Ubisense Installation Guide

for SmartSpace

Version 3.8.1

Part Number: SS_INS_3.8.1_EN

Contents

Ubisense Installation Guide for SmartSpace

A SmartSpace installation comprises the core Ubisense software required to run SmartSpace along with any further software for additional SmartSpace features you have licensed.

These instructions guide you through installing Ubisense SmartSpace software. They include a description of the organization and installation of Ubisense software across server, admin and client machines; prerequisites for installation; and the stages of the installation process.

SmartSpace installations can range in size from a single computer running all SmartSpace software, for example for evaluation purposes, to several servers with associated admin and client machines in a large industrial setting. SmartSpace can be installed and run on Windows and Linux computers and an installation may include a mix of these, for example with Linux servers and Windows client machines, and the instructions that follow include information for both Linux and Windows.

For Windows computers, installation files (.msi) are supplied. However, because of the variations between Linux distributions and between package management systems used by different Enterprise configurations, Ubisense do not provide an automatic installation method for Linux. Instead the instructions describe the necessary state of a Ubisense platform installation on Linux, prerequisites for the installation, the layout and permissions expected, and example scripts.

> ⚠️ **From SmartSpace version 3.8, installed Ubisense applications are 64 bit. For interoperability between 32-bit and 64-bit applications, you require the new 64-bit Ubisense Application Manager. Using an older 32-bit version of the application will prevent you from downloading other 64-bit Ubisense applications, for example SmartSpace Config.**

The next section provides an overview of the installation process and subsequent sections take you through installing SmartSpace step by step.

The Ubisense Installation Process

Ubisense requires software to be installed on three types of machine: *server*, *admin* and *client*.

- Servers run the core and controller software, and the Ubisense platform from which you can start and stop the Ubisense servers. Ubisense servers can run on either Windows or Linux (see *Requirements* for supported versions).

- The software installed on an admin machine enables you to manage the installation and deployment of SmartSpace features across your entire SmartSpace installation.

- On client machines, you will find the SmartSpace applications available to users according to the SmartSpace features you have licensed.

Depending on your requirements, you might install all three SmartSpace machines on a single computer or you might spread the installation over a number of machines.

The following are the steps required to set up a new installation:

1. Install licenses.

2. Install the Ubisense server software onto the relevant machine(s).

3. Start the core server and service controllers.

4. Install the Ubisense admin software onto the relevant machines.

5. Install and deploy licensed SmartSpace features.

6. Install the Ubisense client software onto the relevant machines.

7. Download SmartSpace software to client machines.

Depending on the features you have licensed, users may access SmartSpace via a web interface, for example to view Web maps. In this case additional steps are required to set up a web server: these are described in *Ubisense Installation Guide for SmartSpace Web* from https://docs.ubisense.com/portal/Content/Downloads.htm. You do not need to install additional software for end users to access the browser-based features.

Installation instructions for Windows start at *Installing SmartSpace on Windows* and installation instructions for Linux start at *Installing SmartSpace on Linux*.

Requirements

This section describes the hardware and software prerequisites for a SmartSpace installation.

Server Hardware Requirements

Exact requirements for server hardware will depend on such things as the number of sensors and tags in your installation or the number of users querying any browser-based applications you have licensed. Contact Ubisense for further guidance on the specific requirements for your installation.

The following is an illustration of an installation with two servers running DIMENSION4 and SmartSpace with the Visibility component.

Feature	Server 1 (DIMENSION4 + SmartSpace core)	Server 2 (Windows server server running IIS)
Processor	Quad-core Intel® Xeon® processors 3400 series	16-core Intel® Xeon® processors
Memory	8 GB	8 GB
Ethernet Interface	Gigabit Network Adapter	Gigabit Network Adapter
Virtualization	For information about virtualization, contact Ubisense Support.	For information about virtualization, contact Ubisense Support.

Server Software Requirements

Ubisense supports the following operating systems:

Windows

- All versions of Windows server currently supported by Microsoft from Windows Server 2012 onwards.

