

## Scenario = Explain SCC “Building & Deploying 1st Comm Application”?

### Step 1 = Get SAP Commerce Cloud Subscription.

Check if you have access in SAP Customer Experience Cockpit = <https://cockpit.cx.cloud.sap/>

You can enter into this cockpit only if you are “Super Admin / Super Admin given access”.

Q = How to check who is my “Super Admin”?

<https://launchpad.support.sap.com/> & Login with ur credentials & Request User.

The screenshot shows the SAP User Management interface. At the top, there is a search bar with the placeholder "Enter search term". Below the search bar, there are four categories: "Users" (0), "Requested Users" (0), "Deleted Users" (0), and "Important Contacts" (395). The "Important Contacts" section is highlighted with a green box. The main table lists users with columns: User ID, Last Name, First Name, Customer Number of User, Function, Global, CCoE Number, and CCoE Name. A user named "S00134563" is selected, and their details are shown in a modal dialog. The "Function" field is "Super Administrator", which is also highlighted with a green box. The URL in the browser address bar is "https://launchpad.support.sap.com/#/user/management".

**User Management - SAP ONE Subsystem**

Request User

User ID	Last Name	First Name	Customer Number of User	Function	Global	CCoE Number	CCoE Name
S00134563	Na	Sree	12777	Super Administrator			

Authorization Packages: [Red Box]

Request User [Green Box]

### Step 2 = Provision Environments

Login to SAP CX Cockpit = <https://cockpit.cx.cloud.sap/>

After login → SAP Commerce Cloud

For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN



== Now we can see the Commerce environments that we have access to, as well as their status, location and license details.

If our Env have status = “Open”, then click “Provision” button to trigger provisioning process. Once finished, Env changes its status to “Active”.

That means that we can start using it to **build and deploy** application.

The screenshot shows the SAP C/4HANA Cockpit interface under the Entitlements section for SAP Commerce Cloud. The left sidebar includes Home, Entitlements (selected), Security, Authorizations, Audit Logs, and Extensibility. The main area displays a list of products for SAP Commerce Cloud, showing two entries: 'Productive' (Status: OPEN) and 'Staging' (Status: OPEN). The 'Staging' entry has a green arrow pointing to its 'Status' field, which is currently 'OPEN'. Below the table, a yellow arrow points to the 'ACTIVE' status of the 'Staging' environment, indicating it has been provisioned. The bottom navigation bar includes fields for System ID, System Name, Provider, Location, License Information, and Actions, along with links for Details and Provision.

**Step 3 = SAP Commerce Cloud Portal [SCCP] [URL = <https://portal.commerce.ondemand.com/> ]**

It's a self-service tool for managing our SAP Commerce Cloud environment.

To access Cloud Portal, needs to have a valid **S-user ID** & Role = Customer System Administrator.

**Note:** - Access rights to **SAP CX Cockpit & SAP Commerce Cloud Portal** are handled separately.

The screenshot shows two views of the SAP Commerce Cloud User Management interface. The left sidebar contains navigation links for Deployment (Environments, Builds), Subscription Resources (Repository, Security, Static Files, Site Availability), and User Management. The right panel displays the 'User Management' section with a table of existing users and a 'Create' button. The 'Create' button and the 'Add User' form are highlighted with green boxes. The 'Add User' form includes fields for S-User ID (S0001234), Email (chennarrrs@gmail.com), Role (selected as CUSTOMER\_SYS\_ADMIN, also showing CUSTOMER\_DEVELOPER and CUSTOMER\_LOG\_VIEWER), and an Enforce Two-Factor Authentication checkbox.

**Q = Explain Customer Roles?**

- 1) CUSOMTER\_SYS\_ADMIN
  - a. Can Deploying builds
  - b. Can Performing administrative tasks
  - c. Can Managing users
- 2) CUSOMTER\_DEVELOPER
  - a. Can Accessing Subscription
  - b. Can Deploying builds
- 3) CUSOMTER\_LOG\_VIEWER
  - a. Can only Viewing Kibana logs
  - b. No access to commerce cloud portal interface

**SAP Commerce Cloud**

DEPLOYMENT

- Environments
- Builds

SUBSCRIPTION RESOURCES

- Repository
- Security
- Static Files
- Storefront Availability
- User Management
- Extension Factory

## Environments

Manage environments for your current application.

stag1		Commerce Cloud Overview Environment	
Type	Staging	Type	Staging
Build	baseline_1905 20191024.1 Deployed	Status	Available
Status	Available		

**Step 4 = Connect SCC instance to a git-based repository.**

Each subscription for SCC comes with ability to connect to a single Git-based repository.

Git repository needs to be on a publicly accessible server (not behind corporate env firewall).

GIT can be accessed “SSH & HTTPS” methods.

Goto GIT Repo → "Clone / Download" button & Copy. We can use either HTTPS or SSH.

For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

Goto SCCP → Repository

The screenshot shows the SAP Commerce Cloud interface with the sidebar navigation. Under 'Subscription Resources', the 'Repository' option is selected. On the main page, there is a 'Repository' section with a 'Repository URL' input field containing 'ssh://git@github.com/l349940/core-customize.git'. A red arrow points to the 'git@github.com/l349940/core-customize.git' part of the URL with the text 'Replace : with /'. Below the URL, there is a 'Public Key' section with a long RSA key.

**Note:** - Code repository needs to be accessed from Azure and an authorization is required.

To provide the authorization, I'll generate a new Public key & Copy.

This screenshot shows the same SAP Commerce Cloud interface as above, but with a green arrow pointing to the 'Copy to Clipboard' button next to the 'Regenerate' button. The 'Public Key' section still displays the RSA key.

Goto GIT Repo → Settings → “SSH and GPG keys” from the side menu and then “New SSH key”.

The screenshot shows the GitHub settings page under 'SSH and GPG keys'. A green box highlights the 'SSH and GPG keys' link in the sidebar. A green arrow points to the 'New SSH key' button at the top right of the main area. A yellow arrow points upwards from the 'New SSH key' button towards the 'Signed in as l349940' dropdown menu. A green box highlights the 'Settings' link in the dropdown menu. A green box also highlights the 'Give Title' and 'Paste Public Key' fields in the 'SSH keys / Add new' modal window.

For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

## Results =

The screenshot shows the 'Personal settings' section of the SAP Cloud Platform account. The 'SSH and GPG keys' tab is selected. Under 'SSH keys', there is one entry: 'Commerce Cloud SSH Key' with a key icon, ID 3d:ef:e7:84:3a:4f:a4:1a:7f:f4:fb:0b:6f:1c:7c:bd, added on Mar 3, 2020, and labeled 'Never used — Read/write'. A 'Delete' button is visible. Below the list is a link to 'Check out our guide to generating SSH keys or troubleshoot common SSH Problems.' On the right, a green 'New SSH key' button is present.

SSH keys

New SSH key

Commerce Cloud SSH Key  
3d:ef:e7:84:3a:4f:a4:1a:7f:f4:fb:0b:6f:1c:7c:bd  
Added on Mar 3, 2020  
Never used — Read/write  
Delete

Check out our guide to generating SSH keys or troubleshoot common SSH Problems.

GPG keys

New GPG key

## Step 5 = Running SAP Commerce Instance Locally

To start quickly, use free code sample rep = <https://github.com/SAP-samples/cloud-commerce-sample-setup>

The screenshot shows a GitHub repository page for 'SAP-samples/cloud-commerce-sample-setup'. The '2005-spartacus' branch is selected, highlighted with a green box. The page displays 4 branches and 0 tags. A message indicates the branch is 29 commits ahead of master. The repository structure includes 'tylernmac-sap -Removing SSR', 'core-customize' (with a note 'Upgrade to Spartacus 2.0 (non-SSR)'), and 'js-storefront' (with a note '-Removing SSR'). On the right, there are options to 'Clone with HTTPS' (with a URL), 'Open with GitHub Desktop', and a 'Download ZIP' button, also highlighted with a green box. Below the repository view, two file explorers show the directory structure of the downloaded ZIP file. The left explorer shows the contents of 'chenarrs' folder: 'core-customize', 'js-storefront', 'LICENSE', 'NOTICE', and 'README.md'. The right explorer shows the contents of 'hybris/bin': 'custom', 'modules', and 'platform'. A large green arrow labeled 'Copy & Paste' points from the 'modules' folder in the right explorer to the 'modules' folder in the bottom-right file explorer, which contains the text 'Standard SAP Comm 2005'.

GitHub - SAP-samples/cloud-commerce-sample-setup

Select Branch

2005-spartacus 4 branches 0 tags

This branch is 29 commits ahead, 16 commits behind master.

tylernmac-sap -Removing SSR

core-customize Upgrade to Spartacus 2.0 (non-SSR)

js-storefront -Removing SSR

Clone with HTTPS

https://github.com/SAP-samples/cloud-commerce-sample-setup

Open with GitHub Desktop

Download ZIP

OS (C:) > chenarrs >

library Share with

Name

- core-customize
- js-storefront
- LICENSE
- NOTICE
- README.md

OS (C:) > chenarrs > core-customize > hybris > bin >

library Share with Burn New folder

Name Date modified

- custom
- modules
- platform

OS (C:) > rrrssoftware > hybris > bin >

library Share with Burn New folder

Name

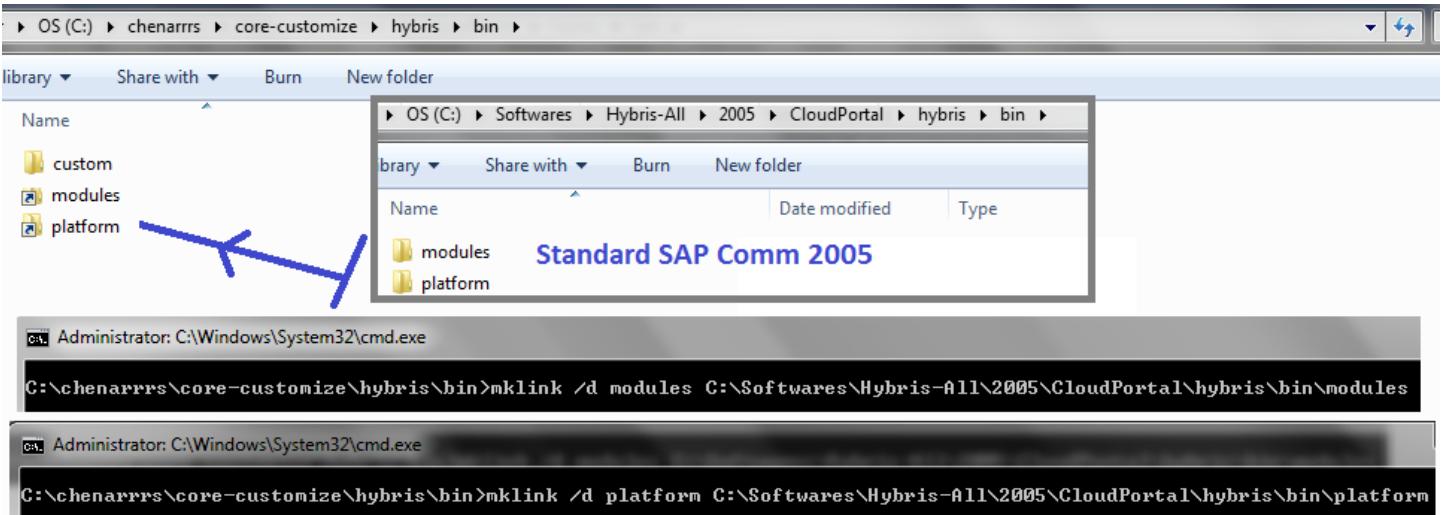
- custom
- modules
- platform

Standard SAP Comm 2005

Copy & Paste

Results -- After Download  
... Unzip ... Rename

**Note:** - If we get error saying “file name(s) would be too long for destination folder” -- do below:



**Note:** - Now -- Set Up Your Git Repository & Get it working locally

In this example – We are looking for a working template to get started various code samples.

We will see – How to get started with **sample code** which contains the structure for a working **Spartacus-based storefront** as well as a sample accelerator-based storefront.

**Development team** mostly be developing and testing customizations **locally**. So, we should try to structure repository to allow for building **both locally and on your SAP Commerce Cloud [SCC] environments**. To do this, you'll need to keep in mind a few things: -

**1)** Ensure you're using your **gitignore** file to only upload items to your repository that aren't automatically added as part of SCC build automation process.

**Example** = We shouldn't add SAP Commerce "bin" folder as relevant extensions in manifest file & will be automatically added as part of the build

**2)** This sample code assumes that we want to develop with **Cloud Hot Folders locally**.

If we are on Commerce **2005** the **sap-ccv2-hotfolder** module is included.

If we are on Commerce **1905**, we need to get extensions from **Cloud Extension Pack**.

Below is the more Doc for “Testing Cloud Hot Folders in a Local Environment” =

<https://help.sap.com/viewer/403d43bf9c564f5a985913d1fbfb8d7/SHIP/en-US/ba9c1917d4234292ba87dc3f9f1acd83.html>

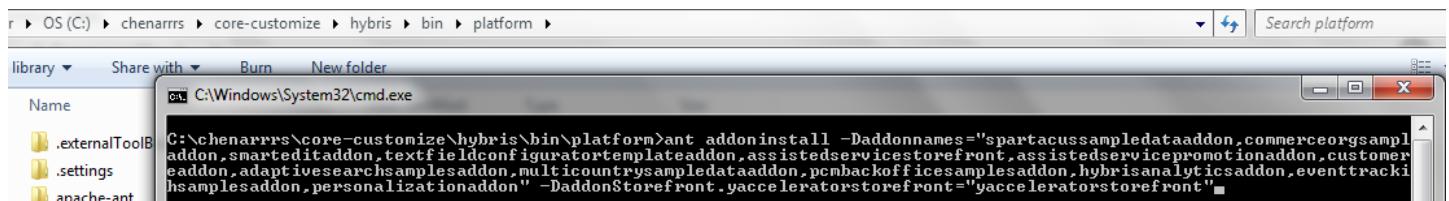
**Step 6** = Complete the steps for the installation steps now

**6.1** = Setting up Apache Ant

**6.2** = Setting up Java 11

**6.3** = Run / Install required add-ons

```
C:\chenarrs\core-customize\hybris\bin\platform>ant           addoninstall  
Daddonnames="spartacussampledaddon,commerceorgsamplesaddon,promotionenginesa  
mplesaddon,smarteditaddon,textfieldconfiguratortemplateaddon,assistedservicestorefront,as  
sistedservicepromotionaddon,customerticketingaddon,orderselfserviceaddon,adaptivequeries  
amplesaddon,multicountrysampledaddon,pcmbackofficesamplesaddon,hybrisanalyticsadd  
on,eventtrackingwsaddon,personalizationsearchsamplesaddon,personalizationaddon"  
DaddonStorefront.yacceleratorstorefront="yacceleratorstorefront"
```



**6.4** = Run below If SAP Commerce <=1905

**Note:** - Not required this step for 2005 as webservices are no longer addons.

```
ant      addoninstall      -Daddonnames="acceleratorwebservicesaddon,cmsoccaddon"  
DaddonStorefront.ycommercewebservices="ycommercewebservices"
```

