**WAS: -** WebSphere Application Server.

**PROTOCOL: -** Protocol is a set of rules and regulations.

**Example**:-

**Http** – Hypertext transfer protocol.

**Https** – Hypertext transfer protocol security.

**TCP** – Transmission control protocol – It will divide into packets

**IP** – Transfer to destination address.

**How to find IP in windows –** IP configuration.

**How to find IP in Linux –** IP configuration.

**What is internet** – Internet is nothing but it enabled globally.

**What is intranet**–Intranet is enabled within the organization.

**NETWORK**

**LAN** – Local area network.

**MAN** –Metropolitan area network.

**WAN** –Wide area network.

**OSI Layers: -**

1. **Physical Layer** – Transmits raw bit stream over the physical medium.
2. **Data link Layer** – Defines the format of data on the network.
3. **Network Layer** – Decides which physical path the data will take.
4. **Transport Layer** – Transmits data using transmission protocols using TCP and UDP.
5. **Session Layer** – Maintains connections and it is responsible for controlling ports and sessions.
6. **Presentation Layer** – Ensure that data is in a usable format and is where data encryption is occurs.
7. **Application Layer-**Human computer interaction layer, where applications can access the network services.

**What is 3 – Tier architecture: -**

It is nothing but inter connection between the 3 components

Type 1 – Web Server.

Type 2 – Application Server.

Type 3 – Database.

Data Base

**What is Web Server**:- Webserver is a server and it is responsible for static content and it will run on ports.

**Webserver Logs:-**

Webserver logs can derived into two logs

was7host01:/opt/IBM/HttpServer/logs.

* error\_log
* access\_log

**Static content** – It is nothing but it will not change for anyone same content will be served for all.

**Example** – GIF files, HTML files, JPG files and Login page (perfect)

**Port** – port is a service identifier to identify the service type.

**Plug in** :-

* Plugin is a module and it interface between Webserver and Application server.
* Plugin receives the request first and it will decide the request is for static content on dynamic content.
* Based on that it will send request to Webserver or Application server.

**What is Application server** –

* Application server is a runtime environment on which we deploy, manage and run the java and J2EE applications.
* It is majority or majorly responsible for dynamic.

**Application Server logs**:-

**JVM Logs(Java Virtual Machine)**

was7host01:/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs/servername

There two logs can be derived in to the JVM logs

* SystemErr.log
* SystemOut.log

**Process Logs:-**

was7host01:/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs/servername

Process logs can be derived into two types.

* native\_stdout.log
* native\_stderr.log

**IBM Service Log:-**

was7host01:/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs

* Activity.log

**Trace logs:**

* Trace.log

**What is LDAP** - Lightweight directory access protocol.

* It is responsible for authentication and authorization.
* Authentication is nothing but verifying the user.
* Authorization is nothing but checking the permission for requested content.

**Context root** –The location at which the module can be accessed. The context root is a part of the URL we use to connect to the application.

**URL – Uniform Resource Locator.**

* It is a unique identifier used to locate a resource on the internet and it also referred to as a web address

**URI – Uniform Resource Identifier.**

* It is a sequence of characters that distinguishes one source from another.

**Application –**

Application is nothing but set of modules.

Modules is nothing but set of pages.

Pages is nothing but combination of code.

**What are the files we deploy :-**

**. Ear** – Enterprise archive.

**. War** – Web archive.

**. Jar** – Java archive.

**What are the components in application server?**

1. **Web container** – It is responsible of user requests and will respond with Servlet’s and JSP’s.
2. **EJB (Enterprise java beans) container** – It is used for backend servers using JNDI.
3. **JNDI** – Java naming directory interface.

**IBM and WebSphere application serve**r – It have multiple versions

6.0 and 6.1

7.0

8.0 and 8.5

9.0 – The latest version.

Now using 8.5.5.21

Here 8.5 is Base version and .5 is feature pack and .21 is fix pack.

**Packages** :- In the Packages are 3 Components

1. **Base** – Standalone package.
2. **Express** – Standalone package.
3. **Network deployment** – Distributed environment package.

