	Run the install script .....	10
5.4.1	Unpack downloaded file .....	10
5.4.2	Run the install Script .....	10
5.5	Running over SSL or HTTPS .....	10
5.5.1	Obtaining SSL Certificate.....	11
5.6	Change MySQL Password .....	16
5.7	Start the Tomcat Webserver.....	17
5.8	Verify access to wizard pages .....	19
6	Next Steps.....	20

# 1 Document Version Control

Version	Description
0.1	Initial Document Creation

*Item 1*

## 2 Assumptions & Constraints

### 2.1 Assumptions

No	Description

*Item 2*

### 2.2 Constraints

No	Description

*Item 3*

## 3 Introduction

### 3.1 Document Overview

This document covers the installation prerequisites for KeyScaler's server components of the Device Authority Engine (DAE), Key Management Services (KMS) and Control Panel (CP)..

### 3.2 Terms and Definitions

Term	Meaning
KMS	Key Management Store
DAE	Device Authority Engine
CP	Control Panel
DDKG	Dynamic Device Key Generator

*Item 4*

### 3.3 Related Documentation

Doc #	Title	Comment
[1]	<a href="#">DAKS-62-INST-KEYSCALER_PREREQUISITES</a>	KeyScaler Installation document

*Item 5 – Related Documentation*

## 4 Hardware and Software Requirements

Please review the following to ensure you have met all necessary hardware and software requirements:

<https://deviceauthority.zendesk.com/hc/en-us/articles/217078538-KeyScaler-Hardware-Requirements>

<https://deviceauthority.zendesk.com/hc/en-us/articles/217078568-KeyScaler-Software-Requirements>

# 5 DAE, KMS and CP Installation

## Prerequisites

This document covers the installation prerequisites for KeyScaler's server components of the Device Authority Engine (DAE), Key Management Services (KMS) and Control Panel (CP).

### 5.1 Overview

Acquire a single dedicated Linux server instance using a supported OS version.

This host must be configured to allow HTTP and HTTPS traffic. You'll need root access to this server. For Amazon instances, you can use the `ec2-user` and `sudo su` to gain root access.

Acquire an SSL certificate for this instance.

- You will need a wildcard SSL cert for the server, in p12 format. Note: you can use `openssl` to convert an existing certificate to p12 format.
- For development instances, you can self-sign a certificate. This is not recommended for production systems as browsers will detect a self-signed certificate and issue warnings. Instructions on creating a self-signed certificate are provided below in this document.

The URL structures / naming conventions used for the server are of the form:

- `<master tenant name>.<domain>.com` - must resolve to the server where the CP will be running
- `<first tenant name>.<domain>.com` - must resolve to the server where the CP will be running. The tenant url must be different from the master url. For example: Our convention is `<tenant name>.deviceauthority.com` so we have `keyscalerdemo.deviceauthority.com` and `cp.deviceauthority.com` (for master access).

Two administrator emails and their OS-type if using device authenticated access (Windows or MAC OSX). These individuals will be set up as the initial administrators for the Control Panel. The same email can be used for both master and the first tenant.

Post installation during the [DAE and CP Configuration](#) activities, you'll need SMTP credentials for Control Panel email notifications and administrator invite emails. The following values are needed:

- Protocol (for example SMTP)
- Mail Server Host
- Server Port Number
- From Address
- TLS (check box if using TLS)
- User Name / Password (Optional. Needed if your SMTP host requires authentication.)

## 5.2 Install summary

The install script will be installing and configuring the following components for you:

Synchronizing Time with NTP

1. Installing Java with Cryptography Extension (JCE)
2. Installing Memcached
3. MySQL Installation and Database Creation
4. Installing the Tomcat Webserver
5. Modifying your iptables rules to secure Tomcat

After running the install script, you will:

1. Install the SSL certificate
2. Change the MySQL passwords.
3. Startup Tomcat and verify the Wizard is running. The Wizard is used to facilitate installation of the KeyScaler software modules.



