Installation Guide

Edge Intelligence

v7.1

September 2017

Preface

This guide provides information about how to install the Edge Intelligence software on virtual or bare-metal servers.

Overview

The installation of the Edge Intelligence software comprises of the following steps:

- Download and run the installer on a jump server.
- Running the installer on the jump server to install the catalog servers.
- Running the installer on the jump server to install each edge server as and when required.

These steps are performed using scripts which automatically install all of the software required, generate and install all security certificates and perform all of the necessary configuration operations. In addition, each script performs checks to ensure that the environment is as expected both before and after execution of the script.

The jump server provides a central command point for the remote installation and control of the software onto other servers. Conventionally, the jump server is entirely separate from the catalog servers - but can be hosted on a catalog server if preferred.

Linux Distributions

The installer and the Edge Intelligence software runs on any of the following Linux distributions:

- Ubuntu (v14.04)
- Debian7
- RedHat7
- Centos7

Note that the Linux distribution does not have to be the same on every server and can vary between the jump server, catalog servers and edge servers. Moreover, some or all of the servers may be hosted in the Cloud, on premise or mixed as required.

The install scripts will identify the operating system on each server and will install the corresponding package version.

Base Installation

A base installation of the platform consists of three catalog servers only without any edge servers or networks. These catalog servers provide the connection points for connecting, managing and accessing any networks that are subsequently created.

For production environments, exactly three catalog servers are required for resilience and high-availability; but for testing and evaluation purposes, you can install just one catalog server for simplicity.

Note that edge servers are installed as and when required when networks are being built or extended and only after the base installation has completed.

Tiers

Multiple tiers can be installed for the purposes of separating production and test environments or for the separation of departmental systems, for example; where each tier is effectively a completely separate installation from any other tiers.

Each tier is assigned a unique name and an operation performed from the jump box can name the tier to perform the operation on.

When there is only going to be a single tier, you can omit the tier name and the tier will be called 'default'.

Errors

If a part of the installation process fails a message will normally be raised on screen and in a log file and the process will terminate.

If it is not clear from an error message how to resolve the problem, please send the details in an email to support@edgeintelligence.com or raise a support request on the Edge Intelligence Knowledge Base.

Pre-Requisites

During the installation process, the jump server will connect to and install software on one or more remote target servers. The following is required before the installation begins:

- One of the supported Linux distributions on jump server and every target server.
- A common user account with sudo permission and a common sudo password on every target server.
- A public ssh key on the jump server with a corresponding private ssh key on every target server.
- Connectivity to all of the hosts from the jump server.

Note that an empty sudo user password is acceptable provided it is not set on all of the target servers.

Step 1 : Get the software distribution

Register an email address

If you have already registered your email address on the Edge Intelligence Knowledge Base then you can skip this registration step.

Go to the Edge Intelligence Knowledge Base.

Register your details by clicking on the Submit a Request link at the top right of the page.

Enter your email address with 'Registration' in both the subject and description boxes. Complete the captcha to verify you are not a robot and press the submit button.

You will receive an email with a link to verify your email address. Click the link in the email to complete the registration process.

Download the installer file

Go to the Edge Intelligence Knowledge Base.

Sign in using your registered email address and password.

Go to the Resources/Software section.

Go the Installer page.

Download the installer file.

Copy the downloaded installer file (`edgeinstall_<version>.run`) to your jump server.

Run the installer file

Log on to the jump server and run the installer file (`edgeinstall_<version>.run`).

```
bash:~$ sudo sh edgeinstall_<version>.run
```

⚠ Note, if an earlier version had been installed this step will overwrite packages previously installed.

The installation of the jump server is now complete and the jump server is ready to install the Edge Intelligence software on remote servers.

All of the scripts used throughout the remainder of the document can be found in the `edgeintelligence/bin` directory.

To simplify the execution of these scripts you can add `edgeintelligence/bin` to your path; or else you will need to cd to `edgeintelligence/bin` before you execute a script.

Step 2 : Define catalogue servers

Declare all of the catalog servers using the `es-define-catalogs` script giving a list of IP addresses for the target catalog servers.

```
bash:~$ ./es-define-catalogs 192.168.122.22 192.168.122.25 192.168.122.221
```

If you wish to name a tier for the installation, use the -t option with a tier name. By default the installation will be placed in the 'default' tier.