**Standalone** – It is nothing but single server and single node.

**Distributed environment** – It will have multiple servers and multiple nodes.

**Network deployment :-**

**Profiles** – We have different kinds of profiles and profile is a set of core binary files that represented the WebSphere application server(WAS).

**We have 7 profiles :-**

1. Application server profile
2. Custom profile
3. DMGR (Deployment manager) profile
4. Cell profile
5. Job manager profile
6. Administrator agent profile
7. Secure proxy profile

1**)Application server profile** :-When we install application server profile we get

* Application server – server1
* Application server profile node
* Console

**Console** – it is nothing but GUI to manage WebSphere application Environment

2**)Custom profile** – When we inst all custom profile we get EMPTY node

3**)DMGR profile** – When we install deployment manager profile we get

* DMGR server
* DMGR node
* DMGR console

4**)Cell profile** – when we install cell profile we get

* Application server
* Application node
* DMGR server
* DMGR node
* DMGR console
* Node agent

**Cell profile** – It is combination of application profile and DMGR profile

**Node agent** – The communication between Application profile and DMGR profile.

**Node** – Logical group of servers.

**Profile installation in terminal –**

cd /usr/software\_cds/WAS7 – ./Launchpad.sh

**Product binary path** - /opt/IBM/WebSphere/AppServer

**Core binary path** - /opt/IBM/WebSphere/AppServer/Profiles/AppSvr01 or Dmgr01

**Web container default port number**

9443 – Secure

9080 – Non Secure