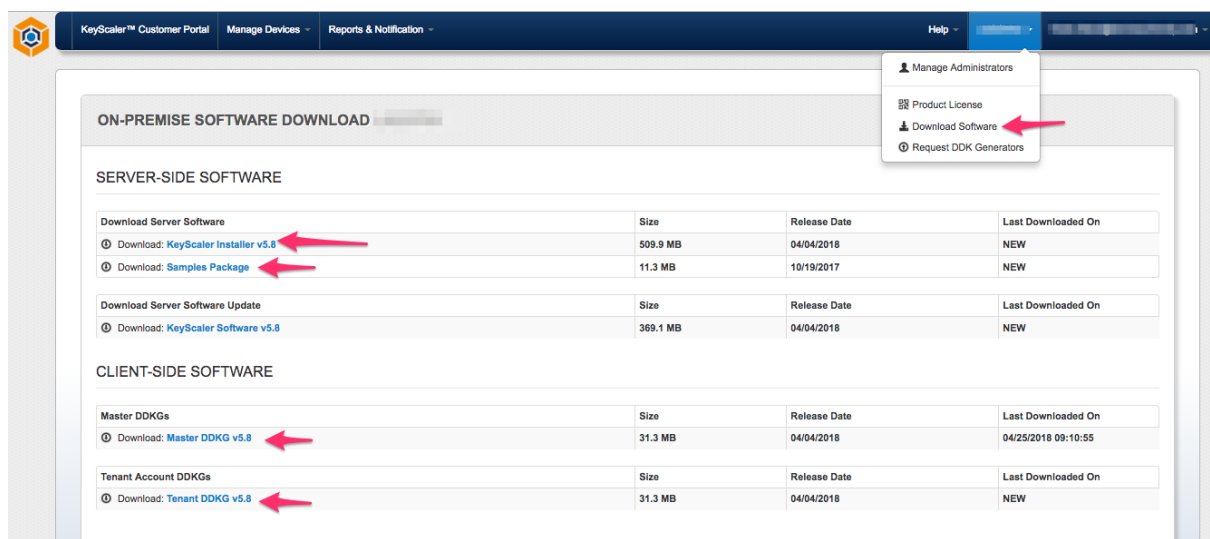
## 5.3 Download the installation package from the Device Authority Customer Portal

Note. This installation document assumes all files are placed in the server's home directory, e.g. /home/ec2-user.

Access the Device Authority Customer Portal (also called the Device Authority Download Center), navigate to the pull-down menu under your tenant name, and select Download Software. Download all the software pointed to by red arrows and transfer all of these files onto your target server to /home/ec2-user



Please note the download links will as of October 2018 read v6.2 not 5.8 (below screenshot is from an old system)



**ON-PREMISE SOFTWARE DOWNLOAD**

SERVER-SIDE SOFTWARE

Download Server Software	Size	Release Date	Last Downloaded On
Download: <a href="#">KeyScaler Installer v5.8</a>	509.9 MB	04/04/2018	NEW
Download: <a href="#">Samples Package</a>	11.3 MB	10/19/2017	NEW
Download Server Software Update	Size	Release Date	Last Downloaded On
Download: <a href="#">KeyScaler Software v5.8</a>	369.1 MB	04/04/2018	NEW