- Additionally, for client machines (and for proofs of concept, with the agreement of Ubisense Support), versions of Windows from Windows 8.0 which are currently supported by Microsoft.

Note: "Home" versions of Windows, e.g. Windows 10 Home, are *not* suitable for use as SmartSpace *servers* in production environments, although they may be acceptable for evaluation purposes.

Linux

- SUSE Linux Enterprise Server (SLES) 12 SP2

- Red Hat® Enterprise Linux® (RHEL) 8+

If you are installing a web server for use with the Visibility component, see the additional considerations discussed in *Linux Requirements for SmartSpace Web* in the Ubisense Installation Guide for SmartSpace Web.

- For Windows, you *must* install the latest ASP.NET Core Runtime 6.0.x Hosting bundle.

- For Linux, we *strongly recommend* you install the latest ASP.NET Core Runtime 6.0.x.

For a list of SmartSpace releases and their respective .NET versions, see the Ubisense Documentation Portal.

Microsoft Visual C++

With the release of SmartSpace 3.8, Windows and Linux 32-bit executables were replaced with 64-bit executables. If you are installing or upgrading to SmartSpace 3.8 or higher, you need to ensure you have the latest C++ runtime dlls by installing the Visual C++ Redistributable Package from *https://aka.ms/vs/17/release/vc_redist.x64.exe*.

Typed API modules (.NET API)

If you built Typed API modules with versions of SmartSpace earlier than 3.8, you need to rebuild these modules using x64.

C++ API

If you have been using a version of the C++ API earlier than SmartSpace 3.8, you must upgrade to a 64-bit version of Microsoft Visual Studio 2015 or later.

ManagedBrowser (Application .NET API)

If you have ManagedBrowser clients /services that you want to rebuild against the latest SmartSpace release (not required), you will have to switch to x64.

Database Server for Reporting or RDBMS map

Either of the following are required *only* if the Reporting component or the RDBMS map feature is licensed:

SQL Server

Database versions: 2008 R2 or higher.

- Windows servers using ODBC and SQL Server Native Client library

- Linux servers using Microsoft ODBC Driver 17 for SQL Server

Oracle

Database versions: 11G R2 or higher.

- Windows servers using Oracle Instant Client 21.x library

- Linux servers using Oracle Instant Client 21.x library

For information on configuring servers for use with the Reporting component, see SmartSpace Reporting on the Ubisense Documentation Portal.

For information on configuring servers for use with the RDBMS map feature, see RDBMS map configuration on the Ubisense Documentation Portal.

Additional Requirements for Linux Installations

The following are requirements for Linux computers onto which server or admin machines are to be installed:

- Platform executables require:

  - a 64-bit libstdc++.so.6

  - a 64-bit libssl.so.1.1

- The firewall should be disabled on the server

- In order to work around kernel bind(0) behavior, the local dynamicport range should be changed

  - **Either**: place the following in an init script such as /etc/rc.d/rc.local: sysctl -w net.ipv4.ip_local_port_range=32768 49978

  - **Or**: place the following in /etc/sysctl.conf: net.ipv4.ip_local_port_range=32768 49978

  After reboot or applying sysctl -p, the property net.ipv4.ip_local_port_range can be checked with sysctl -a

Platform user

A user should be configured to execute platform services. We will refer to this as the *platform user*.

Operations group

A group should be configured for operations. Users in this group should be able to perform production operations, including starting and stopping the platform services, making and restoring backups, and performing other diagnostic and support roles, such as license management and platform service upgrades. The platform user might be in the operations group.

Windows client machine

In order to run the SmartSpace Config software, you will need access to a Windows computer to install a client machine.

Unzipping Software to a Distribution Directory

The SmartSpace software is supplied as a zipfile with the name SmartSpace followed by numbers indicating the version of the software, for example **SmartSpace_3_3_669.zip**. Before you install SmartSpace, you need to unzip this file into a *distribution directory* accessible to the machines on which you will be installing the software.