**Admin console default port number**

9043 – Secure

9060 – Non Secure

**Soap connector port** – 8879

**Console** – <http://localhost:9060/ibm/console>

**In the console –**

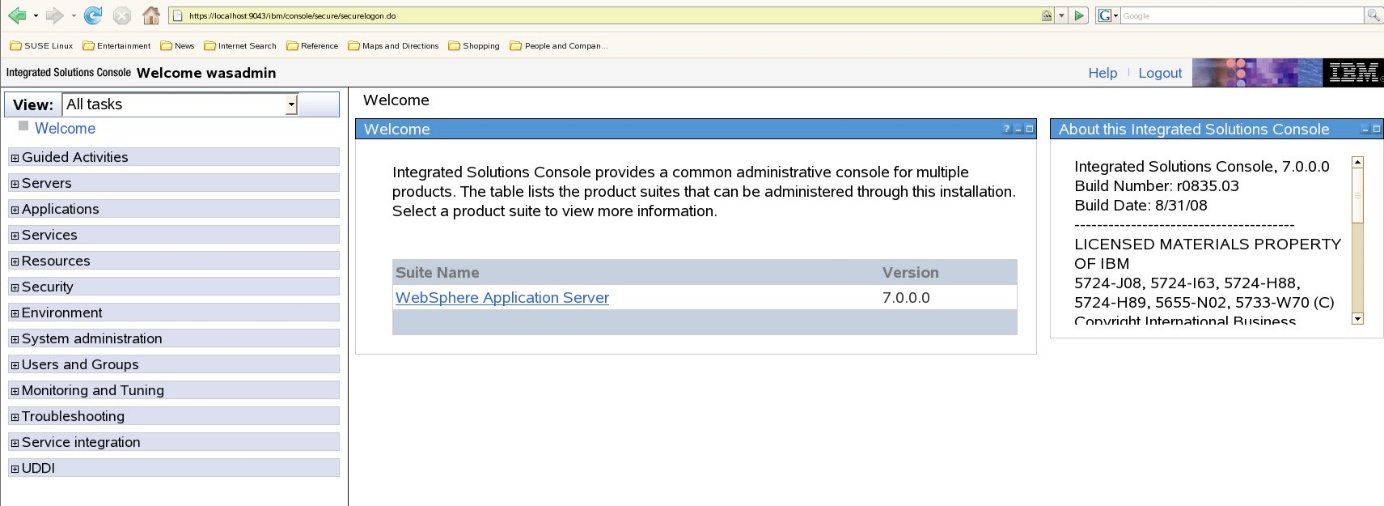
**Left side** – Navigation tree

**Middle side** – Work area

**Right side** – Help

**Upper side** – Message area or welcome

**Console :-**

****

**To start the server in terminal –**

**Go to the product binary path** – cd /opt/IBM/WebSphere/AppServer

**Go to the profiles path** – cd profiles

There we find two profiles AppSvr01 and Dmgr01

**Go to the Dmgr01 profile** – cd Dmgr01

**Go to the bin** – cd bin

**Here start the Dmgr01** – ./startManager.sh

**After starting Dmgr01 go back to profiles using** – cd ../..

**Here go to AppSvr01** – cd AppSvr01

**Go to the bin** – cd bin

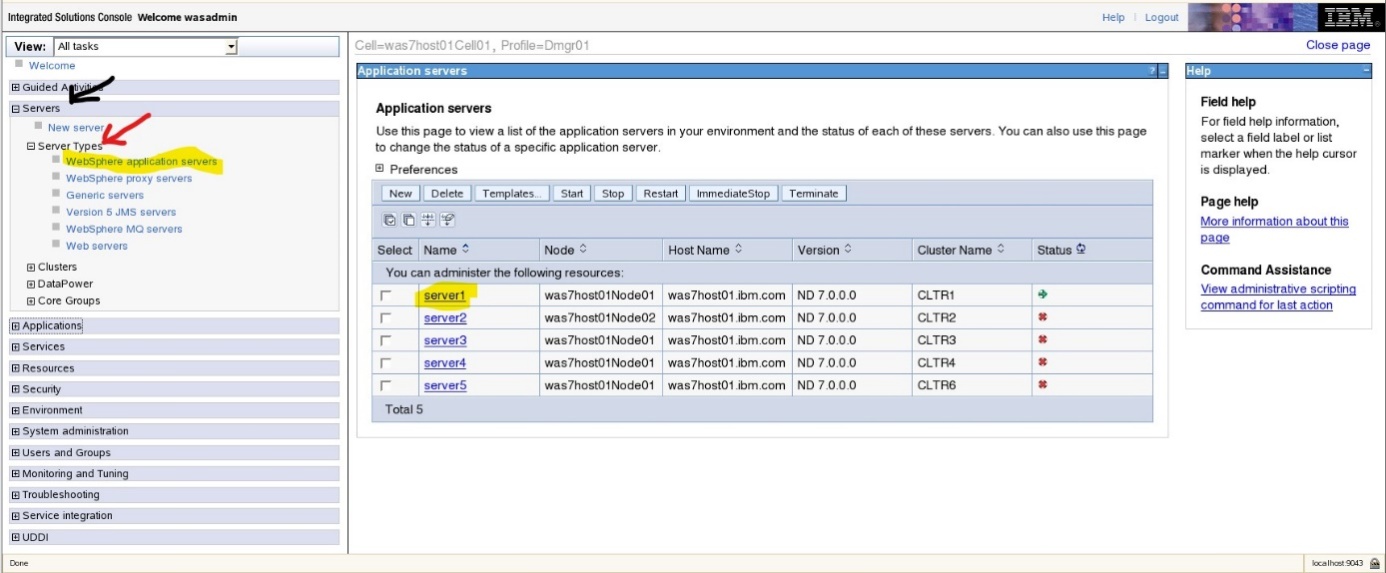
**Here start the node**– ./startNode

**After that start the server**– ./startServerservername

**Note:** - For stop process we should do as it is shown in the above but we should start from the last line –

./stopserver.sh servername> ./stopNode.sh> ./stopManager.sh

**To open application –**



Go to console

Click on servers

Open server types

WebSphere application servers

Click on Server1

Installed applications

Plants by WebSphere

Context root for web modules

Copy the url (/plantsbywebsphere)

Open new tab Open console uri using admin port number

**Profile creation in Console:-**

****

**Profile creation:-**

Go to product binary path – cd /opt/IBM/WebSphere/AppServer

Go to bin – cd bin

Go to the Profile Management – cd ProfileManagement

./pmt.sh(profile management tool)