On running the `es-define-catalogs` script, you will be prompted to enter three distinct passwords.

1. The sudo user password. This is the password set for the sudo user on all of the servers.
2. A vault password. This is a password that is set, confirmed and used to access a vault for secure configuration information.
3. The SSL root certificate password.

⚠ The vault password is not stored anywhere and is required for subsequent installation and update operations. It is **VERY** important that this password is not forgotten!

⚠ The SSL root password is not stored anywhere. If forgotten, all server and client certificates will need to be re-generated

This step has recorded the IP address details and security certificate details for the catalog servers.

Step 3 : Request Licences

Collect catalog license information using the `es-request-license` script.

```
bash:~$ ./es-request-license -c
```

Like the other scripts, this script will prompt for the vault password.

The `es-request-license` script collects various machine credentials from each the catalog servers and stores them in a zip file.

The name of the zip file is in the form license-requests-yyyy-mm-dd-hh-mi-ss.zip and it's full path appears on screen when the es-request-license script is run.

Please email the checksum (also on screen) and attach the zip file to the Edge Intelligence support team at support@edgeintelligence.com

The license keys will be returned to you in another zip file with name of the form license-yyyy-mm-dd-hh-mi-ss.zip

Upload this new zip file to your jump server and proceed to the next step.

Step 4 : Install Catalogs

To install the Edge Intelligence software on all of the catalog servers, use the `es-install-catalogs` script:

```
bash:~$ ./es-install-catalogs -f
$PWD/licenses-2017-09-15-15-38-27.zip
```

The -f option is required and specifies the full path of the license zip you uploaded in the previous step.

Like the other scripts, this script will prompt for the vault password.

The installation on the catalog servers is now complete.

Installing Edge Servers

As and when you create or extend networks, you will need to install the Edge Intelligence software on one or more edge servers. It is possible to provision a single server or many servers at the same time.

This can be done using the `es-install-node-server` script.

```
bash:~$ ./es-install-node-server -f
$PWD/licenses-2017-09-15-15-38-27.zip 192.168.122.249 192.168.78.78
```

The -f option is required and specifies the full path of the license zip you uploaded in step 3 above.

Note that when you first run the `es-install-node-server` script, you will be prompted to enter the sudo user password.

Like the other scripts, this script will also prompt for the vault password.

When this script has completed, each edge server targeted will be ready to host a new instance node added to a network.

Optional Steps

The following sections contain optional steps such as product uninstall and re-generation of SSL certificates.

Stop/Start Servers

You may wish to stop a server prior to decommissioning it or restart/start a server following a hardware failure.

Servers may be stopped and (re)started collectively or individually using the `es-edgistore-ctl` script.

For example, to stop all catalog and node servers:

```
bash:~$ ./es-edgistore-ctl -a stop
```

For example, to restart one specific edge server:

```
bash:~$ ./es-edgistore-ctl -H 192.168.122.25 restart
```

Uninstall servers

To uninstall Edge Intelligence software use the `es-uninstall-server` script.

For example, to uninstall all catalog and node servers:

```
bash:~$ ./es-uninstall-server -t MyTier -a
```

For example, to uninstall one edge server:

```
bash:~$ ./es-uninstall -t MyTier -H 192.168.122.25 restart
```

⚠ If you uninstall a catalog server, you will lose access to any networks and data hosted on that server.

⚠ If you uninstall an edge server, you will lose any data hosted on that server.

Generate Client SSL Certificates

If it is necessary generate client certificates to allow a user to connect to one or more catalog servers use the `es-generate-client-cert` script.

For example, create client certificate files for the user 'scott'.

```
bash:~$ ./es-generate-client-cert scott
```

This creates a zip file of the form `ssl-scott.zip` containing the necessary certificate and key files to enable ssl connection from query clients.