Installing SmartSpace on Windows

Installing Licenses on Windows

SmartSpace feature licenses are supplied as a zipfile with the name **FeatureSetup.zip**. Before you install the licenses, you need to unzip this file into a directory accessible to a server machine from which **ubisense_core_server** will be executed.

To install SmartSpace licenses:

1. Go to the directory where you unzipped the licenses.

2. Double-click the **FeatureSetup.msi** file and the Ubisense Feature Licenses Setup Wizard appears.

3. Click **Next** and the Ubisense Feature Licenses Setup wizard appears.

4. By default all licenses are selected for installation to the default location **C:\Program Files (x86)\Ubisense 3\bin**.

   - Click on the directory tree of licenses, click on individual features and choose whether or not they are to be installed

   - Click **Reset** to return the licenses selection to its default setting

   - Click **Browse** to navigate to a different directory to install the licenses in

5. When you have selected the files and location you require, click Next and then click **Install**.

6. When installation is complete, click **Finish** to close the wizard.

Installing the Server Software on Windows

Installing the Server Software

On each machine you want to use as a server, you must install the Ubisense server software.

During the installation process, for each server machine you can choose to install either the core server or service controller or both. If you intend to run SmartSpace on a single server, you need to install both the core server and service controller on that machine. For an installation with more than one server, you need to run the core server on one machine only and the service controller on the rest, and you can install the components accordingly.

To install the server software:

1. Go to the **servers\windows** directory of your SmartSpace distribution directory.

2. Double-click the **UbisenseServers.msi** file and the Ubisense Servers Setup wizard appears.

3. Click **Next** to display the Custom Setup dialog.
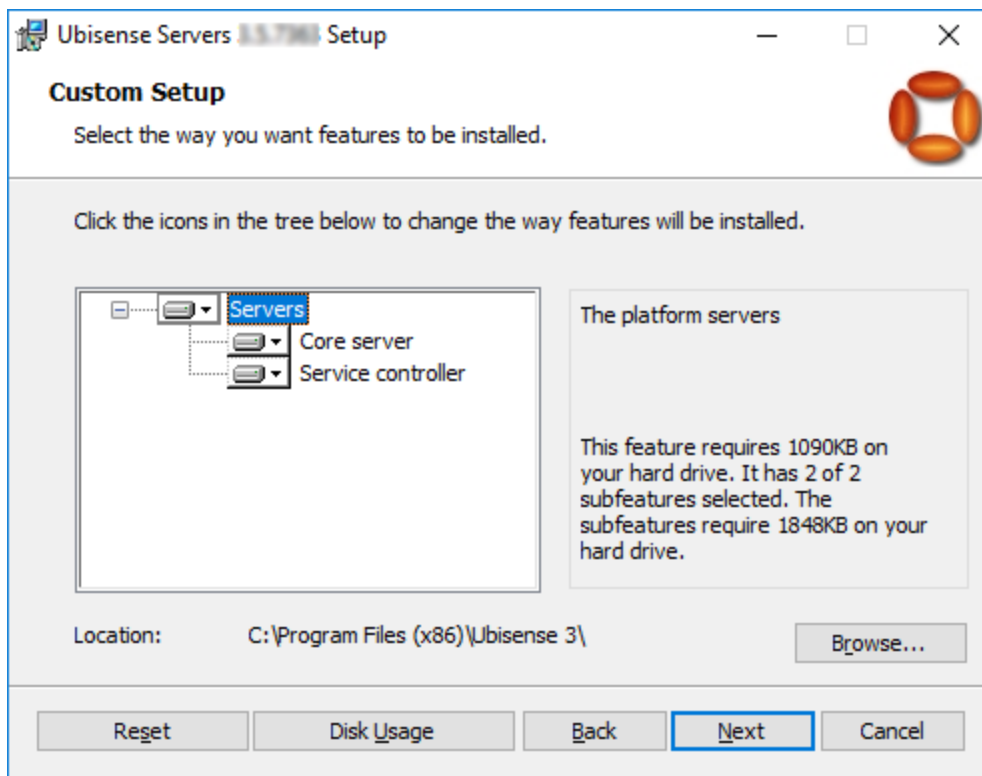
4. Choose the components to install.