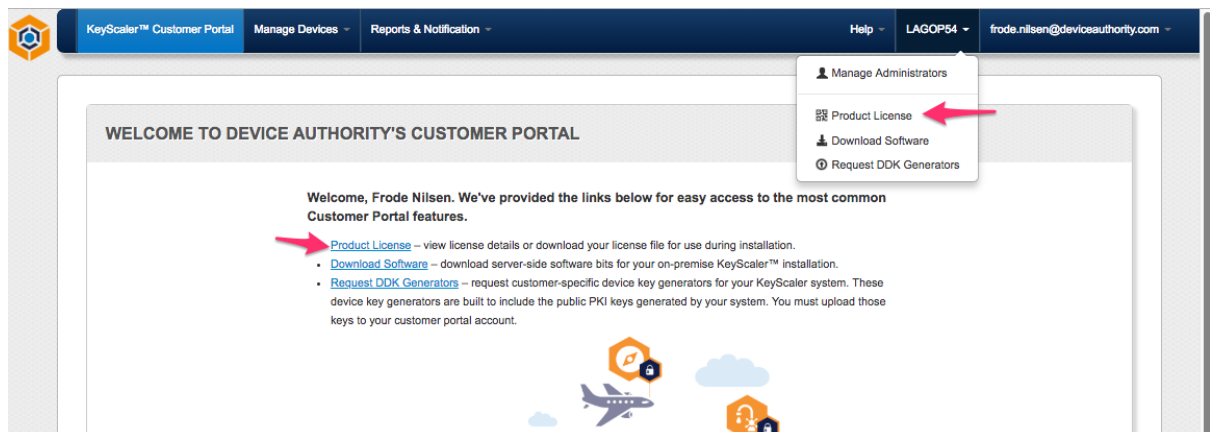
CLIENT-SIDE SOFTWARE

Master DDKGs	Size	Release Date	Last Downloaded On
Download: <a href="#">Master DDKG v5.8</a>	31.3 MB	04/04/2018	04/25/2018 09:10:55
Tenant Account DDKGs	Size	Release Date	Last Downloaded On
Download: <a href="#">Tenant DDKG v5.8</a>	31.3 MB	04/04/2018	NEW

Menu items: Manage Administrators, Product License, Download Software, Request DDK Generators

Item 6 – Download KeyScaler Installer Software

Navigate to the pull-down menu under your tenant name, download the Product License and upload it to your target server directory /home/ec2-user



**WELCOME TO DEVICE AUTHORITY'S CUSTOMER PORTAL**

Welcome, Frode Nilsen. We've provided the links below for easy access to the most common Customer Portal features.

- [Product License](#) – view license details or download your license file for use during installation.
- [Download Software](#) – download server-side software bits for your on-premise KeyScaler™ installation.
- [Request DDK Generators](#) – request customer-specific device key generators for your KeyScaler system. These device key generators are built to include the public PKI keys generated by your system. You must upload those keys to your customer portal account.

Menu items: Manage Administrators, Product License, Download Software, Request DDK Generators

Item 7 – Download product license

## 5.4 Run the install script

Note: if you are installing on an AWS instance and have access using `ec2-user`, first issue the command `sudo su` to become root. These instructions assume you transferred the `tar.gz` file to the `/home/ec2-user` directory.

### 5.4.1 Unpack downloaded file

```
[ec2-user@ip-172-31-4-186 ~]$ sudo su - root
[root@ip-172-31-4-186 ~]# cd /home/ec2-user/
[root@ip-172-31-4-186 ec2-user]# tar -xvzf keyscaler.installer-6.3.tar.gz
```

*Item 8 – Unpack software*

### 5.4.2 Run the install Script

```
[root@ip-172-31-4-186 ec2-user]# cd installer
[root@ip-172-31-4-186 installer]# ./install.sh
```

*Item 9 – Installation script*

## 5.5 Running over SSL or HTTPS

For basic security measures, **Tomcat must be configured to listen over SSL (HTTPS)**. All Device Authority server applications expect to send and receive traffic on a secure channel when deployed.

The Device Authority support team will refer SSL support to the institution that issues the Certificate. The SSL-related instructions below are provided as a reference only.

## 5.5.1 Obtaining SSL Certificate

You will need to obtain an SSL certificate. An SSL certificate is required in order for SSL to work in Tomcat. There are two ways to obtain one, and each is described below:



**Please note:** A default password **'mypassword'** and **'changeit'** is used for the purpose of readability of this document. These passwords must, however, be updated in line with your company's security policy.

### 5.5.1.1 Self-signed certificate



If you are using CA Signed Certificate, skip this section and proceed with 0.

Self-signed certificates are useful in cases where you require encryption but do not need to verify the website identity. They are commonly used for testing and use on internal corporate networks (intranets). Due to the certificate not being signed by a Certification Authority (CA), users will get prompted that the site is untrusted and may have to perform several steps to "accept" the certificate before they can access the site.

The following steps illustrate the creation of a self-signed certificate + key pair converted to PKCS#12 file format (.p12). This file format allows the certificate file to be portable and reused on other Device Authority application servers running Tomcat.