**6.5** = Do the build

```
C:\chenarrs\core-customize\hybris\bin\platform>ant clean all
```

#### After build – [Optional]

(Optional) If you want to have a [local Blob storage](#) instance for testing out Cloud Hot Folders you can go into hybris/config/local.properties file and un-comment the lines with the following keys:

```
cluster.node.groups  
azure.hotfolder.storage.container.hotfolder  
azure.hotfolder.storage.account.connection-string
```

**6.6** = Perform Initialization [INIT]

```
C:\chenarrs\core-customize\hybris\bin\platform>ant initialize
```

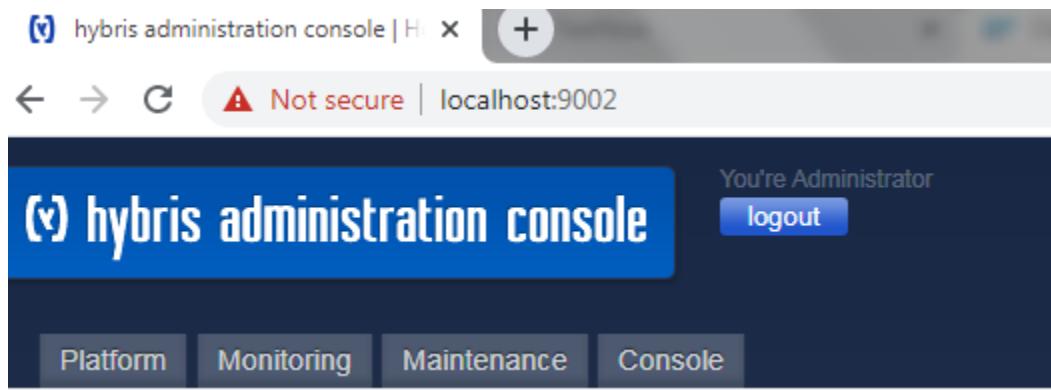
**6.7** = Start the Hybris Server

```
C:\chenarrs\core-customize\hybris\bin\platform>hybrisserver.bat
```

**For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN**

**6.8 = Check SAP Comm Server is up & running by going to hAC**

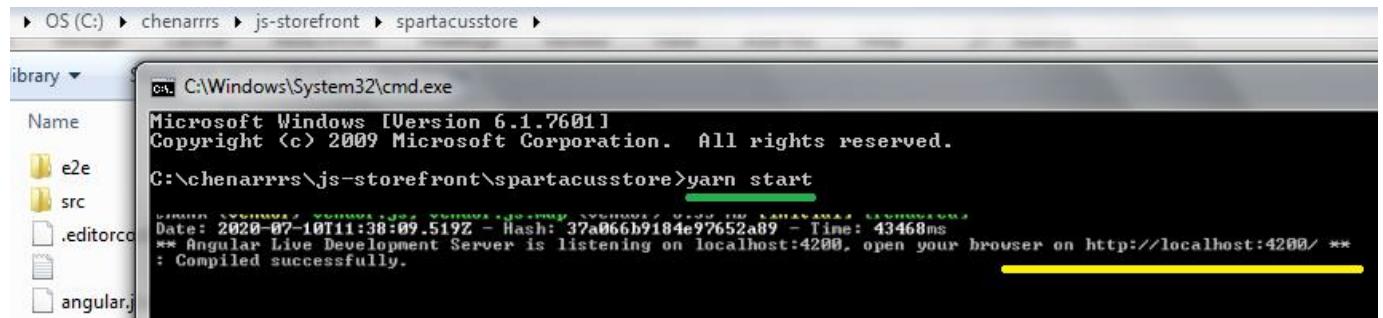
<https://localhost:9002/login> [admin nimda]



**Step 7 = Get Spartacus Storefront working.**

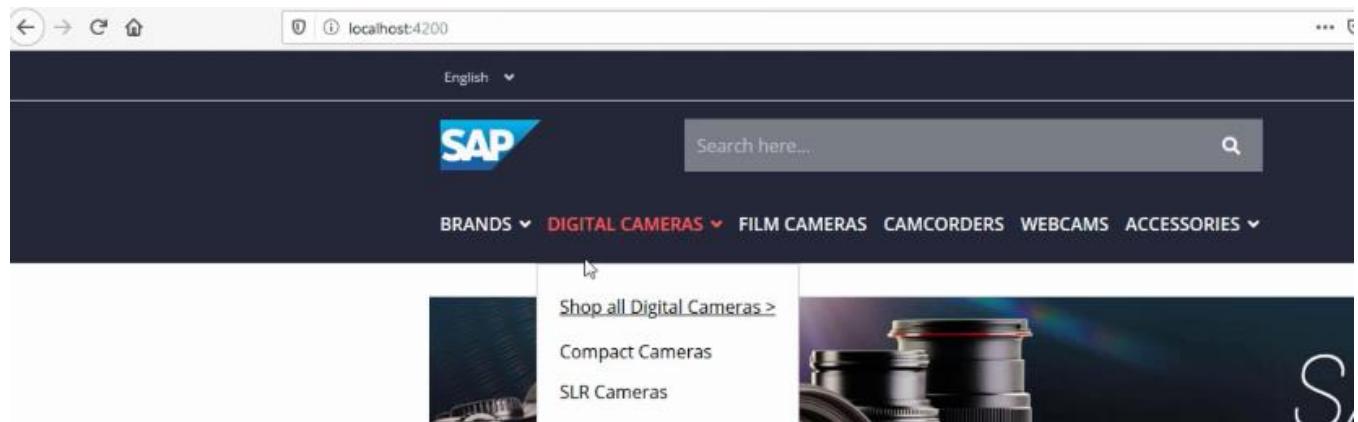
**7.1 = Build & Running Spartacus Storefront**

C:\chenarrs\js-storefront\spartacusstore>**yarn start**



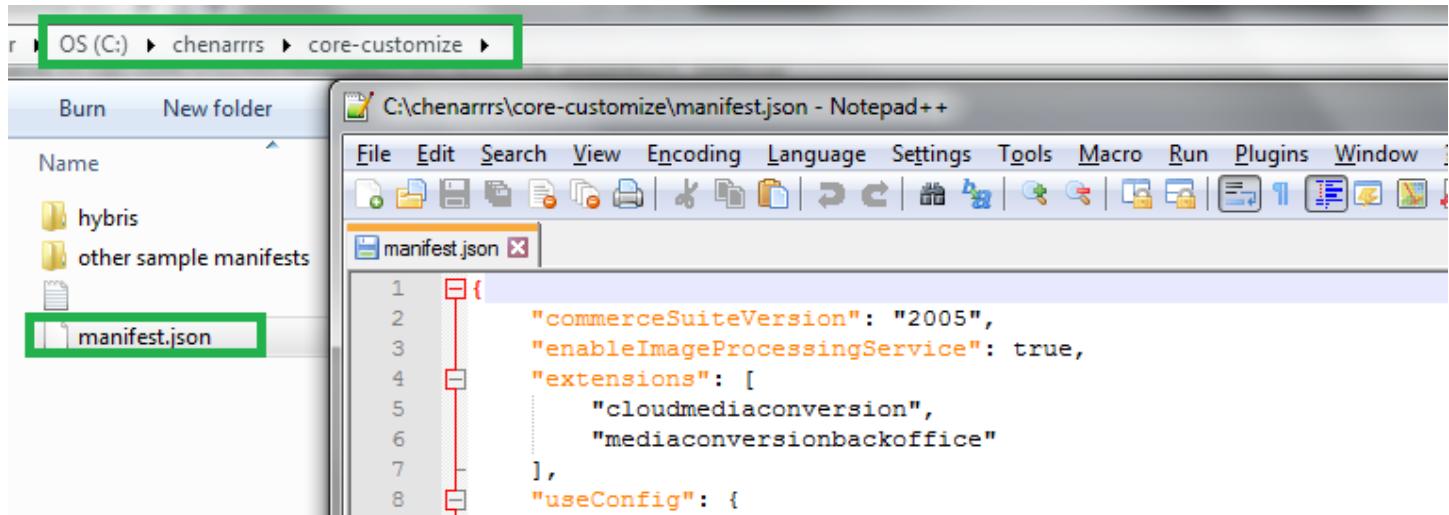
**7.2 = Check Spartacus Storefront**

<http://localhost:4200>



## Step 8 = Verify Build “manifest.json” file

“Build Manifest” enables us to configure SCC solution to our exact specifications.



**Note:** - Now, it's confirmed that, everything works good locally.

## Step 9 = Now push the code changes to GIT Repo

Create PR & Merge it to develop branch.

## Step 10 = SCCP Security Configurations

We don't able to view / edit security configurations after creating. This is to ensure files remain secure. If we create a configuration with a mistake, it's best to delete it and start again.

### 10.1 = Configure IP Filter Sets

IP filter sets – Used to control which IPs can or cannot access our various endpoints.

By default, for each of our Env endpoints are set to "Deny All" access. So, configure allowed IP's.

Remember including the public IPs & our VPN IPs as well.

Have list of IPs' in text file & upload it.

Deployment

- Environments
- Builds
- Subscription Resources
- Repository
- Security**
- Static Files
- Site Availability
- User Management
- Extension Factory

## Security

Manage your firewall rules and SSL certificates.

**IP Filter Sets** (highlighted with a green oval)

SSL Certificates Trusted Certificates Host Alias Sets VPN connections Security Files

**Create**

**IP Filter Sets**

Name *	Chenna RRRS
IP Filter Set *	<div style="border: 1px solid #ccc; padding: 5px;">            Drag TXT file or Browse       </div>
Comment	IP's Uploading

**10.2 = SSL Certificates [Used for = (1) Secure WebSite, (2) Protection from Cyber Criminals & (3) Sensitive data Encryption].**

SCC provides default SSL certificates for each of its endpoints, we may choose to use your own. If we have SSL certificates, then add them now & update our endpoint to point to the new certificate. This way we are testing with your certificates right from the start.

Deployment

- Environments
- Builds
- Subscription Resources
- Repository
- Security**
- Static Files
- Site Availability
- User Management

## Security

Manage your firewall rules and SSL certificates.

**IP Filter Sets** **SSL Certificates** (highlighted with a green oval)

Trusted Certificates Host Alias Sets VPN connections Security Files

**Create**

**SSL Certificates**

Name	Issuer	Domains	Expiry	Connected Endpoints
solutions.████████.com	CN=Sectigo RSA Organization Validation Secure Server CA,O=Sectigo	solutions.████████.com, solutions.████████.com	05/03/2022	P1 : COMM... P1 : BACKOFFICE

**Basic Information**

Name *	Main Certificate	Description
		Storefront Certificate

**Certificate Configuration**

**Certificate File \***

```
-----BEGIN CERTIFICATE-----
MIICUCCARBCFGH86Qpm6l0zHpgF49cYBwP/BwMA0GCSqGSIb3DQEBCwUAMIGJ
MQw+CQYDVQQGEwJVUzEQA4GA1UECAwHSWxsW5vczEQMA4GA1UEBwwHg2hpY2Fn
-----
```

**Key File \***

```
-----BEGIN RSA PRIVATE KEY-----
MIICXAIQAAKRpQCip0HoZQ1TmND0aHDKz3mH08PISgJvSOALdslKCQn4e3K9es
w3tsclclmxZBDJ0IJhBhZRN9Ts3g/T+YRfTu843Ps6OvaqrlbyteadlPW
-----
```

For “SAP Commerce Cloud” --- Contact Us = [ChennaReddyTraining@RRRS.CO.IN](mailto:ChennaReddyTraining@RRRS.CO.IN)

## 10.3 = Trusted Certificates

If we work with external services [Eg – SCPI] that require special certificates, then create those.

The screenshot shows the SAP Commerce Cloud interface under the 'Security' tab. The 'Trusted Certificates' tab is selected, highlighted with a green border. A table lists a single certificate entry:

Name	Alias	Issuer	Domains	Expiry	Action
scpi-qa	scpid	CN=DigiCert CA	*.hci.us3.hana.ond... .com,*.hci.us3.hana...	13/02/2020	

A green 'Create' button is located in the top right corner of the table area.

## 10.4 = Host Alias Sets

Configure the host aliases to establish a connection to my trusted external systems.

The screenshot shows the SAP Commerce Cloud interface under the 'Security' tab. The 'Host Alias Sets' tab is selected, highlighted with a green border. A form is displayed for creating a new host alias set:

**Name \***: Chenna RRRS Aliases

**Host Alias File \***: A file named 'my-first-host-aliases.txt - Notepad' is shown, containing the following content:

```
my-first-host-aliases.txt - Notepad
File Edit Format View Help
8.8.8.8 googledns1
8.8.4.4 googledns2
```

A green arrow points from the 'Host Alias File' input field to a 'Drag TXT file or Browse' button.

**Description**: Chenna RRRS Aliases - Configure

## 10.5 = VPN Connections

There are many different options for enabling and managing VPN connections.

The screenshot shows the SAP Commerce Cloud interface under the 'Security' tab. The 'VPN connections' tab is selected, highlighted with a green border. A message indicates that a VPN connection requires enabling VPN first:

To create a VPN connection you have to enable VPN first.

**Enable VPN**

## 10.6 = Security Files

We want to use Cloud Portal to safely upload a security file. Examples of these include certificates, passwords, salts, and tokens.

The screenshot shows the SAP Cloud Portal interface under the 'Security' section. On the left, a sidebar lists various options: Deployment, Environments, Builds, Subscription Resources (which is selected), Repository, Security (selected), Static Files, Site Availability, User Management, and Extension Factory. The main area is titled 'Security' with the sub-instruction 'Manage your firewall rules and SSL certificates.' Below this are tabs for IP Filter Sets, SSL Certificates, Trusted Certificates, Host Alias Sets, VPN connections, and Security Files (the latter is highlighted with a green box). A 'Create' button is visible. A table lists existing security files: Dev\_JKS (samlKeystore.jks, uploaded on 13/03/2020, connected to DEV JKS). A modal window for creating a new file is open, showing fields for Name (Chenna RRRS Password Files), Security File (passwords.txt, with a 'Browse' button), and Description (Storing Files [Contains Passwords]).

## Step 11 = Static Files

Used to store sensitive properties [Like passwords].

**Note:** - Be aware that anyone with Customer System Administration role in Cloud Portal will be able to see these property values.

Best option is to define the properties as part of each environment's service configuration.

**Example** =Let's keep SCC Backoffice credentials.

→ SCCP → Environments → Select Env [Dev / Staging / Prod] – Goto Services Section

Backoffice Service – Add properties -- Property will be activated after services restarted.

The screenshot shows the SAP Cloud Portal interface under the 'Environments' section. The sidebar includes Environment (selected), Builds, Subscription Resources, and Repository. The main area shows the 'Backoffice' service details. It has tabs for Replicas, Manual Operation History, Properties (selected), Stop, and Restart. In the Properties section, two properties are listed: storefront.resourceBundle.cacheSeconds=1 and sample.password=chenna\$rrrs#. The value for sample.password is highlighted with a red box and a green checkmark.