By default, all features are selected. Choose whether to install or exclude items using the dropdowns beside their names. **Reset** returns you to the default selection.

5. Choose the location for the software.

   You can accept the default **C:\Program Files (x86)\Ubisense 3** or click **Browse** to select another destination.

6. Click **Next** and click **Install**.

7. When the installation is complete, click **Finish** to close the Ubisense Servers Setup wizard.

You have now installed Ubisense Platform Control and the Ubisense server software onto your computer. Using Platform Control to start the server software is described in the next section.

Starting the Server Software

After you have installed the server software, you need to start the core server and service controller(s).

To start the server software:

1. From a SmartSpace server, run Platform Control.

2. For a new installation, you need to choose a location for your dataset:

   In the **Properties** section, browse to the required location (creating a new folder, if needed) and click **OK (new)**.

3. Start the core server and service controller by:

   a. Selecting **UbisenseCoreServer 3** and then clicking **Start**.

   b. Selecting **UbisenseServiceController 3** and then clicking **Start**.

      See the information below for information on starting services with a single server or multiple servers.

      You are offered only the server components installed on the machine (see *Installing the Server Software*).

   The status of each service changes to **to be started**.

4. Click **Apply**. The status of each service changes to **running**.

Running SmartSpace on a Single Server

If you want to configure SmartSpace to run on a single server, run Platform Control on the server and:

- In Services, ensure you have started *both* the core server and the service controller.

- *Don't* select **run in standalone mode** if you want access to your network (and to sensors).

Running SmartSpace on Multiple Servers

If you want to configure SmartSpace to run on more than one server, you must:

- Assign one server to be the core server and *on this machine only* run Platform Control and in Services start the core server.

- On all other server machines, run Platform Control and in Services start the service controller.

- *Don't* select **run in standalone mode** if you want access to your network (and to sensors).

Running SmartSpace in Standalone Mode

If you want to configure SmartSpace to run on a single server with no network communication (and no access to sensors), run Platform Control on the server and:

- In Properties, check **run in standalone mode**

- In Services, ensure you have started *both* the core server and the service controller.

Backing up your Dataset

After you have set up your SmartSpace installation, ensure that you back up your dataset occasionally, so that you can recover your data. To take a backup, use the **Backup Dataset** option, and then compress the folder.

You can also use the **ubisense_backup.exe** command-line tool from the **tools\windows** folder of your distribution directory to backup your dataset.

Installing Admin Machines on Windows

To install the Ubisense software for an admin machine:

1. Go to the **clients\windows** directory of your SmartSpace distribution directory.

2. Double-click the **UbisenseServiceManager.msi** file and the Service Manager Setup wizard appears.

3. Click **Next**.

4. Choose the Destination Folder for the software.
    You can accept the default **C:\Program Files (x86)\Ubisense 3** or change to another destination.

5. Click **Next** and click **Install**.

6. When the installation is complete, click **Finish** to close the Service Manager Setup wizard.

You have now installed Service Manager onto your computer and you can use it to install and deploy SmartSpace features.

> ℹ️ Before you can install and deploy features, you must install their licenses.

1. From an admin machine, run Service Manager 3.

2. Click **Install services...**.

3. Specify the directory from which to install.

    This is generally the **packages** folder in your SmartSpace distribution directory. Click **<Recently used directories>** to select previous locations of features.

4. Select the features you want to install.

    Use **Select all** or **Clear all** or click on individual features to indicate which items you want to install.

    - All SmartSpace features are listed.

    - All licensed features are selected by default.

    - Unlicensed features are shown preceded by **[Unlicensed]**. You cannot select these features.

    As a minimum, you should select SmartSpace core.

5. By default **Deploy services** is selected.

This means that any services you select for installation will also be automatically deployed during the installation process. Deselect **Deploy services** if you want to manually deploy the services after installation.

6. Click **Install**.

7. When installation is complete, click **Finish**.

You have now installed your SmartSpace features. In Service Manager you can see which services have been deployed by the installed features.