#### 5.5.1.1.1 Create temporary directory

Change to root user – create and move to a temporary directory that will be used to create files generated by the OpenSSL utility. In this example, we're using /var/dfactor/cert.

```
[root@ip-172-31-4-186 installer]# mkdir /var/dfactor/cert  
[root@ip-172-31-4-186 installer]# cd /var/dfactor/cert
```

Item 10

```
[root@ip-172-31-4-186 cert]# yum install openssl -y
```

Item 11

#### 5.5.1.1.2 Generate public and private key pair

Generate a public and a private key pair. Important Note: you'll be prompted to enter a pass phrase for the private key. You will need to remember this pass phrase for the subsequent steps.

```
[root@ip-172-31-4-186 cert]# openssl genrsa -des3 -out my_private.key 4096  
Enter pass phrase for my_private.key: mypassword  
Verifying - Enter pass phrase for my_private.key: mypassword
```

Item 12

### 5.5.1.1.3 Generate a certificate request with the public key included

Generate a certificate request with the public key included. You'll be prompted to enter a Common Name, make sure to use a wildcard domain name like \*.xyzcorp63.com (Item 13) so that you can later use it access the other KeyScaler server applications securely using the same self-signed certificate. For example, <https://mytenant.xyzcorp63.com>

```
[root@ip-172-31-4-186 cert]# openssl req -new -key my_private.key -out my_csr.csr
Enter pass phrase for my_private.key: mypassword

Country Name (2 letter code) [XX]:UK
State or Province Name (full name) []:BE
Locality Name (eg, city) [Default City]:Reading
Organization Name (eg, company) [Default Company Ltd]:fbn
Organizational Unit Name (eg, section) []:Tech
Common Name (eg, your name or your server's hostname) []:*.xyzcorp63.com
Email Address []: your.email@yourcompany.com

A challenge password []:mypassword
An optional company name []:test
```

Item 13

### 5.5.1.1.4 Generate a self-signed certificate

Generate a self-signed certificate based on the certificate request. The example below creates a certificate good for one year.

```
[root@ip-172-31-4-186 cert]# openssl x509 -req -days 365 -in my_csr.csr -signkey
my_private.key -out my_cert.crt
Signature ok
subject=/C=UK/ST=CH/L=Chesterfield/O=invma/OU=Tech/CN=*.invma.com/emailAddress=frode.nilsen
@deviceauthority.com
Getting Private key
Enter pass phrase for my_private.key: mypassword
```

Item 14

### 5.5.1.1.5 Package the private key and certificate together as a PKCS#12

At this point, the following you should have the following are present: `my_private.key`, `my_csr.csr`, and `my_cert.crt`. You are ready to package the private key and certificate together as a PKCS#12 file. Important Note: you'll be prompted to enter an Export Password for the p12 file. You will need to remember this password when configuring Tomcat's SSL Connector.

```
[root@ip-172-31-4-186 cert]# openssl pkcs12 -export -inkey my_private.key -in my_cert.crt -
out self_sign_certificate.p12
Enter pass phrase for my_private.key: mypassword
Enter Export Password: mypassword
Verifying - Enter Export Password: mypassword
[root@ip-172-31-4-186 cert]#
```

Item 15

At this stage, a `self_sign_certificate.p12` file is created with a corresponding password.

```
[root@ip-172-31-4-186 cert]# ls /var/dfactor/cert/  
my_cert.crt  my_csr.csr  my_private.key  self_sign_certificate.p12
```

Item 16

Next, follow the instructions in the Configuring Tomcat's SSL Connector section below.

#### 5.5.1.1.6 Importing the Self-Signed Certificate to the Java Keystore

NOTE: The instructions in this step pertain to self-signed certificates only – you may skip this step if you have configured Tomcat to use a CA signed certificate.

As root user, import your self-signed certificate to the Java keystore. **Please note** ‘changeit’ is the default password and should be updated to comply with your company’s security policies for a production system.

```
[root@ip-172-31-4-186 cert]# cd /var/dfactor/cert  
[root@ip-172-31-4-186 cert]# /usr/java/default/bin/keytool -import -alias dfactor -file  
my_cert.crt -keystore /usr/java/latest/lib/security/cacerts -storepass changeit  
Trust this certificate? [no]: y
```

Item 17

Verify the self-signed certificate was added

```
[root@ip-172-31-4-186 cert]# /usr/java/latest/bin/keytool -list -keystore  
/usr/java/default/lib/security/cacerts -storepass changeit | grep dfactor  
dfactor, Mar 15, 2017, trustedCertEntry,
```

Item 18

The self-signed certificate is now valid for any host under the domain that is associated with the common name created in \*.xyzcorp63.com.