**Step 12** = After setting up all security settings, we want to apply them to our endpoints.

**Example** – JS-storefront endpoint. Same concepts apply to all endpoints.

The screenshot displays two main sections of the SAP Commerce Cloud interface. The top section shows the 'Commerce Cloud Overview Environment' with a 'Status' of 'Available', a 'Build' of 'Commerce Cloud Overview Build 20200227.7 1905.0', and a 'Deployment' status of 'Deployed'. It lists 'Public Endpoints' including 'Backoffice' and 'JS Storefront'. The 'JS Storefront' entry is highlighted with a yellow box and has a green checkmark next to it. The bottom section shows the 'Edit Endpoint' configuration page for 'JS Storefront'. It includes fields for 'Name' (JS Storefront), 'Domain' (https://jsapps.cyih0n-pseemea1-s5-public.model-t.cc.commerce.c), 'Web Proxy' (Default (IP 13.80.23.130)), 'Service' (JS Storefront), 'SSL Certificate' (Main Certificate), and 'Proxy Timeout (seconds)'. The 'IP Filter Sets' section shows rules: 'Allow all' (Base Rule) and 'Deny North Korea'. The 'Redirect Sets' section has a placeholder for uploading a file. Green checkmarks are placed on the 'Name' field, 'Domain' field, 'Web Proxy' dropdown, 'Service' dropdown, 'SSL Certificate' dropdown, 'Allow all' rule, and the 'Deny North Korea' rule.

**Note** = We can repeat above process for as many endpoints as needed.

====

At this point we confirmed that below are completed: -

- 1) Code working locally
- 2) Code pushed to a GIT Repo
- 3) Security Settings Configured

**For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN**

## Step 13 = Configure GIT Repo

The screenshot shows the SAP Commerce Cloud interface. On the left, there's a navigation sidebar with sections like Deployment, Environments, Builds, Subscription Resources, and Repository. The Repository section is currently selected, indicated by a blue background. The main area is titled "Repository". It has a "Repository URL \*" field with "ssh:// git@github.com/chennarrrs/CHENNA.git" entered. There are green checkmarks pointing to the "Repository URL" field and the "git@github.com/chennarrrs/CHENNA.git" value.

## Step 14 = Build and Deploying the Code

→ SCCP → Builds → Create [Enter Name & Branch] → Save]

Now system displays status "**Scheduled**" = Means that the build is scheduled to run.

Once the build process starts, the status changes to "**Building**".

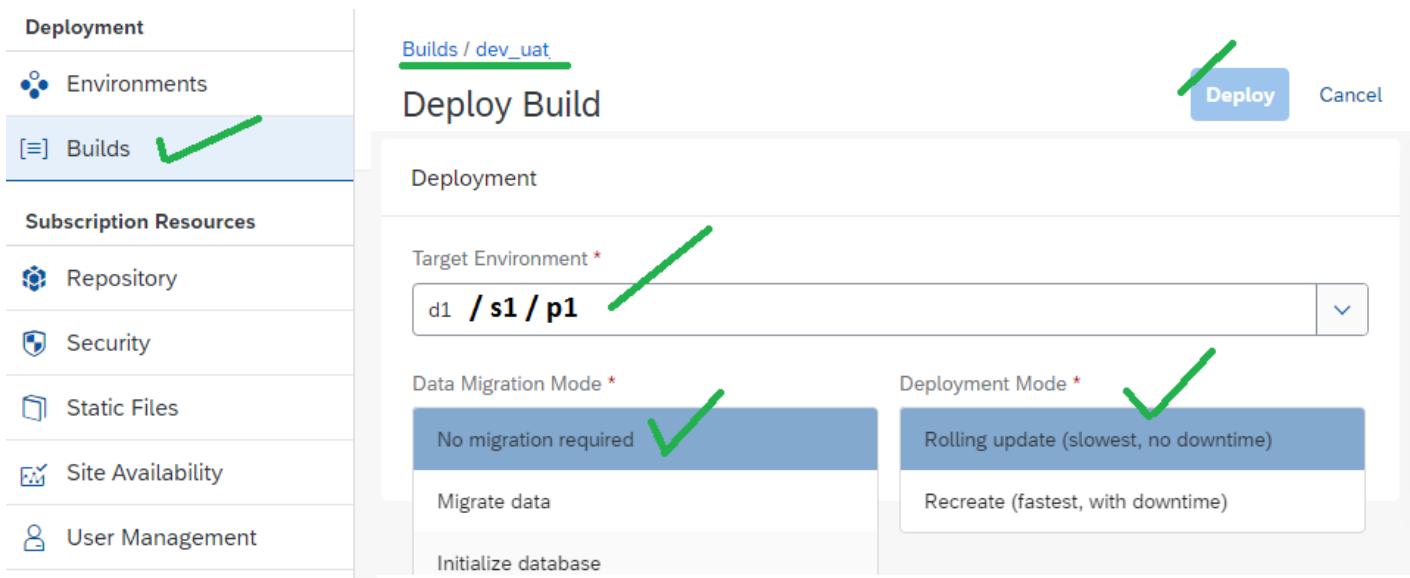
When the build process is complete, the status displays either "**Success**" or "**Failed**".

We can **download build log** by going inside build. Log tells all steps that were performed in build.

The screenshot shows the SAP Commerce Cloud interface with the "Builds" section selected in the sidebar. A "Create Build" dialog is open, showing "Basic Information" with "Name" set to "Chenna RRRS Deploy" and "Git Branch or Tag" set to "develop". Below the dialog, a table lists existing builds: "dev\_uat\_ban..." and "dev\_newpro...". Both builds show a "Success" status and were started on the same day. In the main pane, a build named "dev\_uat" is selected. Its "Basic Information" includes a "Success" status for the last build, source information (develop / 39098 1811.26), and built by user S002195. It also shows deployment details: "d1 D1" and "s1 S1" under "Actively Deployed To", a duration of "17 minutes 5 seconds", and a "Last deployment" status of "Deployed". A "Deploy to Environment" button is visible next to the build name.

==> After build **Success** then we can do the deployment.

→ SCCP → Builds → Go inside success build → Deploy to Environment

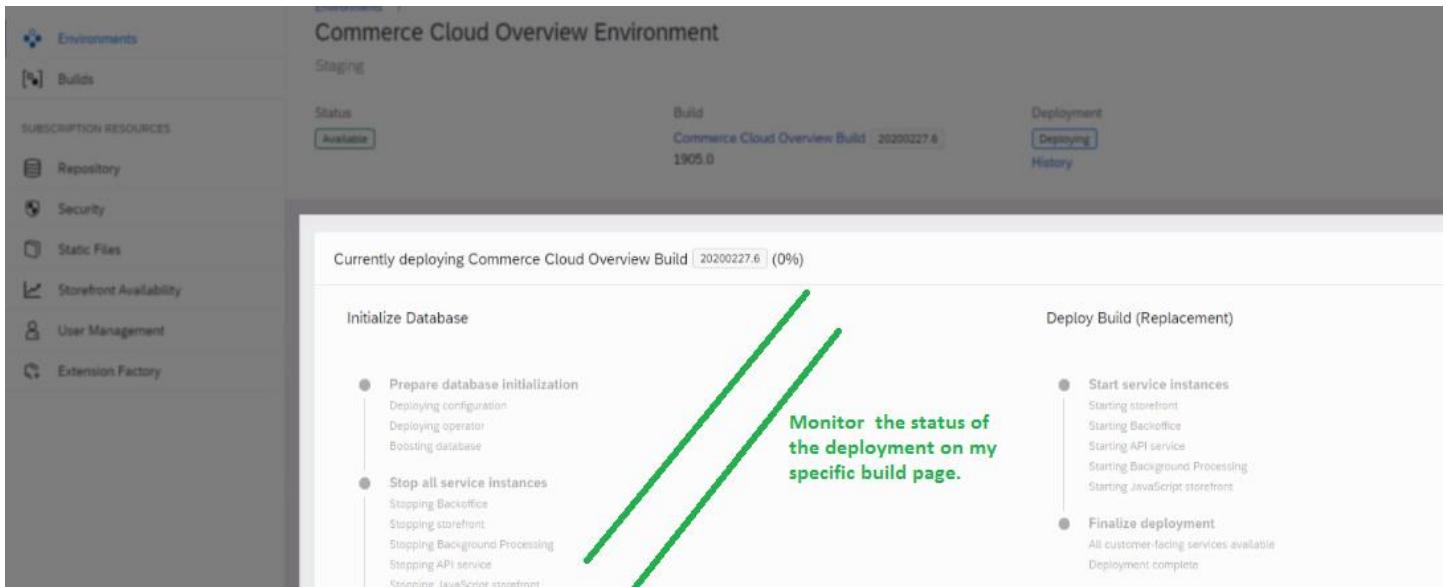


**Q =** What are the Database Actions?

- 1) No Migration required = Select when no DB changes.
- 2) Migrate data = Select when changes in type system.
- 3) Initialize database = Select when 1<sup>st</sup> time deployment.

**Q =** Deployment Methods?

- 1) Rolling update = Slowest & No-downtime
- 2) Recreate = Fastest & downtime [This activates maintenance page on your storefront until the deployment process has completed].



For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

## Step 15 = Accessing Endpoints / Testing the site

**Commerce Cloud Overview Environment**

**Public Endpoints**

Name	Web Proxy	URL	Service
API	Default	SSL api.cyih0n-pseemea1-s5-public.model-t.cc.commerce.ondemand.com	API
Backoffice	Default	SSL backoffice.cyih0n-pseemea1-s5-public.model-t.cc.commerce.ondemand.com	Backoffice
JS Storefront	Default	SSL jsapps.cyih0n-pseemea1-s5-public.model-t.cc.commerce.ondemand.com	JS Storefront

### JS Storefront Results =

Homepage

English \$ USD

SAP

Search here...

DIGITAL CAMERAS FILM CAMERAS CAMCORDERS WEBCAMs ACCESSORIES

SAVE BIG

**Q = How to find backoffice credentials.**

If we have not configured any employees or user groups as part of my initial data setup, then can access Backoffice using the **default admin account**.

Do below to get password for the **default admin account**:-

SCCP → Environments → Select Env (Dev / Staged / Prod) → Services → hcs\_admin → Properties

Environments / Commerce Cloud Overview Environment / Services / hcs\_admin

Status: --

Replicas: --

Last Modified By: 1000000 08/04/2020 10:54 AM

Properties

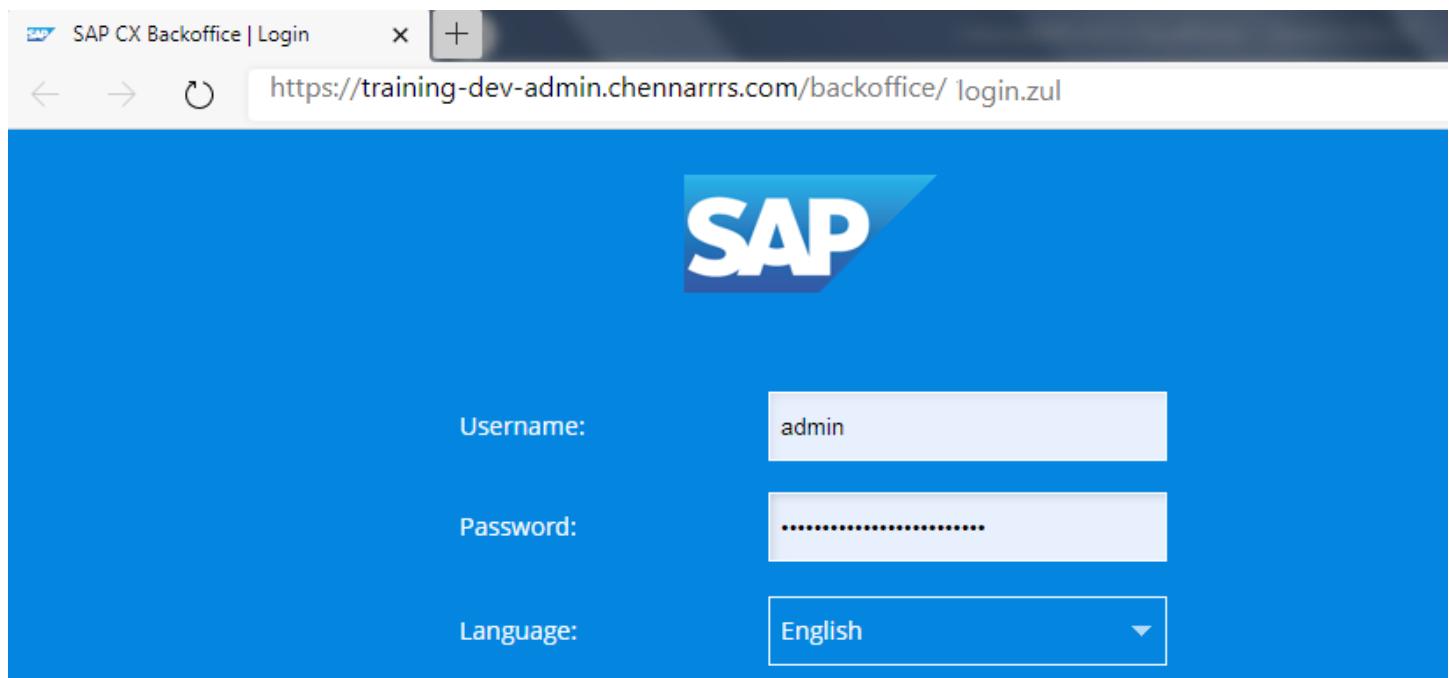
Properties	admin dYgW5tp&P=~nN^~IX\$3Uj\$x&
------------	-------------------------------------