Installing Client Machines on Windows

In Windows, the Ubisense Application Manager allows you to perform the following configuration activities on a client machine:

- Set up Start menu shortcuts for client applications
- Download various command-line tools and SmartSpace documents to a specified location on a client machine

To install the Ubisense software for a client machine:

1. Go to the **clients\windows** directory of your SmartSpace distribution directory.
2. Double-click the **UbisenseApplicationManager.msi** file and the Ubisense Application Manager Setup wizard appears.
3. Click **Next**.
4. Choose the Destination Folder for the software.
   You can accept the default **C:\Program Files (x86)\Ubisense 3** or change to another destination.
5. Click **Next** and click **Install**.
6. When the installation is complete, click **Finish** to close the Ubisense Application Manager Updater Setup wizard.

You have now installed the Ubisense Application Manager and can now configure shortcuts to client applications and download documents and other files to your client machine.

Managing applications

To create shortcuts to SmartSpace applications:

1. Run the Ubisense Application Manager and click on **APPLICATIONS**.
2. Available applications are listed, with their version numbers and, where applicable, location on the Start menu.

   Choose the applications you want to install.

   - Double-click a single application
   - Select several applications and press Enter

   The following SmartSpace client program is available:

- SmartSpace Config (the main SmartSpace configuration GUI)

3. Click **Create shortcuts for selected applications**.

   Shortcuts are created in the Start menu in the locations indicated.

Managing tools and documents

To download SmartSpace command-line tools and documents to a selected directory:

1. Run the Ubisense Application Manager and click on **DOWNLOADABLES**.
   Command-line tools and documents are listed in different categories. The tools and
   documents available to you depend on the features you have installed.

2. Choose the tools or documents you want to download.

3. Specify the directory to install the files in and click **Start download**.

   The files are downloaded to the specified directory.

> ⓘ Whenever you upgrade your SmartSpace installation, you must follow the process
> described above to replace your existing tools and documents with upgraded
> versions.

Installing SmartSpace on Linux

Installing Licenses on Linux

License files must be placed on the server so that the platform can find them. The default location is in the directory **/etc/ubisense**. If a different location is required, then the **license_search_path** can be defined in **platform.conf** (see *Configuration Parameters* for information on the location of this file). Each program also searches for licenses in the same directory as its executable. Licenses should be readable by both the platform user and by the operations group.

Installing the Server Software on Linux

Installing the Server Software

For Linux servers, there are two executables: **ubisense_core_server** and **ubisense_local_control**. You can find them in the following locations in the distribution directory:

```
servers/linux/ubisense_core_server
servers/linux/ubisense_local_control
```

If you want to run SmartSpace on a single server, copy both of these files to that machine.

If you want to run SmartSpace on several servers, copy **ubisense_core_server** onto one server machine only and **ubisense_local_control** onto the remainder of the machines.

Starting the Server Software

On each server machine, one or both of the **ubisense_core_server** and **ubisense_local_control** services should be executed on startup, depending on whether the machine is to act as a core server, a service controller, or both. These services should be executable by the platform user, and no other user, to avoid accidental execution. Because of the variations between Linux distributions, Ubisense do not ship standard startup scripts for these executables, but examples are provided:

Sample init.d scripts for core server and service controller

Core Server

```
#!/bin/bash
#
# Init file for Ubisense core platform server
#
# chkconfig: 345 98 02
# description: Ubisense core platform for linux
# processname: ubisense_core_server
# config: /etc/ubisense.conf

# source function library
```

```
if [ -e /etc/rc.d/init.d/functions ]
then
    . /etc/rc.d/init.d/functions
else
 # steal status() from /etc/rc.d/init.d/functions on a RH box
 status() {
        local base=${1##*/}
        local pid

        # Test syntax.
        if [ "$#" = 0 ] ; then
                echo $"Usage: status {program}"
                return 1
        fi

        # First try "pidof"
        pid=`pidof -o $$ -o $PPID -o %PPID -x $1 || \
            pidof -o $$ -o $PPID -o %PPID -x ${base}`
        if [ -n "$pid" ]; then
                echo $"${base} (pid $pid) is running..."
                return 0
        fi

        # Next try "/var/run/*.pid" files
        if [ -f /var/run/${base}.pid ] ; then
                read pid < /var/run/${base}.pid
                if [ -n "$pid" ]; then
                        echo $"${base} dead but pid file exists"
                        return 1
                fi
        fi
        # See if /var/lock/subsys/${base} exists
        if [ -f /var/lock/subsys/${base} ]; then
                echo $"${base} dead but subsys locked"
                return 2
        fi
        echo $"${base} is stopped"
        return 3
 }
fi
```

```
# pull in sysconfig settings
[ -f /etc/ubisense.conf ] && . /etc/ubisense.conf

PLATFORM_USER=${PLATFORM_USER:-platform}
UBISENSE_CORE_SERVER=/home/platform/bin/i586_linux_2.6/ubisense_core_server
export UCONFIG=/etc/ubisense/platform.conf

RETVAL=0
prog="ubisense"

start()
{
        echo -n $"Starting ubisense_core_server:"
        if [ -e /etc/rc.d/init.d/functions ]
        then
           daemon --check ubisense_core_server --user=platform ${UBISENSE_CORE_SERVER} -d
        else
           startproc -u platform ${UBISENSE_CORE_SERVER} -d
        fi
         touch /var/lock/subsys/ubisense_core_server
        echo
}

stop()
{
        echo -n $"Stopping ubisense_core_server:"
         if [ -e /etc/rc.d/init.d/functions ]
        then
             killproc ubisense_core_server
        else
             killproc ${UBISENSE_CORE_SERVER}
        fi
         rm -f /var/lock/subsys/ubisense_core_server
        echo
}


case "$1" in
        start)
                start
                ;;
        stop)
                stop
                ;;
        restart)
                 stop
                start
                ;;
        status)
                 status ubisense_core_server
                ;;
        *)
                echo $"Usage: $0 {start|stop|restart|status}"
```

```
                RETVAL=1
esac
exit $RETVAL
```

Local Controller

```bash
#!/bin/bash
#
# Init file for Ubisense local controller
#
# chkconfig: 345 99 01
# description: Ubisense local controller for linux
# processname: ubisense_local_control
# config: /etc/ubisense.conf

# source function library
if [ -e /etc/rc.d/init.d/functions ]
then
    . /etc/rc.d/init.d/functions
else
 # steal status() from /etc/rc.d/init.d/functions on a RH box
 status() {
        local base=${1##*/}
        local pid

        # Test syntax.
        if [ "$#" = 0 ] ; then
                echo $"Usage: status {program}"
                return 1
        fi

        # First try "pidof"
        pid=`pidof -o $$ -o $PPID -o %PPID -x $1 || \
            pidof -o $$ -o $PPID -o %PPID -x ${base}`
        if [ -n "$pid" ]; then
                echo $"${base} (pid $pid) is running..."
                return 0
        fi

        # Next try "/var/run/*.pid" files
        if [ -f /var/run/${base}.pid ] ; then
                read pid < /var/run/${base}.pid
                if [ -n "$pid" ]; then
                        echo $"${base} dead but pid file exists"
                        return 1
                fi
        fi
        # See if /var/lock/subsys/${base} exists
        if [ -f /var/lock/subsys/${base} ]; then
                echo $"${base} dead but subsys locked"
                return 2
        fi
        echo $"${base} is stopped"
        return 3
 }
fi

# pull in sysconfig settings
[ -f /etc/ubisense.conf ] && . /etc/ubisense.conf
```

```
PLATFORM_USER=${PLATFORM_USER:-platform}
UBISENSE_LOCAL_CONTROL=/home/platform/bin/i586_linux_2.6/ubisense_local_control
export UCONFIG=/etc/ubisense/platform.conf

RETVAL=0
prog="ubisense"

start()
{
        echo -n $"Starting ubisense_local_control:"
        if [ -e /etc/rc.d/init.d/functions ]
        then
                daemon --check ubisense_local_control --user=platform ${UBISENSE_LOCAL_CONTROL}
-d
         else
                startproc -u platform ${UBISENSE_LOCAL_CONTROL} -d
        fi
        touch /var/lock/subsys/ubisense_local_control
        echo
}

stop()
{
        echo -n $"Stopping ubisense_local_control:"
        if [ -e /etc/rc.d/init.d/functions ]
        then
           killproc ubisense_local_control
        else
          killproc ${UBISENSE_LOCAL_CONTROL}
        fi
         rm -f /var/lock/subsys/ubisense_local_control
        echo
}


case "$1" in
      start)
                start
                ;;
      stop)
                stop
                ;;
      restart)
                 stop
                start
                ;;
      status)
                 status ubisense_local_control
                ;;
      *)
                echo $"Usage: $0 {start|stop|restart|status}"
                RETVAL=1
esac
```

```
exit $RETVAL
```