### 5.5.1.1.7 Configure Tomcat SSL Connector

At this stage, you have an SSL Certificate, and need to configure Tomcat to use it when sending/receiving traffic over SSL/HTTPS. Update the `server.xml` file to specify the name and location of your `p12` file along with the keystore password you supplied when creating the `p12` file.

Edit the file `/var/www/tomcat/conf/server.xml` and add the following connector definition

```
[root@ip-172-31-4-186 cert]# vi /var/www/tomcat/conf/server.xml
```

Item 19

```
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
    SSLEnabled="true"
    scheme="https"
    secure="true"
    clientAuth="false"
    sslProtocol="TLS"

    maxHttpHeaderSize="8192"
    maxThreads="150"
    minSpareThreads="25"
    enableLookups="false"
    disableUploadTimeout="true"
    acceptCount="100"
    useBodyEncodingForURI="true"

    keystoreType="pkcs12"
    keystoreFile="/var/dfactor/cert/self_sign_certificate.p12"
    keystorePass="mypassword" />
```

Item 20

The important thing to specify is that `keystoreType="pkcs12"`, the `keystorePass` is the export password you gave when generating `pkcs12` file in the earlier section, and the `keystoreFile` is the path to the file.

### 5.5.1.2 CA Signed certificate



If you are using Self Signed Certificate, this section can be skipped.

If you have a CA Signed Certificate and in the format of a keystore file, it should simply be a matter of uploading the keystore file, e.g. myKeystore to /var/www/tomcat/conf and enter the relevant password in the /var/www/tomcat/conf/server.xml file shown below. Ensure also the keystoreType is JKS

#### 5.5.1.2.1 Configure Tomcat SSL Connector

```
<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"
  SSLEnabled="true"
  scheme="https"
  secure="true"
  clientAuth="false"
  sslProtocol="TLS"

  maxHttpHeaderSize="8192"
  maxThreads="150"
  minSpareThreads="25"
  enableLookups="false"
  disableUploadTimeout="true"
  acceptCount="100"
  useBodyEncodingForURI="true"

  keystoreType="JKS"
  keystoreFile="conf/myKeystore"
  keystorePass="mypassword" />
```

Item 21

## 5.6 Change MySQL Password

**IMPORTANT:** If you need to change MySQL passwords once KeyScaler is installed and running, please contact [customer\\_support@deviceauthority.com](mailto:customer_support@deviceauthority.com) for assistance as the following instructions are only applicable prior to installing KeyScaler.

To change the default passwords used for MySQL, use the following commands, supplying your new passwords as indicated. The default root password for MySQL is admin.  
**REMEMBER** these passwords as you'll be using them to install and administer KeyScaler.



Please note: A default password 'mypassword' and 'changeit' is used for the purpose of readability of this document. These passwords must, however, be updated in line with your company's security policy.

```
[root@ip-172-31-4-186 cert]# mysql -u root -p
Enter password: admin
mysql> use mysql;
mysql> update user set password=PASSWORD('mypassword') where User='root';
mysql> update user set password=PASSWORD('mypassword') where User='dfactor_user';
mysql> flush privileges;
mysql> quit
```

*Item 22*

Then, verify access by logging in using both users and their new passwords.

```
[root@test ]# mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.5.30 MySQL Community Server (GPL)
...
mysql> quit

[root@test ]# mysql -u dfactor_user -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
...
mysql> quit
```

*Item 23*



## 5.7 Start the Tomcat Webserver

As root, start the service (called dfactor)

```
[root@test ]# service dfactor start
Starting DeviceAuthority D-Factor
Using DFACTOR_HOME: /var/dfactor
Using IDP_HOME: /var/dfactor/idp
Using CATALINA_BASE: /var/www/tomcat
Using CATALINA_HOME: /var/www/tomcat
Using CATALINA_TMPDIR: /var/www/tomcat/temp
Using JRE_HOME: /usr/java/latest
Using CLASSPATH: /var/www/tomcat/bin/bootstrap.jar:/var/www/tomcat/bin/tomcat-
juli.jar
Tomcat started.
```