Each application will have its own requirements for importing the certificates. The Appendices describe how to enable ssl connections from the *Tableau Desktop.*

Appendix 1 : edgeinstall script

To see all available .run file options:

```
bash:~$ sudo sh edgeinstall_<version>.run help
```

```
*** ./install.sh: [ install | clean | uninstall | purge | help ] ***
***    install (default) - install this product ***
***
***    uninstall         - uninstall this product ***
***    purge             - uninstall this product and delete the data
directory (debian/ubuntu only) ***
***    help              - this message ***
```

⚠ Note, the uninstall option removes only the installation framework ie the edgeintelligence folder, the purge option removes the installation and network configurations ie both `edgeintelligence` and .`edgeintelligence` folders

Appendix 2 : es-define-catalogs script

```
Usage: ./es-define-catalogs [options] HOST_OR_IP [HOST_OR_IP ...]
```

Option	Long format	Description
-d	--debug	Enable debugging
	--force-num-catalogs	Force through with not exactly 3 catalog servers
-h	--help	Show this help
-i [IP ADDRESS] [...]	--ip [IP ADDRESS] [...]	Specify IP address for a matching host in the host list (default use hostname)
-l [LOCALE]	--locale [LOCALE]	Specify the locale to use (default host locale)
	--no-ssl	Set up without SSL configuration
-p [PRIVATE_KEY_FILE]	--private-key PRIVATE_KEY_FILE] [...]	Specify private SSH key file for a matching host in the host list. Specify - to use the default (default ssh default)
-H (HOST]	--host [HOST]	Apply changes to this host only
	--serial	Apply changes to catalogs serially
-t [TIER]	--tier [TIER]	Specify the tier e.g. development, test, live (default 'default')
-u [USER]	--user [USER]	Use an alternative login name for installation (default 'matts')
	--vault-password-file	Use the file supplied for the vault password
-v	--verbose	Run in verbose mode

For example, to see all the options available when defining your catalogues:

```
bash:~$ ./es-define-catalogs -h
```

⚠ Note, all options must be specified on the command line BEFORE the HOST_OR_IP space-separated string. DNS hostnames may be used. Alternatively you can assign arbitrary names to the HOST_OR_IPs providing there are corresponding -i flags to determine the actual host to connect to.

For example:

```
-bash-$ ./es-define-catalogs -i 192.168.122.22 -i 192.168.122.25 -i
192.168.122.221 host1 host2 host3
```

Appendix 3 : es-request-license script

```
Usage: ./es-request-license [options]
```

Option	Long format	Description
-c	--catalogs --catalogues	Operate on catalog servers
-d	--debug	Enable debugging
-f	--file	Specify zip file containing licenses
-h	--help	Show this help
-H (HOST]	--host [HOST]	Apply changes to this host only
	--serial	Apply changes to catalogs serially
-t [TIER]	--tier [TIER]	Specify the tier e.g. development, test, live (default 'default')
	--vault-password-file	Use the file supplied for the vault password
-v	--verbose	Run in verbose mode

For example, to see all options available when requesting licences:

```
bash:~$ ./es-request-license -h
```

Appendix 4 : es-install-catalogs script

```
Usage: es-install-catalogs [options]
```

Option	Long format	Description
-d	--debug	Enable debugging
-f	--file	Specify zip file containing licenses
-h	--help	Show this help
	--serial	Apply changes to catalogs serially
-t [TIER]	--tier [TIER]	Specify the tier e.g. development, test, live (default 'default')
	--vault-password-file	Use the file supplied for the vault password
-v	--verbose	Run in verbose mode

For example, to see all options available when installing catalogues:

```
bash:~$ ./es-install-catalogs -h
```

Appendix 5 : es-install-node-server script

```
Usage: es-install-node-server [options] HOST_OR_IP [HOST_OR_IP ...]
```

Option	Long format	Description
-d	--debug	Enable debugging
-f	--file	Specify zip file containing licenses
-h	--help	Show this help
-i [IP ADDRESS] [...]	--ip [IP ADDRESS] [...]	Specify IP address for a matching host in the host list (default use hostname)
-p [PRIVATE_KEY_FILE]	--private-key PRIVATE_KEY_FILE] [...]	Specify private SSH key file for a matching host in the host list. Specify - to use the default (default ssh default)
	--serial	Apply changes to catalogs serially
-t [TIER]	--tier [TIER]	Specify the tier e.g. development, test, live (default 'default')
	--vault-password-file	Use the file supplied for the vault password
-v	--verbose	Run in verbose mode

For example, to see all options available when installing node servers:

```
bash:~$ ./es-install-node-servers -h
```

Appendix 6 : es-edgistore-ctl script

```
Usage: es-edgistore-ctl [options] [start|stop|reload|restart|status
```