Launch profile management

Click on create

Create a profile (custom profile or any others)

Finish

Close the recent tabs

**Profile Deletion Process in Terminal:**

was7host01:/opt/IBM/WebSphere/AppServer/bin # ./manageprofiles.sh -delete -profileName Appsrv

**Federation:- i**t is nothing but adding Custom or AppServer profiles to the Deployment manager.

Go to custom profile in the terminal cd Custom01 > cd bin >

Here add a node to federate using

./addNode localhost 8879

**Cluster :-**

Cluster is nothing but collection of servers which will do load balancing, high availability, scalability

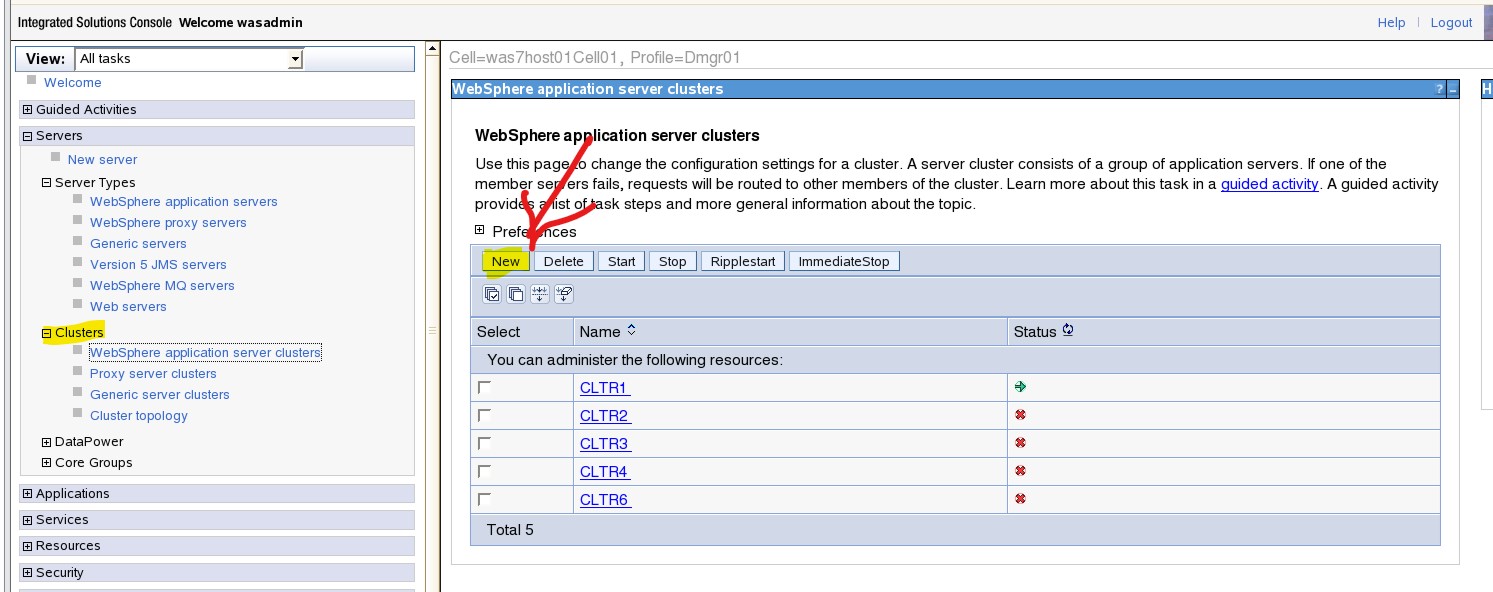
We have two types of clusters

* Vertical cluster
* Horizontal cluster

**Vertical cluster** – It is nothing but when we create a cluster using single memory location.

**Horizontal cluster** – It is nothing but When we create a cluster using different memory locations.

**Cluster Creation :-**



Go to console

Clusters

WebSphere application service cluster

New

Give a name (Example – CLtR1)