*Item 24*

You can tail the Tomcat logs to verify the test page located in ROOT/index.html under /var/www/tomcat/webapps/ has deployed:

```
[root@test ~]# tail -f /var/www/tomcat/logs/catalina.out

2017-02-14 19:39:57,122 localhost-startStop-1 ERROR RollingFile contains an invalid element
or attribute "suppressExceptions"
14-Feb-2017 19:39:58.608 INFO [localhost-startStop-1]
org.apache.catalina.util.SessionIdGeneratorBase.createSecureRandom Creation of SecureRandom
instance for session ID generation using [SHA1PRNG] took [1,434] milliseconds.
log4j:WARN No appenders could be found for logger
(org.apache.tapestry5.ioc.RegistryBuilder).
log4j:WARN Please initialize the log4j system properly.
14-Feb-2017 19:39:59.489 INFO [localhost-startStop-1]
org.apache.catalina.startup.HostConfig.deployWAR Deployment of web application archive
/var/lib/dfactor-apache-tomcat-8.5.11/webapps/wizard.war has finished in 3,713 ms
14-Feb-2017 19:39:59.491 INFO [localhost-startStop-1]
org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory
/var/lib/dfactor-apache-tomcat-8.5.11/webapps/ROOT
14-Feb-2017 19:39:59.553 INFO [localhost-startStop-1]
org.apache.jasper.servlet.TldScanner.scanJars At least one JAR was scanned for TLDs yet
contained no TLDs. Enable debug logging for this logger for a complete list of JARs that
were scanned but no TLDs were found in them. Skipping unneeded JARs during scanning can
improve startup time and JSP compilation time.
14-Feb-2017 19:39:59.555 INFO [localhost-startStop-1]
org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application
directory /var/lib/dfactor-apache-tomcat-8.5.11/webapps/ROOT has finished in 64 ms
14-Feb-2017 19:39:59.562 INFO [main] org.apache.coyote.AbstractProtocol.start Starting
ProtocolHandler [http-nio-8080]
14-Feb-2017 19:39:59.567 INFO [main] org.apache.catalina.startup.Catalina.start Server
startup in 3829 ms
```

Item 25

## 5.8 Verify access to wizard pages

Before you can verify wizard page, you need to ensure the `wizard.xyzcorp63.com` domain is resolvable. Note the domain will be different for your installation. This can be done using an active DNS server or adding an entry to your local hosts file, e.g. as follows:

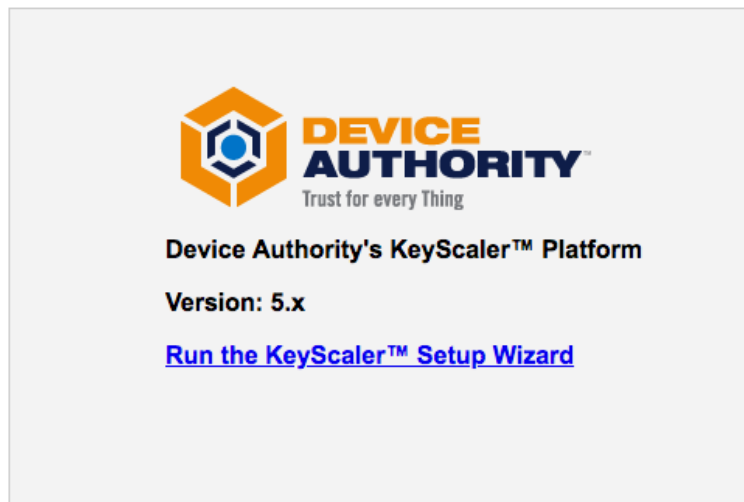
```
35.166.122.161 wizard.xyzcorp63.com
```

*Item 26*



If deployed in AWS you may need to configure security groups to allow for HTTP/S access.

Enter the following URL <https://wizard.xyzcorp63.com> in your browser and verify you can reach the wizard page.



*Item 27*

## 6 Next Steps

Install KeyScaler 6.2 by following the steps outlined in [1].