Sample systemd scripts for a RedHat Linux machine

The following example illustrates the use of systemd scripts for SmartSpace with the core and controller executables on a single Red Hat® Linux machine.

> ℹ The instructions assume the core server and local controller executables (**ubisense_core_server** and **ubisense_local_control**) are in **/home/platform/bin/i586_linux**. If this is not the case, the service files (**ubisense_core_server.service** and **ubisense_local_control.service**) will have to be updated to reflect the location of the executable.

1. Add a target file **ubisense_service.target** in **/etc/systemd/system** containing the following:

   ```
   [Unit]
   Description=ubisense_service Target
   Requires=multi-user.target
   After=multi-user.target
   AllowIsolate=yes

   [Install]
   WantedBy=multi-user.target
   ```

2. Run the following commands:

   ```
   systemctl list-units --type service
   systemctl daemon-reload
   systemctl enable ubisense_service.target
   systemctl isolate ubisense_service.target
   ln -sf /etc/systemd/system/ubisense_service.target
   /etc/systemd/system/default.target.wants/
   ```

3. Reboot the machine.

4. Check the status of the target using the command below to make sure the target is active and running:

   ```
   systemctl list-units --type target
   ```

5. Add a service file **ubisense_core_server.service** in **/etc/systemd/system** containing the following:

```
[Unit]
Description=ubisense_core_server daemon
After=multi-user.target

[Service]
Type=forking
ExecStart=/home/platform/bin/i586_linux_2.6/ubisense_core_server -d
Environment=DOTNET_ROOT=/usr/lib64/dotnet
User=platform
Group=operations

[Install]
WantedBy=ubisense_service.target
```

6. Add a service file **ubisense_local_control.service** in **/etc/systemd/system** containing the following:

```
[Unit]
Description=ubisense_local_control Daemon
After=multi-user.target

[Service]
Type=forking
ExecStart=/home/platform/bin/i586_linux_2.6/ubisense_local_control -d
User=platform
Group=operations
TasksMax=3072

[Install]
WantedBy=ubisense_service.target
```

Note: The **TasksMax** value highlighted above is an example only and may need to be changed depending on the version of Linux you are using and the particular Ubisense products installed.

7. Run the following commands:

```
systemctl daemon-reload
systemctl enable ubisense_local_control.service
systemctl enable ubisense_core_server.service
```

8. Reboot the machine.

9. To list the status of the services run the following command:

```
systemctl list-units --type service
```

Service security and Authentication using Cached Service Credentials on Linux

If you are running on a Linux server and configure a security policy in Security Manager that requires services to authenticate as a user, using **ubisense_cache_service_credentials**, then you must run the core and controller software with the **-d** flag (as shown in the examples above). Otherwise all services will still have a connected stdin/stdout and will attempt to prompt for credentials rather than reading the cached service credentials. See *Ubisense Security Manager* on the Ubisense Documentation Portal for information on configuring security.