Next

Select (Create the member by converting an existing application server) at last

Next

Select the node

Next

Finish

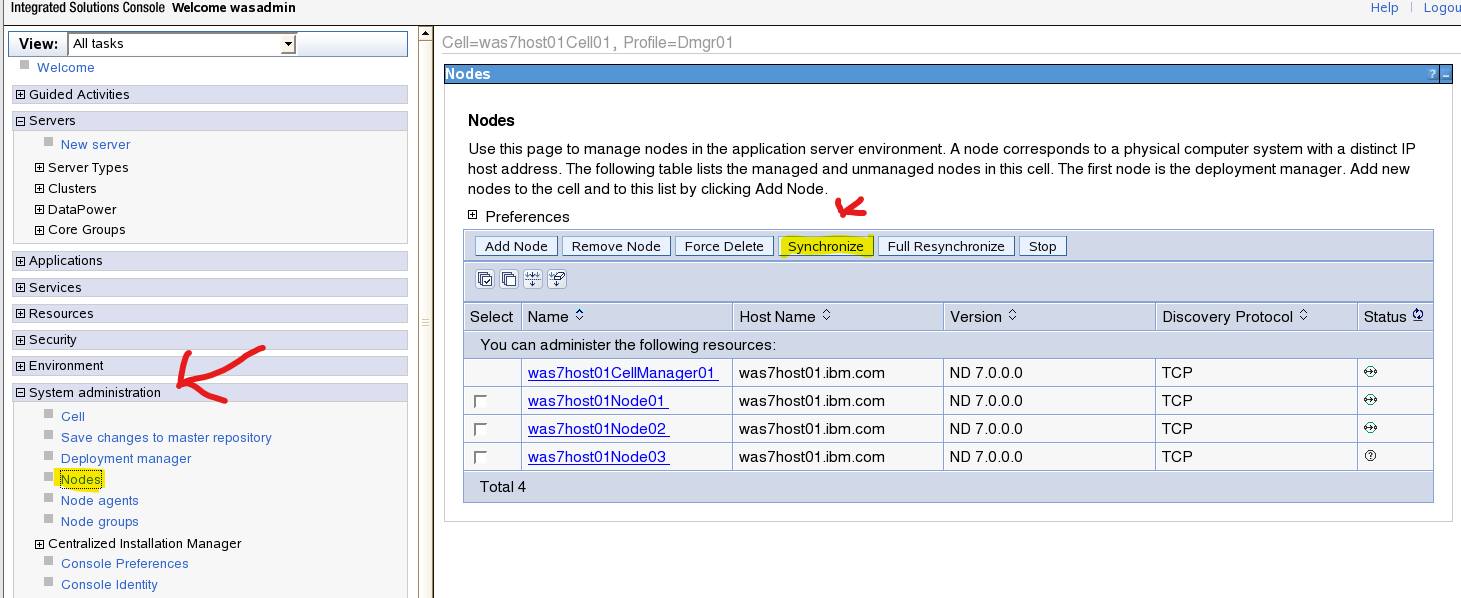
**Synchronization:-** It is nothing but a sinking data between Dmgr and application server, nodeagent will do the same synchronization.