Option	Long format	Description
-a	--all	Operate on all servers
-c	--catalogs --catalogues	Operate on catalog servers
-d	--debug	Enable debugging
-h	--help	Show this help
-H (HOST]	--host [HOST]	Apply changes to this host only
-n	--nodes	Operate on the node servers
	--serial	Apply changes to catalogs serially
-t [TIER]	--tier [TIER]	Specify the tier e.g. development, test, live (default 'default')
	--vault-password-file	Use the file supplied for the vault password
-v	--verbose	Run in verbose mode

For example, to see all the options available to the es-edgistore-ctl script:

```
bash:~$ ./es-edgistore-ctl -h
```

Appendix 7 : es-uninstall-server script

```
Usage: es-uninstall-server-usage [options]
```

Option	Long format	Description
-a	--all	Operate on all servers
-c	--catalogs --catalogues	Operate on catalog servers
-d	--debug	Enable debugging
-h	--help	Show this help
-H (HOST]	--host [HOST]	Apply changes to this host only
-n	--nodes	Operate on the node servers
	--no-purge	Don't purge the package (data/configuration)
-r	--remove	Also remove the specified hosts from the appropriate inventory files
	--serial	Apply changes to catalogs serially
-t [TIER]	--tier [TIER]	Specify the tier e.g. development, test, live (default 'default')
	--vault-password-file	Use the file supplied for the vault password
-v	--verbose	Run in verbose mode

For example, to see all the options available to the es-generate-client-cert script:

```
bash:~$ ./es-uninstall-server-usage -h
```

Appendix 8 :
es-generate-client-cert script

```
Usage: bin/es-generate-client-cert [options] CERTIFICATE_NAME
```

Option	Long format	Description
-d	--debug	Enable debugging
-h	--help	Show this help
-t [TIER]	--tier [TIER]	Specify the tier e.g. development, test, live (default 'default')
	--vault-password-file	Use the file supplied for the vault password
-v	--verbose	Run in verbose mode

For example, to see all the options available to the es-generate-client-cert script:

```
bash:~$ ./es-generate-client-cert -h
```

Appendix 9 : Jump server directories

The installer creates two new directories under the current working directory:

The `edgeintelligence` folder contains the scripts for installing software on remote servers.

The `.edgeintelligence` folder contains the latest packages and custom configuration files.

Appendix 10 : AWS configuration

To permit and persist interconnectivity between servers in a tier created on the AWS platform, some additional configuration needs to be carried out in the AWS console:

1. Elastic IP addresses need to be created for ALL servers in the cluster (this ensures the public ip address of the aws instance remains constant i.e. doesn’t change when the instance is rebooted)

2. Security Group Inbound rules on the Catalogue Servers need to have ALL node IP addresses added with the standard postgresql port of 5432 in the form 0.0.0.0/32