Initial Passwords

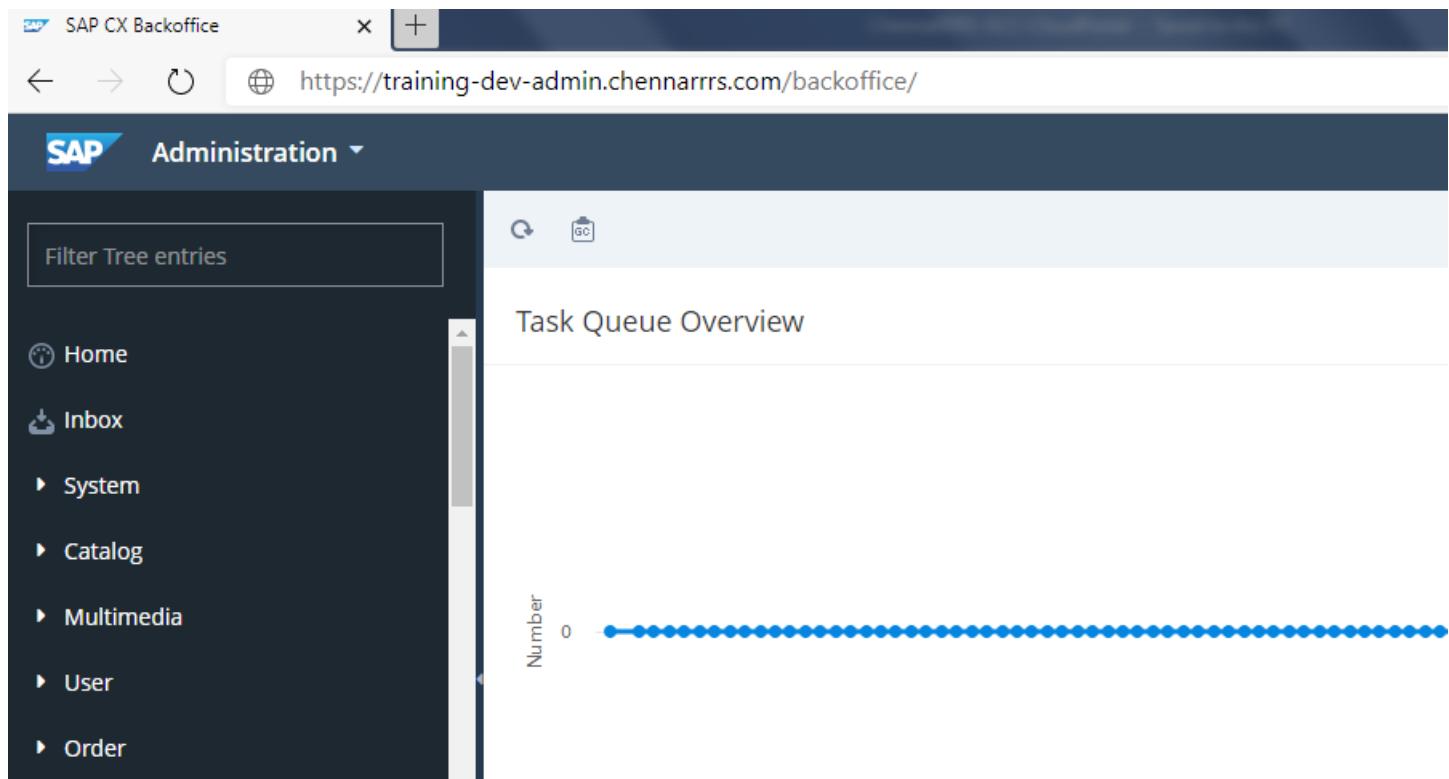
admin	<input type="button" value="Edit"/>
anonymous	<input type="button" value="Edit"/>

For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

## Backoffice Results =



The screenshot shows the SAP CX Backoffice login page. At the top, there is a header bar with the SAP logo and the text "SAP CX Backoffice | Login". Below the header, the URL "https://training-dev-admin.chennarrrs.com/backoffice/ login.zul" is displayed. The main area has a blue background with the SAP logo at the top center. It contains three input fields: "Username" with the value "admin", "Password" with a masked value, and "Language" set to "English".



The screenshot shows the SAP CX Backoffice administration interface. The title bar says "SAP CX Backoffice" and "Administration". The left sidebar is a navigation tree with the following items: Home, Inbox, System, Catalog, Multimedia, User, and Order. The main content area is titled "Task Queue Overview" and features a horizontal bar chart with a single data point at zero. The chart has a legend with the word "Number" and a value of 0.

**Q = Explain how we can manage “Services Configuration Properties in SCCP”?**

A service is an underlying set of functionalities that supports our application.

**Example** = Storefront service is a service dedicated to serve the storefront web pages.

→ SCCP → Environments → Select Dev / Staging / Prod → Goto Services Section & View All

The screenshot shows the SCCP interface under the 'Environments / d1' tab. On the left, a sidebar lists various deployment components like Environment, Builds, Subscription Resources, Repository, Security, Static Files, Site Availability, User Management, and Extension Factory. The main area is titled 'Services' and contains a table with columns: Name, Last Modified, Changed By, Replicas, and Status. The table lists several services: Background processing, Backoffice, Storefront, API, Datahub, JS Storefront, hcs\_admin, and hcs\_common. The 'Storefront' service is highlighted with two green double-headed arrows pointing to it. A large purple rectangular box surrounds the status column of the entire table. To the right of the table, three categories are defined by horizontal lines: 'Service deployed & running' (top), 'Service deployed but not running' (middle), and 'Service not deployed' (bottom). The 'Storefront' service is categorized as 'Service deployed & running'. The 'API' service is categorized as 'Service deployed but not running'. The other services are categorized as 'Service not deployed'.

Name	Last Modified	Changed By	Replicas	Status
Background processing	29/06/2020 12:24 PM	SAP Internal	2 / 2	Running
Backoffice	29/06/2020 12:24 PM	SAP Internal	2 / 2	Running
Storefront	29/06/2020 12:24 PM	SAP Internal	2 / 2	Running
API	29/06/2020 12:24 PM	SAP Internal	0 / 0	Stopped
Datahub	29/06/2020 12:24 PM	SAP Internal	--	--
JS Storefront	29/06/2020 12:24 PM	SAP Internal	--	--
hcs_admin	29/06/2020 12:24 PM	SAP Internal	--	--
hcs_common	29/06/2020 12:24 PM	SAP Internal	--	--

**Note:** - Each service has its own service details page that displays a summary of the service.

**Example** = Let's open the Storefront service. Now we can see “Replicas, Logs & Properties”.

The screenshot shows the 'Environments / p1 / Services' tab for the 'Storefront' service. The top section displays basic information: Status (Running), Replicas (2 / 2 available), and Last Modified By (I000000, 29/06/2020 12:24 PM). Below this, there are tabs for Replicas, Manual Operation History, and Properties, with 'Replicas' currently selected. The 'Replicas' section shows a table with columns: Name, Status, and Logs. It lists two replicas: 'accstorefront-5f5c58c8c9-fjm6f' and 'accstorefront-5f5c58c8c9-hjld4', both in 'Ready' status with a small log icon.

Name	Status	Logs
accstorefront-5f5c58c8c9-fjm6f	Ready	[Log Icon]
accstorefront-5f5c58c8c9-hjld4	Ready	[Log Icon]

**Scenario = How to change a runtime property for our Storefront service.**

**Step 1 = Search with Firm**

**Step 2 = Search Results Page showing 138 Products in many Pages [Each page default - 20 Products]**

**Req = How to display 60 Products in Page?**

**Solution = → SCCP → Env → Select Dev / Staging / Prod → Goto Services Section & View All**

**Environments / p1 / Services**

**Storefront**

**Status**: Running

**Replicas**: 2 / 2 available

**Properties**

**storefront.search.pageSize.Desktop=60**

**Results =**

**Step 1 = Search with Firm**

**Step 2 = Search Results Page Showing 138 Products & We have only 3 Pages [60 Products in Page]**

**For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN**

## Scenario = Monitoring Logs in “SAP Commerce Cloud”?

Be aware that, there's no direct access to the servers in Commerce Cloud.

But, for Prod Env, we might need to see logs (or) know where they're located.

That's why we have to integrate with **Kibana** for viewing the logs.

**Kibana** = An industry-standard tool for searching, viewing, and visualizing operational data into the Cloud Portal. **Kibana** provides: -

- Logging data for each Env
- Querying log data
- Filtering log data
- Presenting log data
- Meaningful log format

**Q** = How to access Kibana from SCCP? = Login into SCCP with “**CUSTOMER\_SYS\_ADMIN**” role.

-- SCCP -- Environments -- Select any Env (Dev / Staged / Prod) -- Logging [Public Endpoints]

The screenshot shows the SCCP interface. On the left, a sidebar lists 'Deployment', 'Builds', 'Subscription Resources', 'Repository', 'Security', 'Static Files', 'Site Availability', 'User Management', and 'Extension Factory'. Under 'Deployment', 'Environments' is selected, and 'd1' is active. In the main area, 'd1' is shown as 'Development' with 'Status' 'Available', 'Build' 'dev\_uat', and 'Last deployment' '1811.26'. A 'Deployed' button is present. Below this, a table titled 'Public Endpoints' lists two entries: 'API' and 'Logging'. The 'Logging' row has its 'URL' field ('SSL logs.cjmr') highlighted with a green box. The 'Service' column for 'Logging' shows 'logging/kibana'.

**Results =**

The screenshot shows the Kibana interface. On the left, a sidebar has 'Discover', 'Visualize', 'Dashboard', 'Timelion', and 'Prometheus' listed. The main area is titled 'Add Data to Kibana' with the sub-instruction 'Use these solutions to quickly turn your data into pre-built dashboards and monitoring systems.' It features four cards: 'APM' (with a graph icon), 'Logging' (with a document icon), 'Metrics' (with a heart and line graph icon), and 'Security anal' (with a shield icon). The 'Logging' card is highlighted with a green box.

For “SAP Commerce Cloud” --- Contact Us = [ChennaReddyTraining@RRRS.CO.IN](mailto:ChennaReddyTraining@RRRS.CO.IN)

21,131 hits

> Search... (e.g. status:200 AND extension:PHP)

**Gives us an unfiltered view of our logging Env.**

**Output below is an aggregate of all webserver, application and Solr logging ordered by time in descending order. Currently showing "Last 15 minutes" logs.**

New Save Open Share Inspect  Auto-refresh <  Last 15 minutes > Options Refresh

Add a filter +

Selected fields

Available fields

fb-\*

Selected fields

? \_source

t \_index

t kubernetes.annotations.ae-version

t kubernetes.annotations.fluentbit\_io...

t kubernetes.docker\_id

t kubernetes.host

@fb\_timestamp

t \_id

#\_score

t \_type

? kubernetes.annotations.csv2\_cx\_sa...

t kubernetes.annotations.csv2\_cx\_sa...

? kubernetes.annotations.checksum/c...

Count

time per 30 seconds

Time \_source

April 17th 2020, 16:16:27.891 @fb\_timestamp: April 17th 2020, 16:16:27.891 log: {"localServerName": "localhost", "remoteHost": "127.0.0.1", "identUserName": "-", "remoteUser": "-", "time": "[17/Apr/2020:15:16:27 +0000]", "requestFirstLine": "GET /healthz HTTP/1.1", "status": "204", "bytes": "-", "referrer": "", "userAgent": "kube-probe/1.16" } stream: stdout time: April 17th 2020, 16:16:27.891 logs.localServerName: localhost logs.remoteHost: 127.0.0.1 logs.identUserName: - logs.remoteUser: - logs.time: [17/Apr/2020:15:16:27 +0000] logs.requestFirstLine: GET /healthz HTTP/1.1 logs.status: 204 logs.bytes: - logs.referrer: - logs.userAgent: kube-probe/1.16 kubernetes.pod\_name: apache2-1gc-788c49966f-cwxt kubernetes.namespace\_name: default kubernetes.pod\_id: 5cadb221-99f2-4a8e-a237-416

April 17th 2020, 16:16:27.600 @fb\_timestamp: April 17th 2020, 16:16:27.599 log: 2020-04-17 15:16:27.599 INFO (qtp33834015-19) [ ] o.a.s.HttpSolrCall [admin] webapp=null path= params={} status=0 QTime=0 stream: stdout time: April 17th 2020, 16:16:27.600 logs.time\_log: 2020-04-17 15:16:27.599 logs.level: INFO logs.msg: qtp33834015-19 logs.class: logs.request: o.a.s.HttpSolrCall logs.message: [admin] webapp=null path= params={} status=0 QTime=0 kubernetes.namespace\_name: default kubernetes.pod\_id: c326c3f5-1228-4a7b-9bc0-a0c4ad5ac409 kubernetes.labels.app\_kubernetes\_io/component: search kubernetes.labels.app\_kubernetes\_io/managed-by: hybris-operator kubernetes.labels.app\_kubernetes\_io/name: solr kubernetes.labels.app\_kubernetes\_io/part-of: hybris ku

New Save Open Share Inspect  Auto-refresh <  Last 15 minutes >

**Time Range**

Quick Relative Absolute Recent

Today	Last 15 minutes	Last 30 days
This week	Last 30 minutes	Last 60 days
This month	Last 1 hour	Last 90 days
This year	Last 4 hours	Last 6 months
Today so far	Last 12 hours	Last 1 year
Week to date	Last 24 hours	Last 2 years
Month to date	Last 7 days	Last 5 years
Year to date		

New Save Open Share Inspect  Auto-refresh <  Last 15 minutes >

**Time Range**

Quick **Relative** Absolute Recent

From April 17th 2020, 16:01:44.622 Set To Now To Now Set To Now

15 Minutes ago 0 Seconds ago

round to the second  round to the second

Go

New Save Open Share Inspect  Auto-refresh <  Last 30 days >

**Important feature is -- Real-time logging is the Refresh Rate. We can configure this in the next pulldown to the left.**

**Refresh Interval**

Off

5 seconds	1 minute	1 hour
10 seconds	5 minutes	2 hours
30 seconds	15 minutes	12 hours
45 seconds	30 minutes	1 day

**Note = There are two Tomcat application server logs for troubleshooting:**

- 1) The standard Catalina console log
- 2) Access log = Useful for connection troubleshooting



**Q = How to see dedicated Tomcat Catalina logs [Console logs]?**

SCCP help Docs provides set of **ready-made filter** queries [For Accessing Apache Tomcat Logs]:-

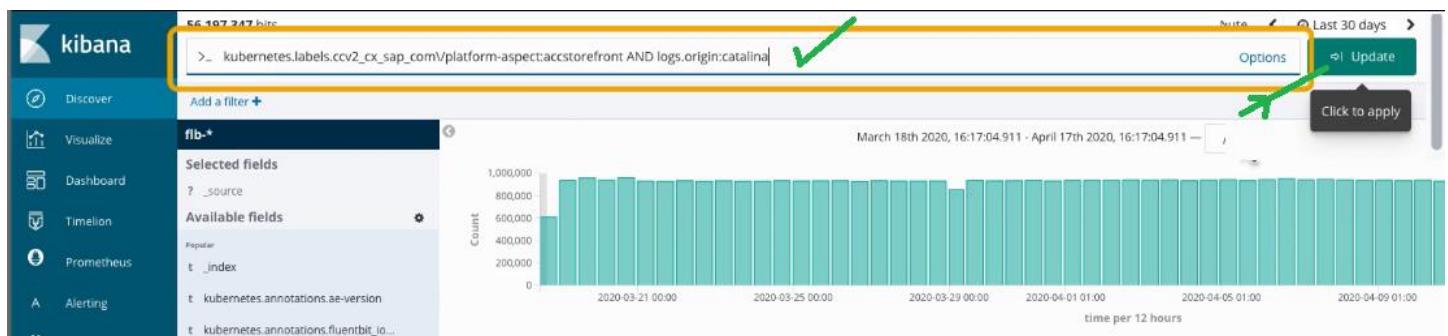
<https://help.sap.com/viewer/0fa6bcf4736c46f78c248512391eb467/v2005/en-US/11d7ed6cc35643a682fe958c0d2a7619.html>

The screenshot shows a SAP Help Docs page for "Accessing Apache Tomcat Logs". The left sidebar has a "Logs" section with "Accessing Apache Tomcat Logs" selected. The main content area lists three queries:

- To see only the access logs, run:  
kubernetes.labels.ccv2\_cx\_sap\_com\platform-aspect:accstorefront AND logs.origin:access-log
- To see only the Catalina logs, run:  
kubernetes.labels.ccv2\_cx\_sap\_com\platform-aspect:accstorefront AND logs.origin:catalina
- To see the application logs together with the Catalina logs, run:  
!(logs.origin:access-log) AND kubernetes.labels.ccv2\_cx\_sap\_com\platform-aspect:accstorefront

A green box highlights the "Copy This Queries" button, and another green box highlights the copy icon in the "Copy This Queries" button's dropdown.