Go to console

System administration

Nods

Synchronize

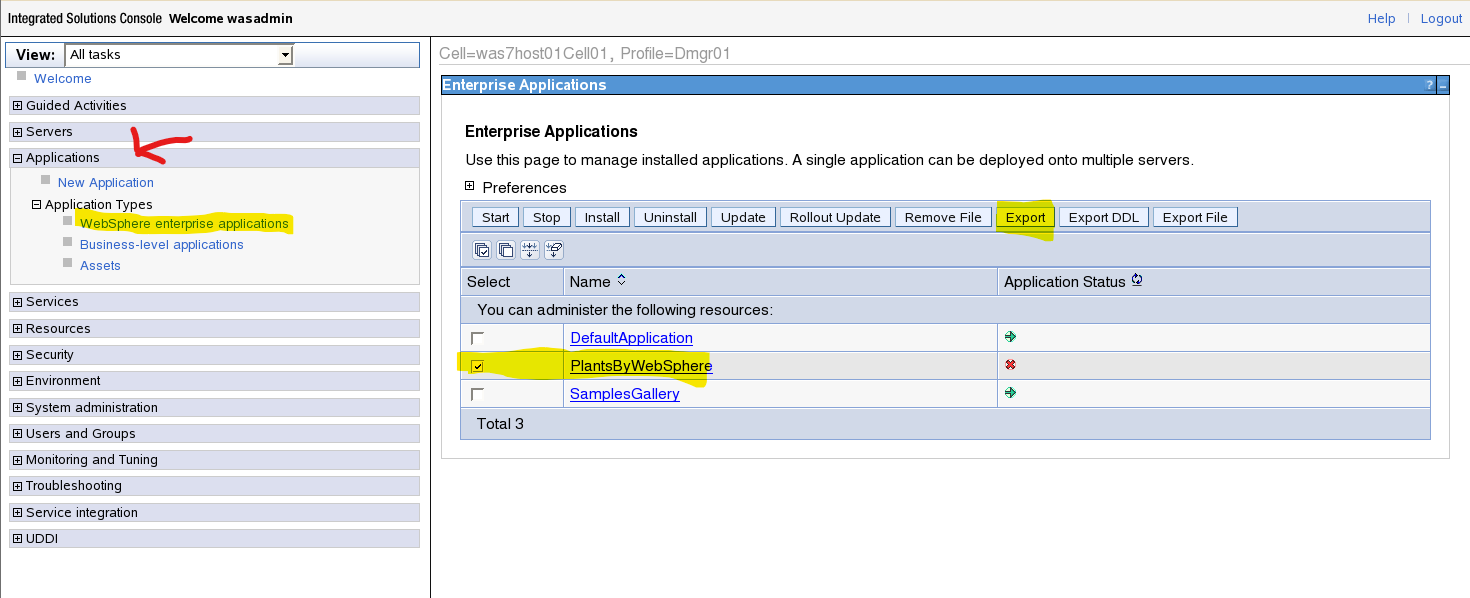
****

**Difference between synchronize and re-synchronize**

**Synchronize** – It will sink particular file and it will change.

**Re-synchronize** – It will complete all directories.

**Backup application :-**

****

Console

Applications

WebSphere enterprise application

Click on application name

Export

**Application Deployment:-**

Deployment is nothing but a putting an application in to the application server.