Configuration Parameters

On Linux, the local configuration parameters for each core or controller machine are set, by default, in a configuration file. This file contains configuration parameters for the local platform processes, such as the location of the dataset and the networking mode. The default location expected by all platform executables is **/etc/ubisense/platform.conf**. If another location is to be used, then the environment variable **UCONFIG** should be defined: it is recommended this be set in startup scripts for all users on the server, but it certainly is required for the platform user and all users in the operations group. **UCONFIG** should be the full path of **platform.conf** in its desired location.

Configuration parameters are each on a single line in the file, with a colon and white-space separating the name of the parameter from the value. For example:

```
platform_dataset: /mnt/syn013/ubisense/production/dataset
no_multicast_mode: 1
server_unicast_addresses: 10.1.5.207,10.1.16.73
```

Warnings and Errors

Immediate warnings and errors when starting the two platform service executables are logged to the Linux syslog. On a typical Linux distribution they will either be in **/var/log/messages** or **/var/log/warn**. If the services will not run, check these locations for more information.

Platform Dataset

The platform dataset is the directory where both the **ubisense_core_server** and **ubisense_local_ control** services store platform state. Files in this directory comprise the configuration and ongoing operational state of the platform core, and of all the services configured to run on the local controller.

This directory should be owned by the platform user with full control. The operations group should also have read permission, to allow backup. Restore requires that the backup be copied here and all files set to have platform ownership. See *Configuring Operations Permissions*.

The default platform dataset location is **/home/platform/dataset**. To set a different location, set **platform_datase** in the **platform.conf** file.

Configuring Operations Permissions

If your Linux distribution supports sudo, then the operations group can be assigned permission to start and stop the platform services, and to change ownership of files to the platform user. For example, the following lines might be added to the end of the **/etc/sudoers** file using visudo.

```
%operations ALL = (root) NOPASSWD: /sbin/service ubisense_core_server *, \
/sbin/service ubisense_local_control *, \
/bin/chown -R platform *, /bin/chown platform *
```

With this configuration, any user who is in the operations group will be able to run **sudo /sbin/service ...** to stop, start and get the status of just the platform services. They will also be able to restore platform dataset backups and set the ownership of the restored files back to the platform user.

Backing up your Dataset

After you have set up your SmartSpace installation, ensure that you back up your dataset occasionally, so that you can recover your data. Use the **ubisense_backup** command-line tool from the **tools\linux** folder of your distribution directory to backup your dataset.

Installing Admin Machines on Linux

Administrative executables, used to configure and maintain the running state of the Ubisense platform, should be executable by the operations group.

Your distribution directory contains the following admin executables:

```
tools/linux/ubisense_backup
tools/linux/ubisense_cache_service_credentials
tools/linux/ubisense_configuration_client
tools/linux/ubisense_file_downloader
tools/linux/ubisense_installer
tools/linux/ubisense_machine_id
tools/linux/ubisense_multicast_test
tools/linux/ubisense_proxyconfig_admin
tools/linux/ubisense_restore_dataset
tools/linux/ubisense_save_dataset
tools/linux/ubisense_service_admin
tools/linux/ubisense_service_ping
tools/linux/ubisense_trace_receiver
tools/linux/ubisense_transfer_config
```

Installing Client Machines on Linux

In order to avoid the use of incompatible versions of SmartSpace administrative and configuration tools, these tools are installed into the platform along with service upgrades. You can then download the current version of each tool onto your Linux client machine using the **ubisense_file_downloader**.

Run the tool with no arguments for help.

For example, to download *all* Linux tools currently available to the current directory, run:

```
 > ubisense_file_downloader download --linux-only .
```

To force the overwriting of existing downloads, add `--force`.

Next Steps

With SmartSpace and your licensed features installed, you can configure your installation:

- If you have features that are accessed using SmartSpace web, you need to install and configure you web server. See the *Ubisense Installation Guide for SmartSpace Web* on https://docs.ubisense.com/portal/Content/Downloads.htm for further information.

- Other SmartSpace features will require additional configuration. See the requisite sections of the SmartSpace documentation website for further information or download individual guides from https://docs.ubisense.com/portal/Content/Downloads.htm.