Copy the Query & Paste in Kibana → Update



Queries have the following **basic form**: -

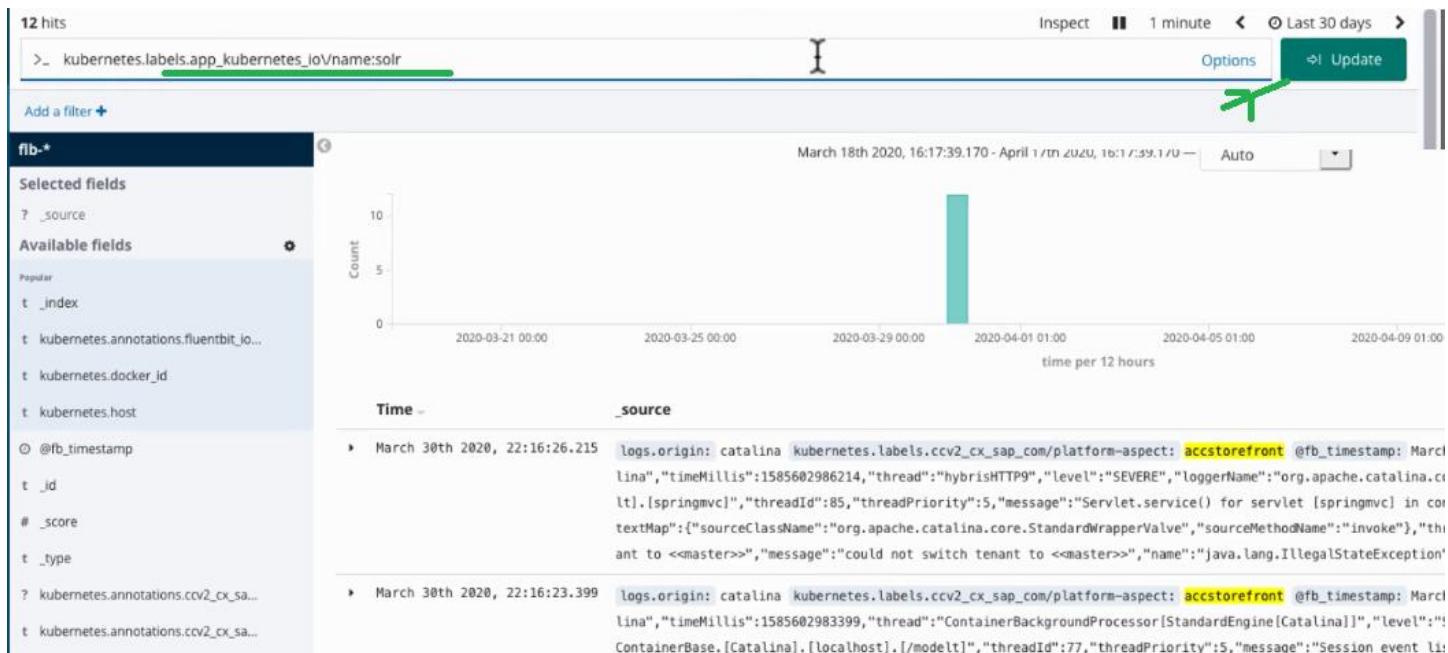
The screenshot shows a SAP Help Docs page for "Kibana Filters and Labels". The left sidebar has a "Logs" section with "Kibana Filters and Labels" selected. The main content area shows the "Label Format" as kubernetes.labels.<label>.<value>. It also displays "Standard Kubernetes Labels" in a table:

Label	Values	Description	Query Examples
app.kubernetes.io/name	hybris solr zookeeper datahub	Name of the application running inside the Kubernetes pod	kubernetes.labels.app_kubernetes_io\name:hybris kubernetes.labels.app_kubernetes_io\name:solr kubernetes.labels.app_kubernetes_io\name:zookeeper kubernetes.labels.app_kubernetes_io\name:datahub

A green arrow points to the "Attribute-Value Pair" in the table header.

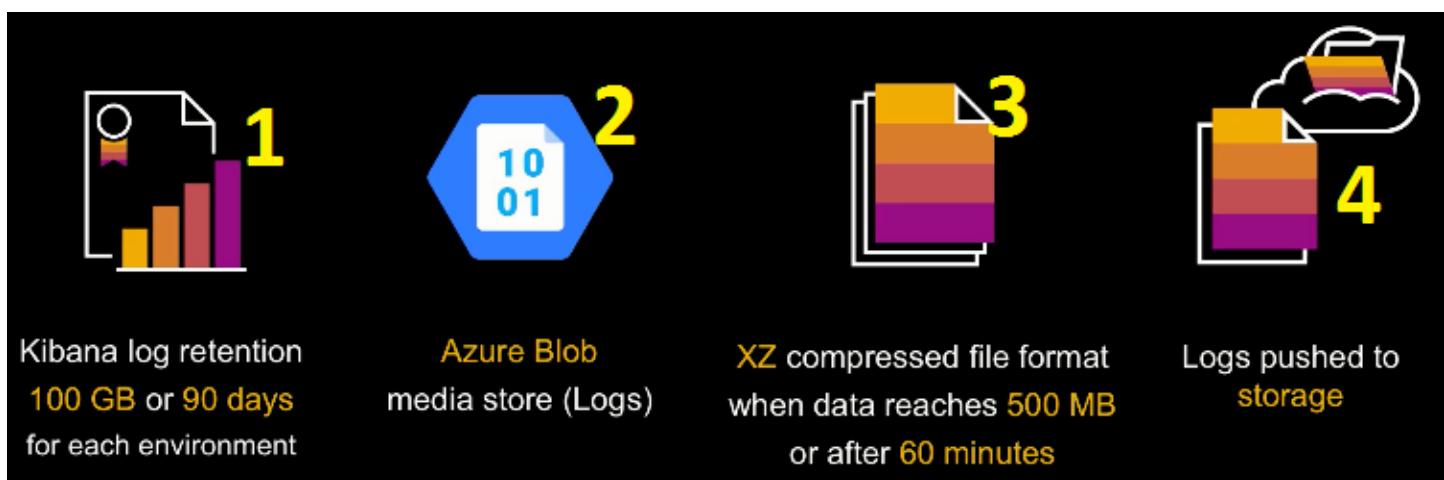
For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

Q = How to see Solr logs? = kubernetes.labels.app\_kubernetes\_io\name:solr



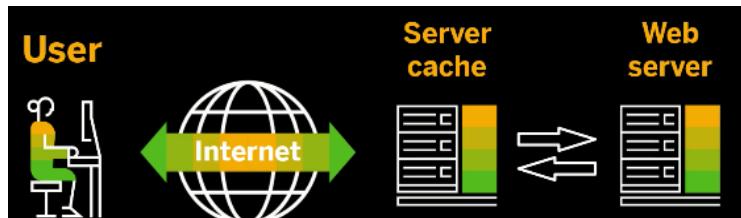
## Conclusions =

- 1) Current log retention in Kibana is limited to 100 GB or 90 days for each environment.
- 2) Longer term log storage is available through your dedicated logs Blob storage.  
Commerce Cloud subscription includes an Azure Blob media store called Logs.
- 3) System saves log data to a **xz-compressed** file format as soon as the data reaches 500 MB, or when 60 minutes have passed.
- 4) It then pushes this file into storage, which means multiple log files appear in the storage.



## Scenario = Enabling / Disabling Web Caching in SCCP?

Web Caching is a mechanism for storing copies of files in a cache so that, cached content can be accessed quickly. This can improve the performance of Commerce website and avoid latency.



→ SCCP [Login as “Customer System Administrator role **or** Customer Developer role”] → Environments → Select the Env [Dev / Staged / Prod] → Open Storefront endpoint

Deployment

Environments

Builds

Subscription Resources

Repository

Security

Static Files

Site Availability

User Management

Extension Factory

Environments / d1

### Edit Endpoint

Configure endpoint for your service.

#### Basic Configuration

Name \* **Chenna RRRS** Site Domain \* **https:// chennarrrs-dev.com**

#### Caching

Default Expire (seconds) 3600 Max Expire (seconds) 86400

Add path **/** Enter URL you want to cache and click on the plus button. Example = /medias ... /\_ui ...

Active paths /medias/ /\_ui/ **+**

We can also remove a specific URL from web caching, on the same Edit Endpoint page click on the “X” icon next to the path.

We entirely disable the web caching applied to our storefront endpoint. For this click on blue slide button next to the “Caching” header.

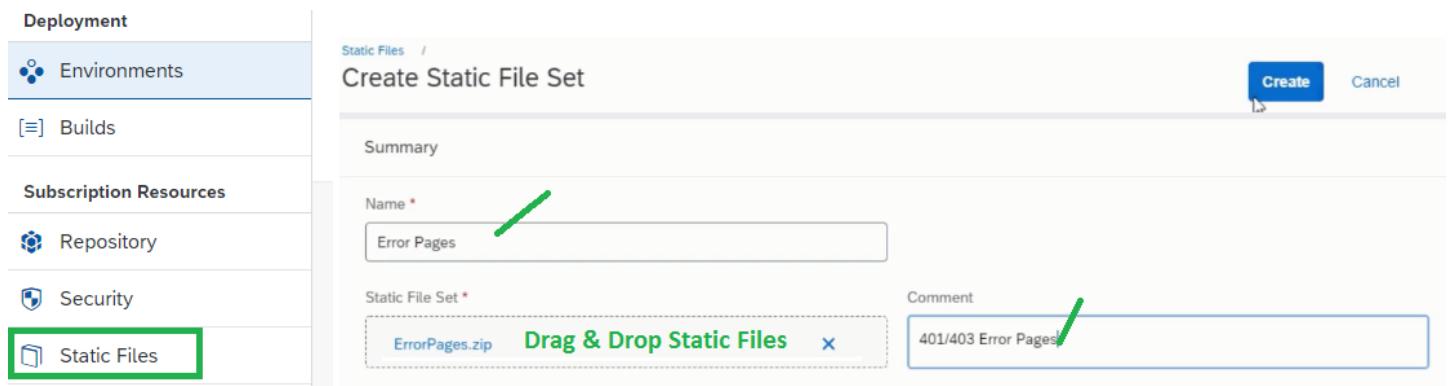
## Scenario = How to manage a Static File set in SCCP?

A Static file can =

- 1) Improves the performance of a website
- 2) Reducing its load and runtimes
- 3) Served by the website in the same fixed form
- 4) Use to set 3<sup>rd</sup> party system secrets, search configurations, sitemaps, and custom error responses.

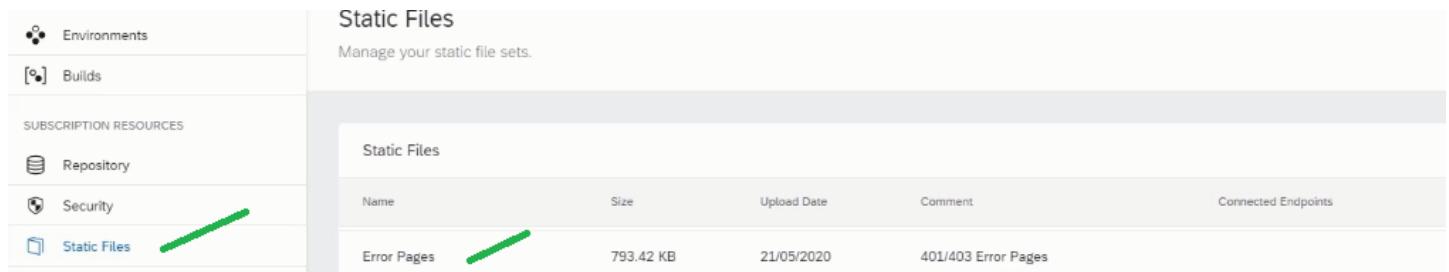
All static files need to be bundled together in a single **ZIP archive up to 1 MB**.

→ SCCP [Login as “Customer Developer role **or** Customer System Administrator role”] → Static Files → **Create** → Enter Name, Drag & Drop Static File Set, Enter Comments → **Create**



The screenshot shows the 'Create Static File Set' dialog. On the left, a sidebar lists 'Environments', 'Builds', 'Subscription Resources', 'Repository', 'Security', and 'Static Files'. The 'Static Files' item is highlighted with a green border. The main form has a 'Summary' section with a 'Name' field containing 'Error Pages' (marked with a green checkmark). Below it is a 'Static File Set' dropdown showing 'ErrorPages.zip' with the text 'Drag & Drop Static Files' overlaid in green. To the right is a 'Comment' field with '401/403 Error Pages'. A blue 'Create' button is at the top right.

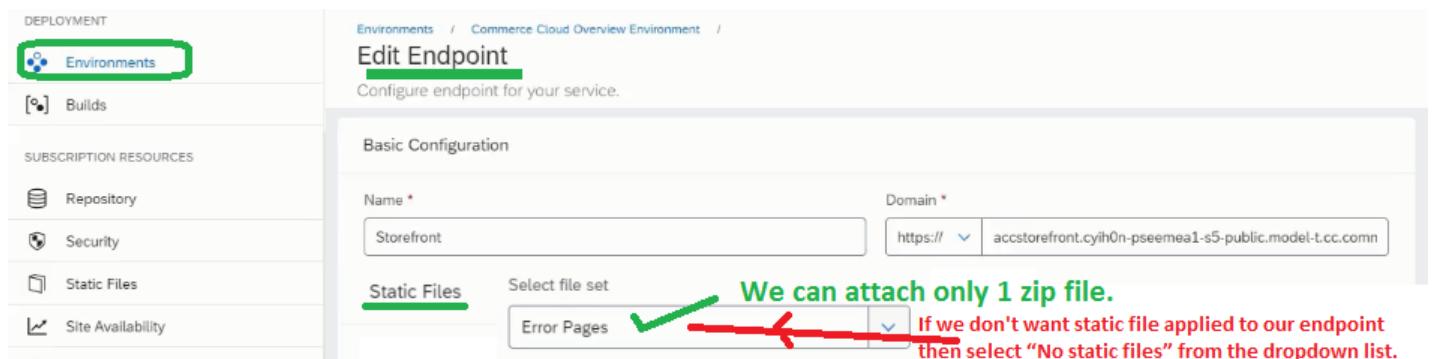
## Results =



The screenshot shows the 'Static Files' list. The sidebar highlights 'Static Files'. The main area displays a table with a single row for 'Error Pages'. The row includes columns for Name, Size (793.42 KB), Upload Date (21/05/2020), Comment (401/403 Error Pages), and Connected Endpoints. A green checkmark is placed over the 'Error Pages' entry in the table.