Applications

Application types

WebSphere enterprise applications

After export

Plants by WebSphere. Ear

Save to disk Ok

Right click and open containing folder

Copy plants by websphere

File system

tmp

Paste

And open websphere enterprise applications

Select plants by websphere

Uninstall

Install

Browse

Open file system

tmp

Click on plants by websphere

Open Next

Fast path Next

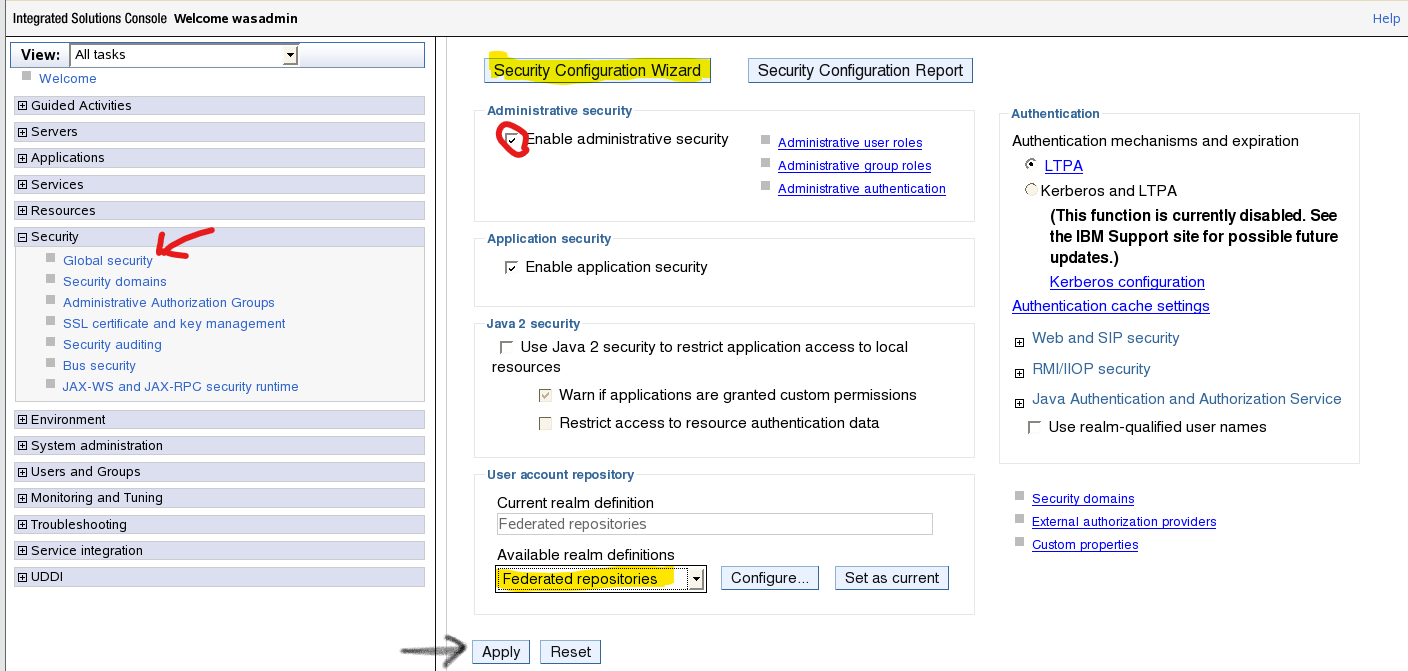
Select on modules and one server apply it Next

Finish

**Types of security repositories:-**

1. Federated repositories
2. Standalone LDAP registry
3. Local os registry
4. Custom registry

**Global security –**it will secure the environment and applications as well as using the federated repositories.



Go to console

Security

Global security

Enable administrative security

Set – federated repositories

Apply

Security conformation wizard

Next

Federated repositories

Enter or set the details username and password

Confirm password

Next

Finish

Review and save

Restart the Dmgr (deployment manager)

**Global Security Disabled:**

was7host01:/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/config/cells/was7host01Cell01 # vi security.xml

**Security Roles:-**

* Admin Security manager
* Administrator
* Auditor
* Configurator
* Deployer
* ISC Admins
* Monitor
* Operator

**Heep Memory:-**

Memory is a Object Storage Space.

**To Change the Heep Memory:-**

Servers

Server Types

Websphere Application Server

Servers

Java Process Management

Process Definition

Java Virtual Machine

Intial Heep Size

Maximum Heep Size

Apply

Review

Restart Servers

**Object:-**

Object is Nothing but a Heep.Each and Every Request As it’s own Object will be created in the JVM Heep Memory.There are two Kinds Of Object.

**Object Types**:-

**Used Object:-**

It is a Live Object. Currently Transactions are Running.

**Unused Object:-**

It is Idle One and No Transactions are Running.

**Heep Dump**:-

It is Nothing but a Blueprint of Object.

**Thread**:-

Thread is Nothing but a Request whenever the Server Receives the Request one Object will be created.in the Heep Memory.

**Different Kinds Of Threads**:-

**How to connect from application server to database ?**

Web container

JDBC

EJB container

Data source

JNDI

* Each and every application have its own database.
* The application has the JNDI details.
* Based on the JNDI details it will connect with the respectable database which will mention in the application.
* During start of the class files will be loaded and JVM will be up using the heap memory that we set.
* During that memory the EJB container will be compile the application code then it will go to the JNDI.
* The JNDI will be defined in the application so based on that it will connect to the respectful database.

**How to connect to database –**

1. **JDBC** :-

Java database connectivity. It is used to Connect to database from Application Server and it will Convert from Java code to SQL Queries Using drivers.

**Types Of Drivers:-**

We Have four Kind of Drivers

* Type 1
* Type2
* Type3
* Type4
* We will Always use Type 4 Drivers because It is Java Related Thin Driver.

