**Deployment Document**

IBCS Portal/Front end

Requirements:

* Install Apache
* Install Dotnet

Apache will be installed by the systems team; Current Apache version is 2.4.37

Dotnet version will depend on the source code compilation version; Current dotnet version installed is 7.0.116

Note: IT team will install both because internet is required for installation

Steps:

1. Copy the published build of IBCS Portal to path /var/www/publish.
2. Change permission of /var/www/publish folder and provide read-write access to this folder.

* To give permission right click on the folder, go to properties, select 777 permission and save

1. Update appsettings.json file,

* set ibcs.api parameter to <https://ibcsadmin.nift.pk/api/>. This is the IBCS API middleware that is working on 10.20.32.181 server which is communicating with FE server on port 443
* set ibcs.img parameter to <https://icl.nift.pk:9000/api/>. This is Imgretriever API which is used for accessing hub icl files on portal. Communicating on port 9000.
* set LogPath parameter to /var/www/publish/IBCS\_APPlogs. Log path need to be defined in order to create the logs

1. Add Kestrel parameter in appsettings.json file

"Kestrel”: {

"EndPoints":

{

"Http": {"Url”: http://localhost:5000"}

}

}

This will be localhost site setting on which the app will be configured

1. Now create a Kestrel service file for application in /etc/systemd/system through following command:

**Nano /etc/systemd/system/IBCS\_APP.service**

1. Paste the following code inside **IBCS\_APP.service** file

*[Unit]*

*Description=Example .NET Web API App running on RHEL 8*

*[Service]*

*WorkingDirectory=/var/www/publish*

*ExecStart=dotnet /var/www/publish/IBCS\_Web.dll*

*Restart=always*

*# Restart service after 10 seconds if the dotnet service crashes:*

*RestartSec=10*

*KillSignal=SIGINT*

*SyslogIdentifier=dotnet-example*

*User=apache*

*Environment=ASPNETCORE\_ENVIRONMENT=Production*

*[Install]*

*WantedBy=multi-user.target*

1. Run the following commands to start the service

**Systemctl enable IBCS\_APP.service**

**Systemctl start IBCS\_APP.service**

1. After the following steps check the status of the IBCS\_APP service

**Systemctl status IBCS\_APP.service**

The status should be Active (running) state and will be showing

1. Now we need to configure apache web server and SSL for IBCS application

Add the following settings in /etc/httpd/conf.d/ssl.conf

*<VirtualHost \*:443>*

*ProxyPreserveHost On*

*ProxyRequests Off*

*ProxyPass / http://localhost:5005/*

*ProxyPassReverse / http://localhost:5005/*

*ServerName ibcs.nift.pk*

*ServerAlias ibcs.nift.pk*

*SSLEngine on*

*SSLCertificateFile "/etc/httpd/conf/ibcs\_nift\_23-24\_1.pem"*

*SSLCertificateChainFile "/etc/httpd/conf/ibcs\_nift\_ca\_bundle.pem"*

*SSLCACertificateFile "/etc/httpd/conf/NIFTECHCA.pem"*

*SSLCARevocationFile "/etc/httpd/conf/NIFTECHCAR310112023.pem"*

*SSLCARevocationCheck leaf*

*SSLVerifyClient require*

*SSLVerifyDepth 2*

*RequestHeader add Subj\_DN "%{SSL\_CLIENT\_S\_DN}e"*

*RequestHeader add Certif "%{SSL\_CLIENT\_M\_SERIAL}s"*

*RequestHeader add Cert\_valid "%{SSL\_CLIENT\_V\_END}e"*

*Header set X-Frame-Options: "SAMEORIGIN"*

*Header always set Strict-Transport-Security "max-age=31536000;includeSubDomains*

*</VirtualHost>*

Place the SSL files and provide path of the SSLCertificateFile, SSLCertificateChainFile, SSLCARevocationFIle and SSLCACertificateFile in the SSL.conf file inside virtualhost 443 area

1. Now finally restart the apache service with the following command:

**Systemctl restart httpd**

1. Once the build is deployed you need to test the production portal which accessible by Faisal khan and RTGS team. Load the link <https://ibcs.nift.pk> and login. Test the changes deployed on the specific pages.
2. Note that the headers play important role in apache configuration in IBCS application.

RequestHeaders are added in apache server in order to get the information from the client certificate and save them on the server headers.

After that the request headers are fetched in the IBCS FE application login page

1. Request Headers include **SSL\_CLIENT\_S\_DN** & **SSL\_CLIENT\_M\_SERIAL.** Both keys are stored in the WAF settings configurations which are done by the Network team.
2. These both headers include the client certificate information which are used in the IBCS portal application

Update SSL certificate:

In order to change SSL files in future just copy the files in the following path /etc/httpd/conf and rename them with the existing SSL files. Following files are replaced

* Cert file *ibcs\_nift\_23-24\_1.pem*
* Cert chain file *ibcs\_nift\_ca\_bundle.pem*
* CA cert file *NIFTECHCA.pem*
* CA revocation file *NIFTECHCAR310112023.pem*

Roll Back Process:

In order to roll back in case of deployment failure just rename the previous version build folder to “publish” and restart the service with this command “Systemctl restart IBCS\_APP.service” in order to get impact of the build.

Common Errors and Resolution:

* If the relevant port in not added in apache httpd then HTTPD service will not run and give error. In order to configure application on HTTP on specific port for example 9001, you need to define port 9001 in /etc/httpd/conf/http.conf file as “Listen 9001” in order to allow port on apache.
* If ports are conflicting lets say you have added same ports against multiple applications then the httpd service will also not run. You may need to switch ports and add a new port

In order to check existing http ports, run the following command

**semanage port -l | grep http\_port\_t**

In order to add new ports, run the following command

**semanage port -a -t http\_port\_t -p tcp 9001**

* If you need to add https then you need to add port in ssl.conf file at path /etc/httpd/conf.d. Remember don’t add ports in both files

http.conf file is for HTTP ports

ssl.conf file is for HTTPS ports

* In order to check detailed errors, run the following command

**journalctl -f**

* If the dotnet service failed to start simple run command

**Systemctl status ibcs\_app.service**

Detailed errors will be displayed which will help to resolve the issue. Most probably there is an error in appsettings.json file format. In case of one more issue you will need to check the root path of the application is mapped correctly in Service at /etc/systemd/system where the published build is placed /var/www/publish