Now, we need to **associate** Error Pages file set with a **specific endpoint**:-

SCCP -- Environments -- Select the Env [Dev / Staged / Prod] -- Open Storefront endpoint URL



The screenshot shows the 'Edit Endpoint' dialog. The sidebar highlights 'Environments'. The main form has 'Name' set to 'Storefront' and 'Domain' set to 'https://accstorefront.cyih0n-pseemea1-s5-public.model-t.cc.com'. Under 'Basic Configuration', there's a 'Static Files' dropdown set to 'Error Pages'. A green note above the dropdown says 'We can attach only 1 zip file.' A red note below it says 'If we don't want static file applied to our endpoint then select "No static files" from the dropdown list.' A red arrow points to the dropdown menu.

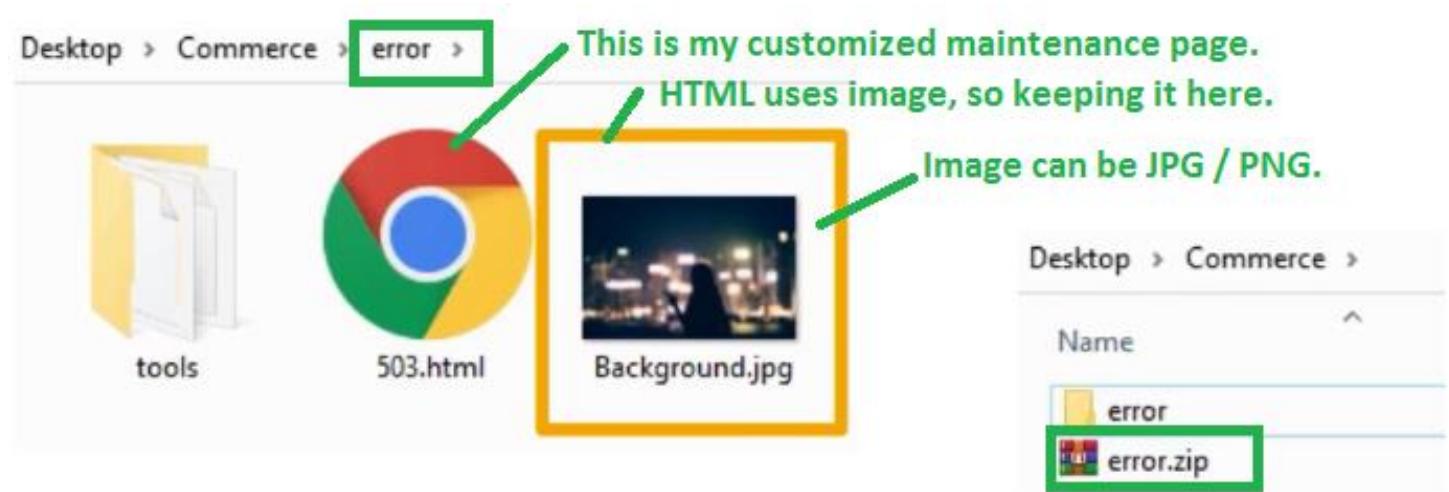
For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

**Scenario** = Managing maintenance pages in the SCCP?

Maintenance pages are used to inform visitors that the site is temporarily unavailable.

Maintenance page can also be used when your site needs to be taken offline for **updates, backups or if the server is temporarily unable** to handle requests.

**Step 1** = Create folder called “**error**” in local Repo [Inside create HTML file and ...]. Finally make error zip file.



**Step 2** = Upload zip file in SCCP

The screenshot shows the SAP Commerce Cloud Control Panel. The left sidebar has navigation links: Deployment, Environments, Builds, Subscription Resources, Repository, Security, and Static Files (which is highlighted with a green box). The main area is titled "Static Files" with the sub-instruction "Manage your static file sets." Below this is a table titled "Static Files". The table has columns: Name, Size, Upload Date, Comment, and Connected Endpoints. A "Create" button is located at the top right of the table. A green checkmark is placed over the "MaintenancePage" row in the table.

Name	Size	Upload Date	Comment	Connected Endpoints
MaintenancePage	✓	793.42 KB		21/05/2020

**Step 3** = Add a customized maintenance page to my storefront endpoint.

-- SCCP [Login as Customer System Administrator] -- Environments – Select the Env [Dev / Staged / Prod] -- Open Storefront endpoint URL.

The screenshot shows the SAP Commerce Cloud interface. On the left, a sidebar lists 'Deployment' options: Environments (selected), Builds, Subscription Resources, Repository, Security, Static Files, Site Availability, and more. In the center, under 'Environments', there's a list for 'd1' (Development) environment: Status (Available), Build (dev\_uat\_1811.26), and Last deployment (Deployed). Below this is the 'Edit Endpoint' screen. It has a 'Basic Configuration' section and a 'Static Files' section. The 'Static Files' section is highlighted with a green bar. Inside, it says 'Select file set' and shows a dropdown menu with 'MaintenancePage' selected. A green callout with the text 'Select the Uploaded Static Files' points to this dropdown.

**Step 4** = Active Maintenance Mode for the same storefront endpoint.

The screenshot shows the SAP Commerce Cloud interface. On the left, a sidebar lists 'Deployment' options: Environments (selected), Builds, Subscription Resources, Repository, Security, Static Files, Site Availability, and more. In the center, under 'Environments', there's a list for 'd1' (Development) environment: Status (Available), Build (dev\_uat\_1811.26), and Last deployment (Deployed). Below this is the 'Public Endpoints' table. It has columns: Name, Web Proxy, URL, and Service. There are two entries for 'Storefront': one with 'Default' and 'SSL accstorefront', and another with 'Default' and 'SSL solutions-dev.i'. To the right of the second entry is a green callout with the text 'Activate maintenance mode'.

**Results** = Hit Storefront URL now & we can see custom page.



For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

**Note:** - If we have multiple country or brand websites running on same Commerce endpoint, then we **can't able to** set up individual maintenance pages or activate maintenance mode for just one of the websites.

In this case, we need to create **separate endpoints** for each of my country or brand websites, so that we can enable maintenance pages for each of them individually.

The screenshot shows a list of storefront endpoints. Each endpoint has a status icon, a name, a 'Default' setting, and an SSL certificate with a URL. The 'FR Storefront' endpoint is highlighted with a yellow border.

Endpoint	Status	SSL Certificate	URL
Storefront	Default	SSL	accstorefront.cyh0n-pseemeal-s5-publ...
FR Storefront	Default	SSL	frstorefront.com
GB Storefront	Default	SSL	gbstorefront.com
PL Storefront	Default	SSL	plstorefront.com

**Note:** - Once maintenance tasks are complete (or) when site is back online after a temporary outage, **maintenance mode needs to be disabled** to let visitors access the site again.

For this – Do below: -

The screenshot shows the same list of storefronts. A context menu is open over the 'FR Storefront' endpoint, with the 'Deactivate Maintenance Mode' option highlighted with a green box and a hand cursor icon.

**Note:** - How to test the website when maintenance page is activated & before deactivating?

The screenshot shows the 'Edit Endpoint' configuration screen. The left sidebar is titled 'Deployment' and shows 'Environments' selected. The main screen shows 'Filtering and Redirects' with an 'IP Filter Sets' section. A green box highlights the 'Allow certain IP's to access website' rule. A red box highlights a note: 'After testing completed, we can change the IP filter range so that our site is again visible for the public.'

For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

## Scenario = Creating / Removing Website Redirects in SCCP?

Website redirects define alternative landing pages for our storefront.

-- SCCP [Login as Customer System Administrator] -- Environments – Select the Env [Dev / Staged / Prod] -- Open Storefront endpoint URL.

The screenshot shows the SAP Commerce Cloud Control Panel. On the left, there's a sidebar with 'Deployment' selected, containing links for 'Environments', 'Builds', 'Subscription Resources', 'Repository', 'Security', 'Static Files', and 'Site Availability'. The main content area is titled 'Edit Endpoint' under 'Environments / d1'. It has a sub-section 'Filtering and Redirects' with 'IP Filter Sets' and 'Redirect Sets'. Under 'IP Filter Sets', there's a 'Base Rule' dropdown set to 'Allow all'. Under 'Redirect Sets', there's a 'Upload from file' section with a file named 'RedirectsSet.txt' and a 'Drag .txt file or Browse' button. A large green checkmark is drawn over the 'Upload from file' area.

**Note:** - Each endpoint supports one redirect set.

**Note:** - We can also remove the redirect sets, when we no longer need redirect traffic to an endpoint [Storefront URL / Some other URL].

This screenshot shows the 'Filtering and Redirects' page. It has 'IP Filter Sets' (Base Rule: Deny all) and 'Redirect Sets' (a file named 'redirectsset.txt' uploaded on 2020-05-22 10:49 UTC). A green box highlights the delete icon next to the file entry.

## Scenario = Access the Dynatrace

**Dynatrace** is a full-stack application monitoring tool that enables to “**observe, track & manage**” the performance of our SCC Environments.

Dynatrace collects **data on** :-

- Java virtual machines [Memory, CPU, Garbage Collector, Thread Count and ...]
- Operating systems [Memory, CPU, Disk, Network and ...]
- Database metrics
- External request information
- Business-side metrics [Conversion rate, bounce rate and ...]
- Client-side end-user actions and performance

Dynatrace **license is included** with our SCC subscription [So no worry in purchase / deploying it]. Dynatrace agents are installed as part of the environment provisioning process, so there's no initial setup involved either.

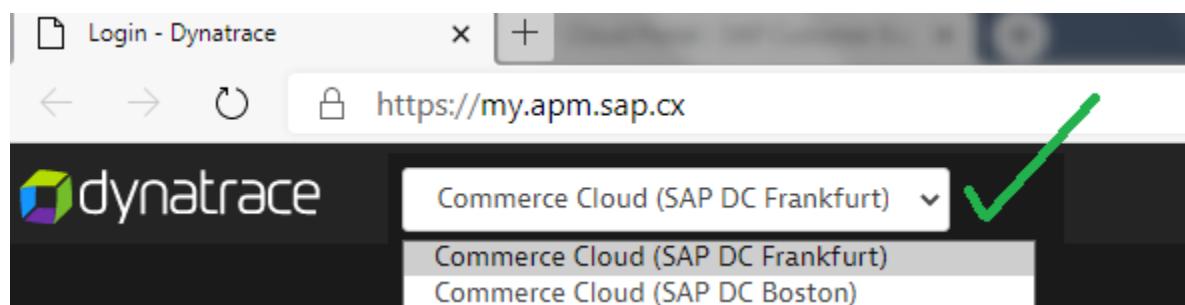
**Q = How to access Dynatrace?** =

-- SCCP [Login with any role] -- Environments – Select the Env [Dev / Staged / Prod] – Monitoring section to see number of challenges

The screenshot shows the SCCP interface. On the left, a sidebar menu includes 'Environments' (selected), 'Builds', 'Subscription Resources', 'Repository', 'Security', 'Static Files', 'Site Availability', and 'User Management'. The main area displays an environment named 'd1' (Development) with status 'Available', build 'dev\_uat 1811.26', and last deployment 'Deployed'. A callout box highlights the 'Monitoring' section, which shows '0 problems' and the text 'Means, No Problems right now.' A green arrow points from this section to a larger callout box titled 'dynatrace' containing links for 'Dashboards and reports', 'Dashboards', 'Create custom chart', 'Reports', 'Analyze', and 'Problems'. To the right, a 'SAP - Business Overview' panel shows 'Environment Health' with 0 problems and 4 host health issues, all marked as 'All fine'.

We can also open Dynatrace at any time using web address URL = <https://my.apm.sap.cx/>

This option works for SCC on both SAP Infrastructure and in the Public cloud.



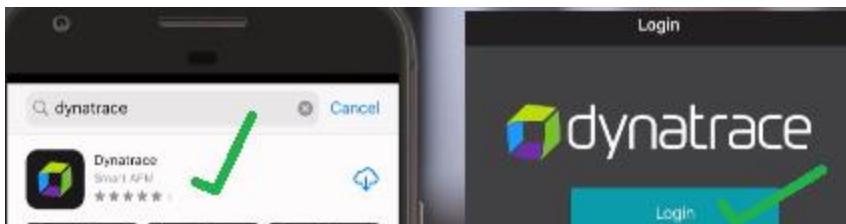
For “**SAP Commerce Cloud**” --- Contact Us = [ChennaReddyTraining@RRRS.CO.IN](mailto:ChennaReddyTraining@RRRS.CO.IN)

## Scenario = Connect a Dynatrace Subscription to a Mobile App

Dynatrace mobile app provides instant notifications about issues detected in SAP Comm Env.

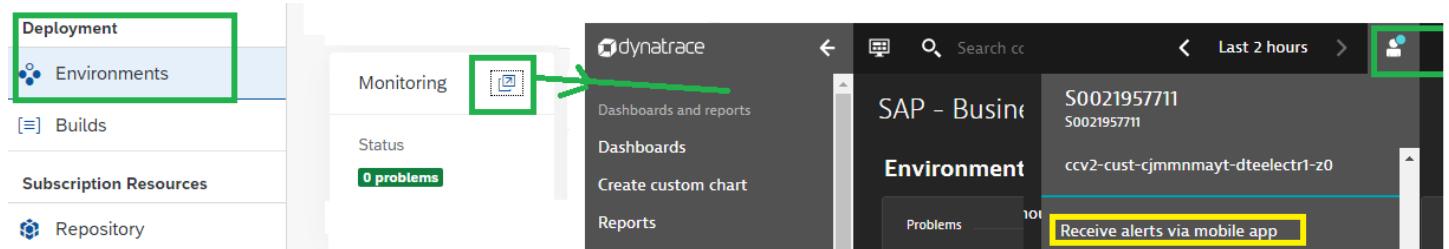
It also provides real-time updates on problems, enabling us to continuously monitor them.