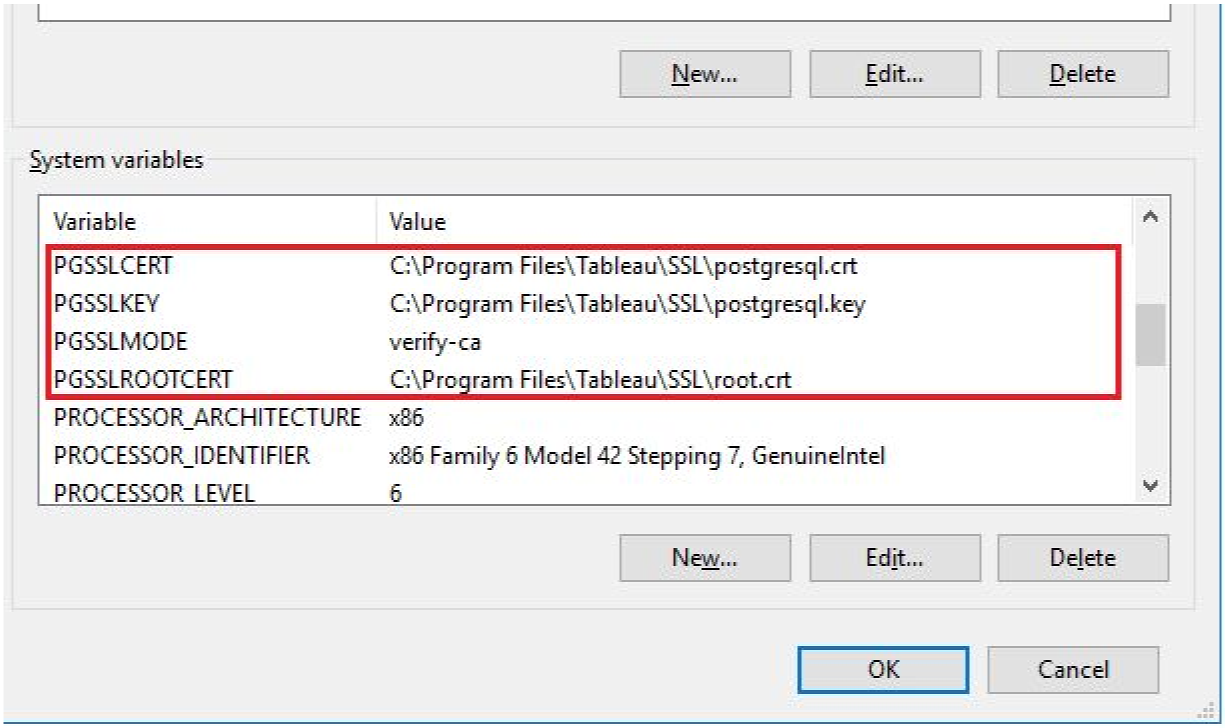
3. Security Group Inbound rules on the Catalogue Servers need to have a Custom TCP rule with protocol=TCP and port=1138 for all addresses ie. 0.0.0.0/0

4. Security Group Inbound rules on the Instance Node Servers need to have ALL Catalog IP addresses AND its own elastic IP address added

Appendix 11 : Tableau Desktop Connectivity

To allow Tableau to make an SSL connection to a Catalog server using the PostgreSQL data source, some windows environment variables need be set.

1. From the Desktop, right-click the very bottom left corner of the screen to get the Power User Task Menu.
2. From the Power User Task Menu, click **System**.
3. Click the **Advanced System Settings** link in the left column.
4. In the System Properties window, click on the **Advanced** tab, then click the **Environment Variables** button near the bottom of that tab.
5. In the Environment Variables window, click the **New** button in the **System variables** section and add the specific variable names (outlined below) with the paths to your unzipped ssl file archive generated from the `es-generate-client-cert` script.



Variable	Value
PGSSLCERT	C:\Program Files\Tableau\SSL\postgresql.crt
PGSSLKEY	C:\Program Files\Tableau\SSL\postgresql.key
PGSSLMODE	verify-ca
PGSSLROOTCERT	C:\Program Files\Tableau\SSL\root.crt
PROCESSOR_ARCHITECTURE	x86
PROCESSOR_IDENTIFIER	x86 Family 6 Model 42 Stepping 7, GenuineIntel
PROCESSOR_LEVEL	6

Appendix 12 : Troubleshooting guide

The bin/es-* installation scripts use ansible to perform various tasks across one or more servers.

When a script encounters a problem (or potential problem) ansible outputs an error (or warning) message to the screen. The root cause of the problem is normally apparent from the message received, however it may not be obvious how to fix the problem,

The following are messages that may be encountered when running the installation scripts, and suggested remedies for these problems.

Script	Message	Problem description	Remedy

es-define-catalogs	host1 | UNREACHABLE! => {
	"changed": false,
	"msg": "Failed to connect to the host via ssh: ssh: Could not resolve hostname host1: Name or service not known\r\n",
	"unreachable": true
	}
		ssh connection to the specified host has failed.
		Could be the hostname or ip_address is not valid, the host is offline, or ssh key is missing or wrong.
		Could also be a firewall issue preventing communication between the host and localhost
			Check hostname / ip_address are correctly defined, ssh key exists and is correct, and firewall is not preventing access.
			Following resolution to any of the above, undo initial catalog definition by running es-uninstall-servers -c -t <TIER>
			Then retry es-define-catalogs

Host unreachable during cat install
Host unreachable during server node install
Host unreachable during cat uninstall
Host unreachable during server node uninstall
Edgeinstall purge run before servers uninstalled

Can’t find a solution to your problem? Please send details to support@edgeintelligence.com