1. **Data source**:-

It is nothing but a source of the Respective Application Data. Each and Every Application has its own Data in Database

IHS

IHS Installation Process:-

was7host01:/usr/software\_cds/WAS7-Supplemental\_1/IHS #./install

IHS Start Procedure:-

was7host01:/opt/IBM/HTTPServer/bin # ./apachectl -k start

ps -ef|grep http

IHS Console:-

http://localhost:80

**What is Plugin ?**

Plugin is a Module. It is a Interface between WebServer and Application Server.

Database

Application

Server

U

S

E

R

Plugin

Web Server

**Plugin Logs:-**

was7host01:/opt/IBM/HTTPServer/Plugins/logs/webserver1.

* http\_plugin.log.

**How to Generate Plugin ?**

Their Generate In Two Types.

* Console
* Command Line.

**Console:**

was7host01:/opt/IBM # cd HTTPServer/Plugins/bin/

was7host01:/opt/IBM/HTTPServer/Plugins/bin # cp configurewebserver1.sh /opt/IBM/WebSphere/AppServer/profiles/Dmgr01/bin/

was7host01:/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/bin # ./configurewebserver1.sh.

**Steps:**

Servers

Server Types

Web Servers

Click on Webserver1

Generate Plugin

Propegate Plugin

Restart Webserver1

**Command Line:-**

**Generate Plugin**:-

was7host01:/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/bin # ./GenPluginCfg.sh

was7host01:/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/bin # cp /opt/IBM/WebSphere/AppServer/profiles/Dmgr01/config/cells/plugin-cfg.xml /opt/IBM/HTTPServer/Plugins/config/webserver1/

**Webserver Logs**:-

The webserver Logs can be derived two logs.

was7host01:/opt/IBM/HTTPServer/logs

* access\_log
* error\_log

**Plugin log:**

was7host01:/opt/IBM/HTTPServer/Plugins/logs/webserver1

* http\_plugin.log

**Create a SSL Certificate**:-

**What is SSL ?**

It is a Secure Socket Layer(SSL),Transmiting data Source and destination using SSL Certificates.

**What is Certificate ?**

Certificate is a digital certificate that Authenticates a website identity and enables Encrypted connection between browser and webserver.

**Create SSL Certificate Steps:-**

was7host01:/opt/IBM/HTTPServer/bin # ./ikeyman

The Above Path can follow to open IBM Key management Tab will be opened.

Steps:-

Click on key database file

Select the key database type

Select CMS(Content Management System)

Create and select New Self sign certificate

Provide Label Name

Common Name

Organization

Organization unit

Locality

Zipcode

Country Region

Click on OK

**Modify the http.conf:-**

**Path:-** was7host01:/opt/IBM/HTTPServer/conf # vi httpd.conf

LoadModule ibm\_ssl\_module modules/mod\_ibm\_ssl.so

Listen 443

<VirtualHost \*:443>

SSLEnable

SSLProtocolDisable SSLv2

KeyFile /opt/IBM/HTTPServer/bin/test.kdb

</VirtualHost>

#KeyFile /opt/IBM/HTTPServer/bin/test.kdb

#SSLDisable

# End of example SSL configuration

**URL:-** <https://localhost:443/>

**What is Public Key ?**

Public key that can be obtained and used by Anyone to encrypt Messages.Intended for a particular Receipient.

**What is Private key ?**

A Private Key is also Known as Secret Key Here we used to Have the Algorithm to Encrypt and Decrypt the data.Secret key Should only shared with the who Generated the Key are Authorized Parties to Decrypt the data .

**What is Trust Store ?**

A Trustore contains the Signer Certificates.Which the end point trusts.

Signer:-

Certificates can be Signed by self are by Certificate Authorities.

Ex:- Veresigns, Interest,Wild craft

They will provide below certificates.

Root, Intermediate,CA

**What is 2 way SSL ?**

In 2 way SSL Authentication the Client Application verifies the Identify of the Server Application and then the Identify of the server Application and then the server Application verifies the identify of the client Application Both parties shares the public certificates and then Validation is Performed.