**Step 1** = Download & Install the Dynatrace app on your mobile device & Login.



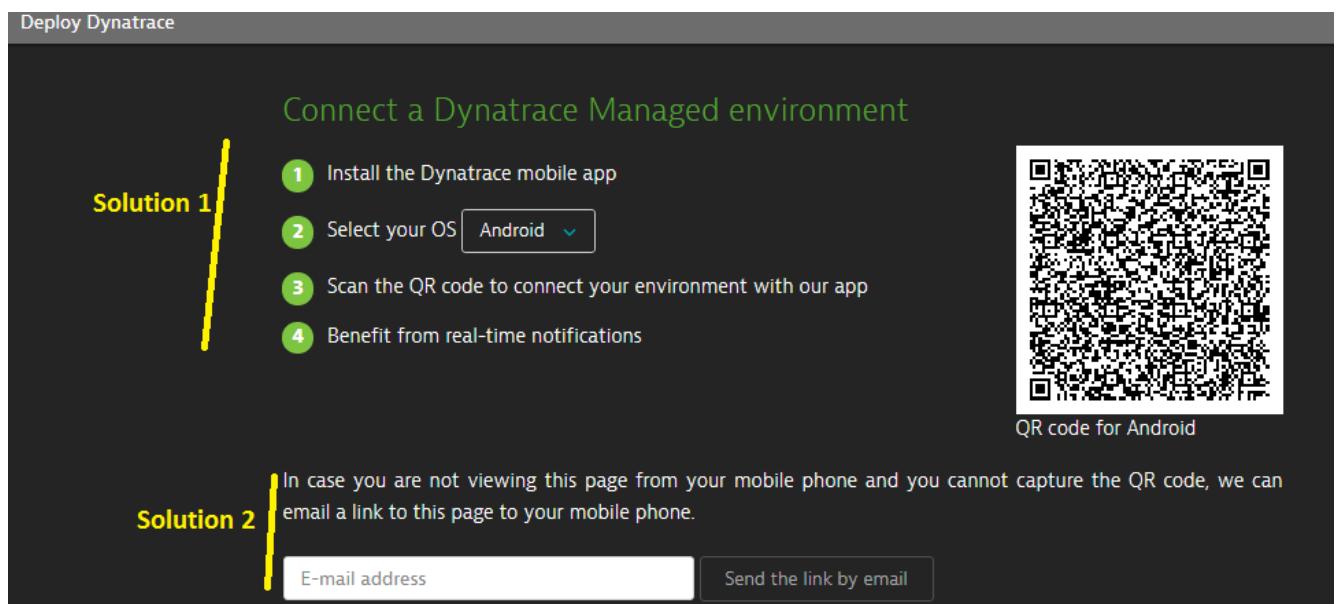
**Step 2** = -- SCCP -- Environments – Select the Env [Dev / Staged / Prod] – Monitoring Section & Click on Right side ICON [This will open Dynatrace].

Inside Dynatrace – User ICON -- Receive alerts via Mobile App



**Step 3** = Follow the instructions for connecting environments to your mobile app.

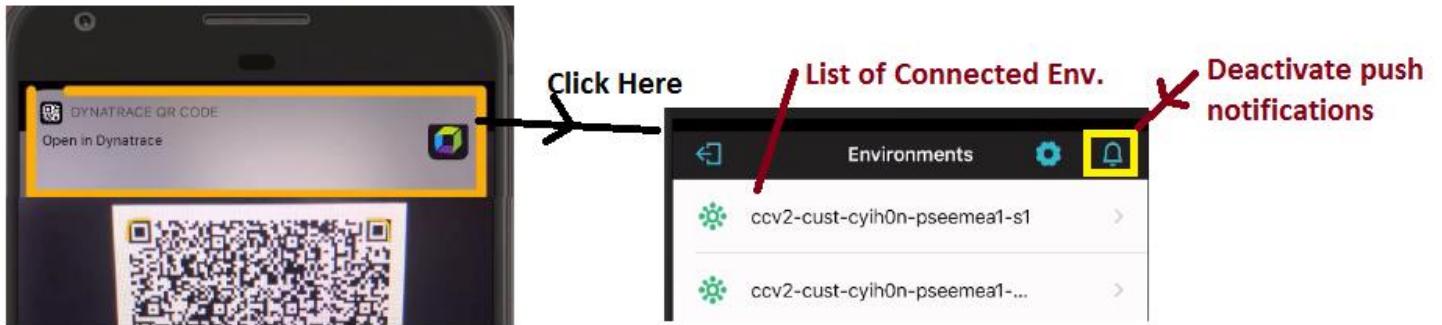
This can be done by **email notification** (or) by **scanning a QR code** displayed on the screen.



If **Solution 2** = Enter email address in this field and click Send the link by email. You then receive a link redirecting you to your monitoring environment in the mobile app.

**For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN**

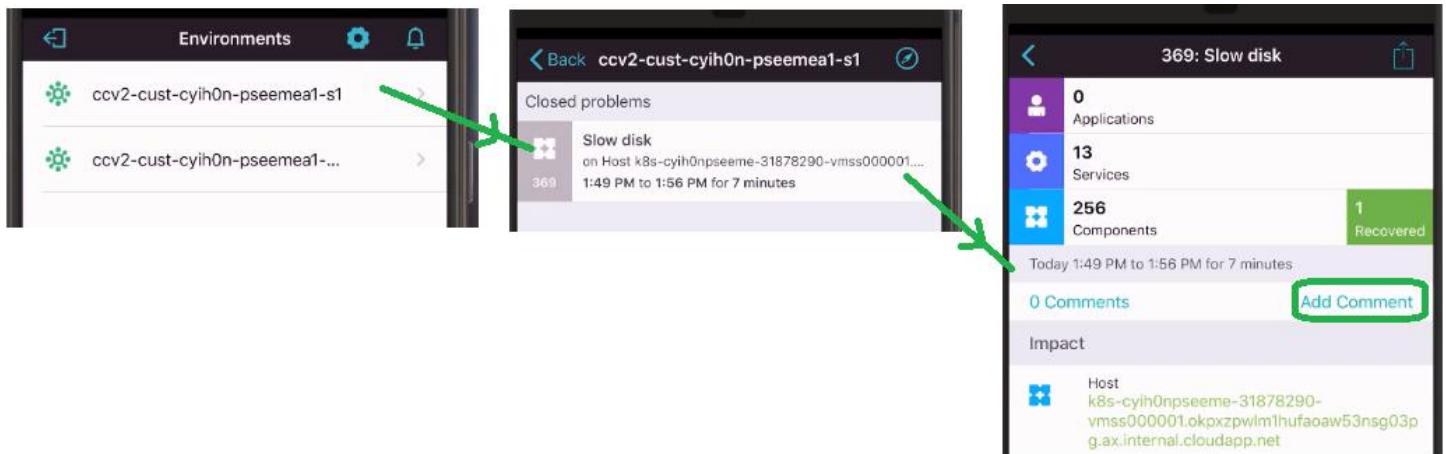
If Solution 1 = 1<sup>st</sup> Select OS & Scan the code in your mobile device.



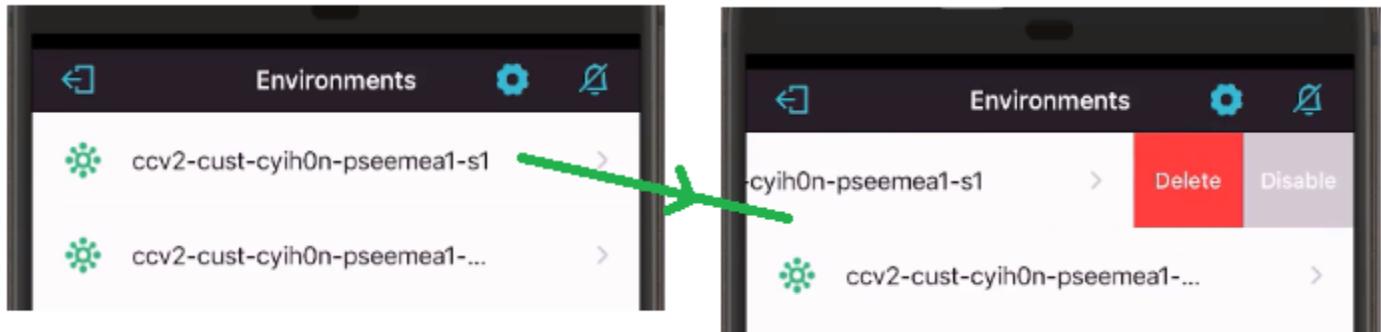
**Step 4** = Now – Select the Env and check the status of reported problems.

App displays details of each problem, including duration, impact, and root cause.

If you want, you can add comments to the problem, or open it in the browser.



**Step 5** = We can deactivate or delete the tenant if we no longer need it in your list.



## Scenario = Create & Customize a Dashboard in Dynatrace

-- SCCP [Login with any role] -- Environments – Select the Env [Dev / Staged / Prod] – Monitoring section to see number of challenges – Click on ICON to open Dynatrace

The screenshot shows the Dynatrace interface. On the left, there's a sidebar with 'Deployment' and 'Environments' selected. An environment named 'd1' is listed as 'Development' with 'Available' status, 'Build dev\_uat 1811.26', and 'Last deployment Deployed'. A 'History' button is also present. The main area shows the 'SAP - Business Overview' dashboard with sections like 'Environment Health' and 'Host health' showing 0 problems. A callout box highlights the '0 problems' status with the text 'Means, No Problems right now.'

## Creating New Dashboard

The screenshot shows the 'Dashboards' section of the Dynatrace interface. On the left, 'Dashboards' is selected in the sidebar. The main area shows a list of dashboards with a 'Create Dashboard' button. A callout box highlights the input field where 'Chenna RRRS Dashboard' is typed, with a checkmark icon indicating the action.

We can search by typing a specific name in search box (or) by tiles using their category type.

The screenshot shows the 'Edit dashboard' screen. On the left, 'Dashboards' is selected in the sidebar. The main area shows the 'Edit dashboard' interface with tabs for 'Tiles' and 'Settings'. Under 'Tiles', there are several categories like 'User behavior', 'User breakdown', 'Top web applications', and 'Application health'. A callout box highlights the 'User behavior' tile with a hand cursor icon.

**Note:** - If the required metric doesn't exist, you can create custom charts.

The screenshot shows the 'Edit dashboard' screen with a 'Custom chart' tile highlighted by a yellow box. The sidebar shows 'Dashboards' selected. The main area shows the 'Edit dashboard' interface with tabs for 'Tiles' and 'Settings', and a 'Custom charts' section.

For "SAP Commerce Cloud" --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

**Example** = Create a dashboard that collects various important metrics used for daily monitoring.

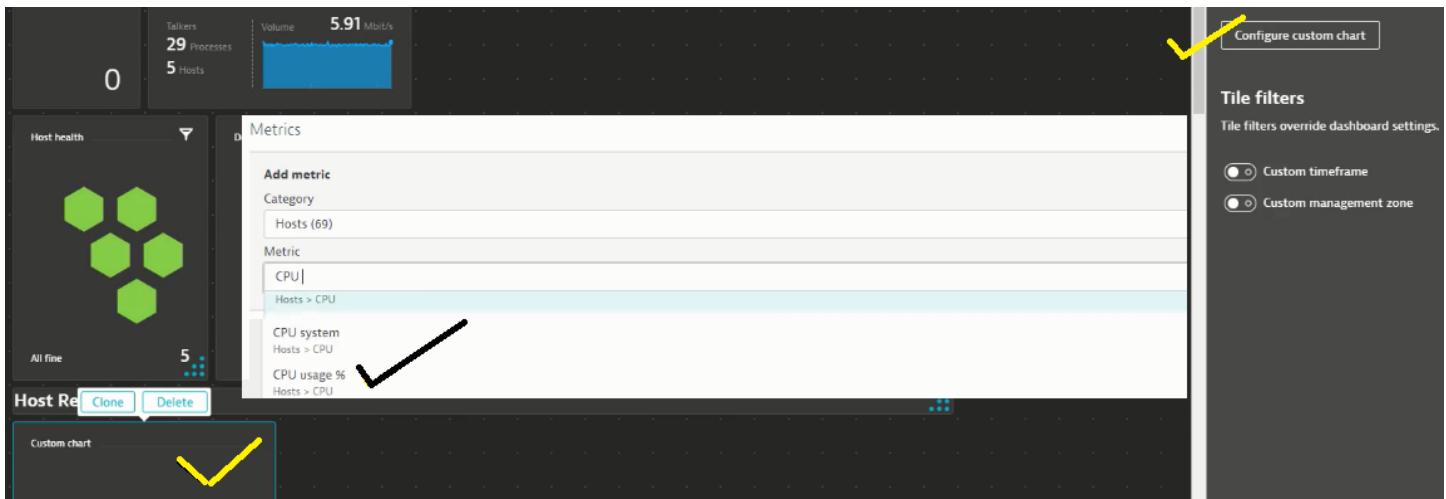
Step 1 = Drop & Drop Header

Step 2 = Set the time frame [= 24 Hrs] for each metric.

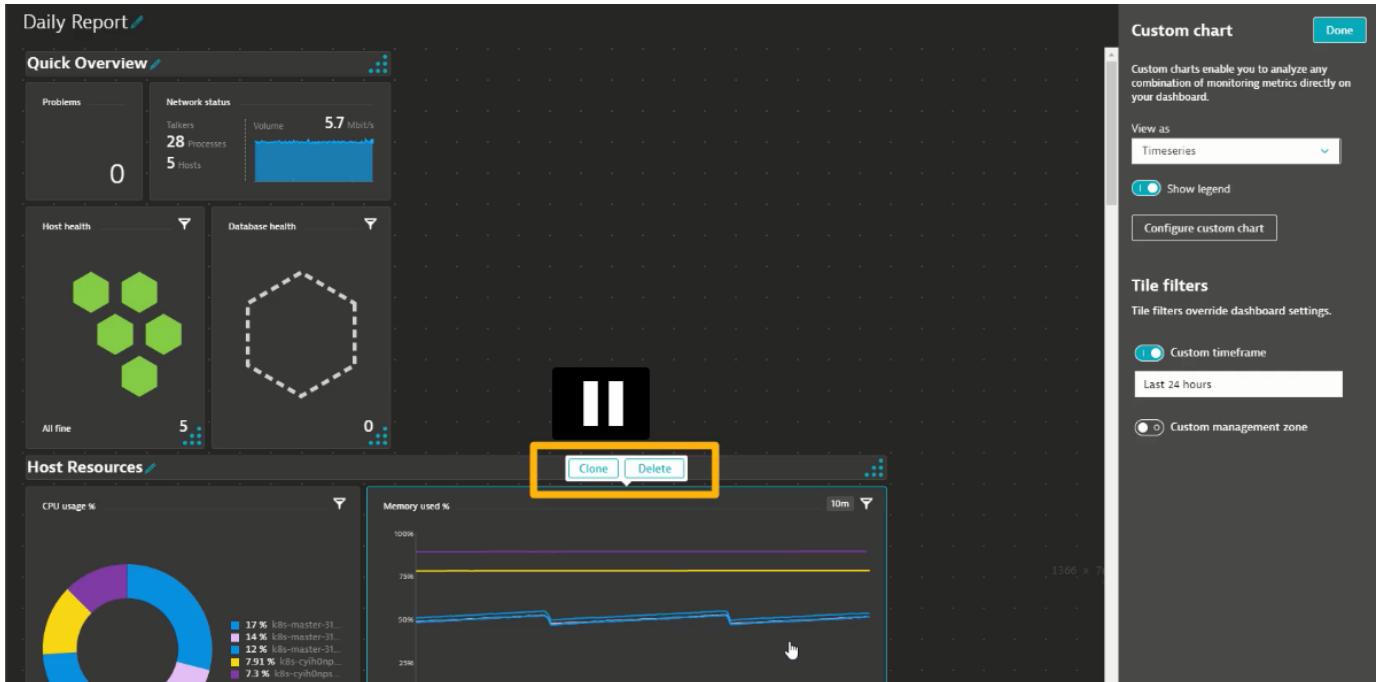
Step 3 = We can add memory & CPU metrics for hosts. These are not in tile catalog, so we will create custom chart.

The screenshot shows the Dynatrace dashboard editor interface. On the left, there's a sidebar with navigation links like Dashboards and reports, Dashboards, Create custom chart, Reports, Analyze, Problems, User sessions, Logs, Smartscape topology, Diagnostic tools, Monitor, Applications, Synthetic, Transactions and services, Databases, Hosts, Network, Technologies, VMware, AWS, and Azure. The main area displays a 'Daily Report' dashboard with tiles for 'Problems' (0), 'Network status' (Tellers: 29, Processes: 5, Volume: 5.91 Mbit/s), 'Host health' (5 hosts, All fine), and 'Database health' (0). A tooltip for the 'Host health' tile provides instructions: 'For each metric, set the timeframe to 24 hours.' A red arrow points from the bottom right of the tooltip to the bottom right of the 'Host health' tile. Another yellow arrow points from the top right of the tooltip to the top right of the 'Edit dashboard' panel. The 'Edit dashboard' panel on the right contains sections for 'Tiles' (Filter tiles, Custom charts, Custom chart with a bar chart icon), 'Settings' (Text, Header, Markdown, Infrastructure, Host health, Network metrics), and a preview area showing the current dashboard layout.

This screenshot shows the same 'Daily Report' dashboard after some changes. The 'Host health' and 'Database health' tiles remain in their original positions. A new 'Host Resources' tile has been added at the bottom left. A yellow arrow points from the bottom right of the 'Host Resources' tile towards the right edge of the dashboard, indicating it is being moved. The rest of the interface is identical to the previous screenshot, including the 'Edit dashboard' panel on the right.



**Note:** - Repeat the steps to create a memory chart. You can also **Clone / Delete** Dashboards.



For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

## Scenario = Submitting an Incident for SCC?

URL = <https://support.sap.com/en/index.html> [Login with S-User credentials].

Before submitting you can search & see if any solution already for your problem.

Welcome to the SAP Support Portal  
Search for SAP Notes, SAP Knowledge Base Articles, SAP Community content  
deploy hybris  Search Access Expert Search

Knowledge Base Search - SAP X https://launchpad.support.sap.com/#/solutions/notesv2/?q=deploy%2520hybris  
SAP Knowledge Base Search Knowledge Base deploy hybris  
Expert Chat Schedule an Expert Submit an Incident  
▼ Search In: SAP Notes & KBAs 347 document(s) found  
SAP Notes & KBAs 347  
SAP Community Wiki 5  
2271470 - Hybris Recommendation  
The BC-Set does not get deployed during the  
CEC-MKT-PRI (Recommendation) 12.02.20

If you do not find any solution, then click on “Submit an Incident”

Launchpad.support.sap.com/#/incident/formats Create Incident Knowledge Base test hybris Create an Incident Alternative Su  
\*Customer Enter Customer Number & S-User ID This information might help you right away Did you know that once we provide you with an  
Customer Number: 0  
S-User ID: 500  
S-User Name: S  
Customer Number: 0001  
Customer Name: 1  
Recently Used My Favorites All Systems  
Is Favorite: System Role: Leading Product: EU Data Processing:  
Search for system ID, number, name, installation, product, or  
Is Favorite System ID System Number System Role Leading Product  
CLO CLO PROD SAP Commerce Cloud, Version 2  
CLO CLO TEST SAP Commerce Cloud, Version 2  
CLO CLO DEVELOP SAP Commerce Cloud, Version 2  
System: CLO Open the system data  
System Details: System Access Data: System Connection: EU Data Processing: System Opener Contact: Enter Contact Name Product: SAP Commerce Cloud, Version 2  
\*System Information Search for a system or select one of your recently used systems from the dropdown list.

Now – Enter “Subject, Description, Language, Component [CEC-HSC (SAP Commerce Cloud)], Attachments, Priority, Contact details and ...”. Finally Review & Submit.

\*Description Please provide us with more information about your issue or question. This detailed information will help to speed up incident processing.  
I give my approval for SAP Support to reproduce the described issue while connected remotely to all environments, including production, using any steps provided by me. This includes steps that might lead to changes being made for the purpose of applying a fix, change, or correction on my behalf. This approval remains valid until my issue is resolved.  
\*Subject: Issues with Commerce Cloud  
Providing additional context will help to speed up incident processing.  
\*Description: Please help me resolve this.  
\*Language: English  
\*Component: CEC-HCS (SAP Commerce Cloud)  
Attachments Submit Save as Draft Cancel

For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

## Scenario = Manage Endpoints in the SCCP?

Without endpoints / URL's, no one can access the code that's been deployed.

-- SCCP [Login as Customer System Administrator] -- Environments – Select the Env [Dev / Staged / Prod] – Create [Enter all below information] → Next

### Step 1 = Create / Basic Configuration

Environments / d1

### Create new Endpoint

Define and configure a new endpoint for your services.

1 Basic Configuration    2 Filtering and Redire...    3 Static Files    4 Caching

Name \* Chenna RRRS Storefront

Domain \* https:// chennarrsstorefront.com

Web Proxy Default (IP 13.68.222.142)

Service \* Storefront [Storefront, JS Storefront, Background Processing, Backoffice, Solr, API]

SSL Certificate \* Let's Encrypt Authority X3, O=Let's Encrypt

Proxy Timeout (seconds) If empty then default value = 180 Sec

### Step 2 = Filtering and Redirects

Environments / d1

### Create new Endpoint

Define and configure a new endpoint for your services.

1 Basic Configuration    2 Filtering and Redire...    3 Static Files    4 Caching

IP Filter Sets

Base Rule IP Filter Sets enable you to manage endpoint access.

Allow all

Add IP Filter Set

Redirect Sets

Upload from file

Drag .txt file or Browse

Previous    Next

## Step 3 = Static Files

The screenshot shows the 'Create new Endpoint' wizard. The current step is 'Static Files' (step 3). A callout box highlights the 'Select file set' dropdown, which contains options like 'No static files' (selected), 'd1\_properties\_hcs\_platform\_common', 'maintenance-page', 's1\_properties\_hcs\_platform\_common', 'p1\_properties\_hcs\_platform\_common', and 'DTE Maintenance Page'. A green checkmark is placed over 'No static files'. Another callout box on the right side of the screen states: 'Can add static files to my endpoint, such as sitemaps, customer error responses, or search configurations.' Navigation buttons 'Previous' and 'Next' are visible at the bottom.

## Step 4 = Enable Caching

The screenshot shows the 'Create new Endpoint' wizard. The current step is 'Caching' (step 4). A callout box highlights the 'Add path' input field, which contains examples: '/medias/' and '/\_ui/'. A green arrow points to this field. Navigation buttons 'Previous' and 'Create' are visible at the bottom.

## Results =

Chenna RRRS Storefront

Default

SSL accstorefront.

Storefront

For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

## Scenario = Self-Service Tenant Provisioning for SCC via SAP C/4HANA Cockpit

SAP C/4HANA Cockpit introduced the new self-service provisioning feature, which took control back into the hands of the system admins.

System admins can use the cockpit to display all the SAP C/4HANA solutions their company has access to right now.

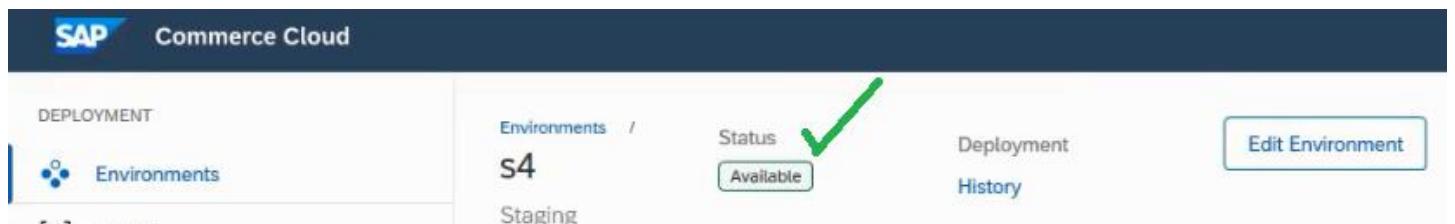
URL = <https://cockpit.cx.cloud.sap/>



This screenshot shows the SAP C/4HANA Cockpit Entitlements page for SAP Commerce Cloud. The left sidebar is identical to the previous screenshot. The main content area is titled "SAP Commerce Cloud" and "Products (1)". Below this, there is a table titled "SAP Commerce Cloud (20)". The table columns include System Number, Role, Status, System Name, System ID, License Information, and Actions. Two rows are visible: one labeled "Productive" with a status of "OPEN" and another labeled "Staging" with a status of "OPEN". Both "OPEN" buttons are highlighted with red boxes. To the right of the table, there is explanatory text and two additional "Provision" buttons, also highlighted with red boxes. The text reads: "1) System admins will also be able to see the systems to which their company is entitled. That is that the company has ordered but which have not yet been provisioned. These systems appear in status open." and "2) When a system admin is ready to start preparing a system for user onboarding, they can use the self-service provisioning functionality to launch the provisioning of the system directly in the cockpit."

if system admin clicks **provision** button, a fully automated SAP provisioning workflow is launched. For SCC, the whole workflow takes about **10 to 15 minutes** to complete.

Once provisioning is complete, system status changes to **active**. Also, URL to access the newly provisioned system is conveniently available in cockpit, providing the system admin with direct SSO enabled access to the system.



For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

## Scenario = Access and Utilize Centralized Logging in SCCP?

### Step 1 = Download “Commerce Cloud Kibana Example JSON” file

The screenshot shows the SAP CX Expert Recommendations interface. At the top, there are tabs for 'EXPERT RECOMMENDATIONS', 'PROJECT FRAMEWORKS', and 'STRATEGIC GUIDANCE'. Below the tabs, a breadcrumb navigation shows 'SAP CX Expert Recommendations / Commerce solutions / Developing SAP Commerce Cloud Locally'. The main content area features a title 'Developing SAP Commerce Cloud Locally' with a sub-section 'Overview'. To the right of the main content, there is a sidebar titled 'Developing SAP Commerce Cloud Locally' with a bullet point: 'Install a local Microsoft SQL Server for SAP Commerce Cloud development in 5 minutes'. Below this, there is an 'Attachments' section containing a link to 'Commerce Cloud Kibana Example.json'.

### Step 2 = -- SCCP [Login with CUSTOMER\_SYS\_ADMIN role] -- Environments -- Select any Env (Dev / Staged / Prod) -- Logging [Public Endpoints]

The screenshot shows the SAP Commerce Cloud Platform (SCCP) interface under the 'Environments' tab. On the left, a sidebar lists various environment-related sections: 'Builds', 'Subscription Resources', 'Repository', 'Security', 'Static Files', 'Site Availability', 'User Management', and 'Extension Factory'. The 'Environments' section is currently selected and highlighted in blue. In the main content area, an environment named 'd1' is selected, with its status shown as 'Available'. Below this, the 'Public Endpoints' section is displayed, listing two entries: 'API' and 'Logging'. The 'Logging' entry has its URL field ('SSL logs.cjmr') highlighted with a green box.

### Results =

The screenshot shows the Kibana interface with the title 'Add Data to Kibana'. On the left, a sidebar lists 'Discover', 'Visualize', 'Dashboard', 'Timelion', 'Prometheus', 'Alerting', 'Dev Tools', and 'Management'. The 'Discover' section is currently selected. The main content area displays three data sources: 'APM', 'Logging', and 'Metrics'. Each section includes a brief description and a 'Add [Source Type]' button. The 'Logging' section's 'Add log data' button is highlighted with a green box.

For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

### Step 3 = Refresh the fields 1<sup>st</sup> if you have not used for long

The screenshot shows the Kibana Management interface. The left sidebar has 'Management' selected. The main area is titled 'Index Patterns' with tabs for 'Index Patterns', 'Saved Objects', and 'Advanced Settings'. Under 'Index Patterns', there is a list of index patterns: 'fb-\*' (marked with a star) and '@fb\_timestamp'. A green box highlights the 'fb-\*' entry. On the right, there are icons for creating a new index pattern, deleting, and editing.

### Step 4 = Import Sample JSON File

The screenshot shows the Kibana Management interface with 'Saved Objects' selected in the sidebar. The main area is titled 'Edit Saved Objects' and contains sections for Dashboards, Searches, and Visualizations. A green arrow points from the text 'Import Sample JSON File which we downloaded in Step 1.' to the 'Import' button in the top right corner. Other buttons shown are 'Export Everything' and 'Delete'.

### Step 5 = See the results of loaded Dashboards & apply filters.

The screenshot shows a Kibana dashboard titled 'Editing Dashboard - Console Logs'. It features a search bar at the top with a green box around it. Below the search bar are two visualizations: a 'Table - Information' card and a 'Pie - Log Level' card. The pie chart shows log levels: INFO (100%), DEBUG (0%), and WARN (0%). At the bottom, there is a time range selector set to '30 seconds' with a yellow box around it, and a search bar with the placeholder 'Uses lucene query syntax'.

For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN

**Q = Explain CCv2 “Data Backup and Restore”?**

<https://help.sap.com/viewer/0fa6bcf4736c46f78c248512391eb467/v2011/en-US/f777fc0a4c524e7aa20d84fc8a55b171.html>

**Step 1 = Go to CCv2 Portal -- Source System [Prod = P1] & Create Data Backup**

The screenshot shows the SAP Commerce Cloud CCv2 Portal interface. On the left, a sidebar menu includes Deployment, Environments, Builds, Subscription Resources, Repository, Security, Static Files, and User Management. The 'Environments' section is highlighted with a blue box. Inside, 'p1 Production' is selected and highlighted with a green box. The main content area displays 'Status Available', 'Build rel-oct14 (20211014.3)', 'Last deployment Deployed', and 'History'. Below this are sections for 'Cloud Storage', 'Data Backups', and 'Hot folders'. A large green arrow points from the 'Create' button in the 'Data Backups' section towards the bottom right.

The screenshot shows the 'Create Data Backup' dialog box. It has tabs for 'Basic Information' and 'Advanced Options'. The 'Basic Information' tab is active, showing a 'Description \*' field containing 'Prod DB Copy', which is highlighted with a green box and has a green arrow pointing to it. The dialog also includes a 'Create' button and a 'Cancel' button.

**Step 2 = Move Data Backup to Target Env [Staged = S1]**

The screenshot shows the SAP Commerce Cloud CCv2 Portal interface. The sidebar menu is identical to the previous screenshot. In the main content area, the 'Data Backups' section is highlighted with a green box. It lists a single backup entry: 'Snapshot p1 . 18T21:35:38.000Z' with 'Build rel-jun1' and 'Status Created'. To the right, a 'Restore to Environment' section shows 'd1' and 's1' (highlighted with a green box) under 'BeforeSnapshot'. A green box highlights the 's1' selection. A green arrow points from the 's1' selection towards the 's1' entry in the 'BeforeSnapshot' dropdown.

**For “SAP Commerce Cloud” --- Contact Us = ChennaReddyTraining@RRRS.CO.IN**