**Mental Health Checkup Installation Guide v1.0.11**

[Introduction](#_bookmark0)

[Service and Data Dependencies](#_bookmark1)

[Infrastructure and Service Dependencies](#_bookmark2) [Data Dependencies](#_bookmark3)

[Property Configurations](#_bookmark4)

[Global Environment Variables in Consul](#_bookmark5)

[Staff and Veteran Environment Variables in Consul](#_bookmark6) [Application Install Instructions](#_bookmark7)

[Core Installation Instructions:](#_bookmark8) [Database Pre-population](#_bookmark9)

[Additional Installation Instructions Documentation](#_bookmark10) [Launchpad Script](#_bookmark11)

[Validation](#_bookmark12)

[Back-Out Procedures](#_bookmark13)

[Back-Out for First Time Deployment (No Existing Container)](#_bookmark14) [Back-Out Deployment with Existing Container](#_bookmark15)

# Introduction

This document provides instructions to install Mental Health Checkup/MHPRO MT and UI Provider and Veteran components in NextGen environment.

|  |  |
| --- | --- |
| **Service Name** | mhpro-web-veteran, mhpro-web-provider, mhpro-resource-provider, mhpro-resource-veteran |
| **Version** | v1.0.11 |
| **Enclave** | Staff and Veteran |

# Service and Data Dependencies

## Infrastructure and Service Dependencies

API Gateway.

Consul Server and Consul Agent. PPG components.

Shared Services (User Service, MVI).

Containers above must already be up and running inside your Docker Swarm.

Jenkins instance must be running in your swarm e.g.: for MAE Dev-Int, it should be [DNS.URL](http://DNS.URL/ToolRequestPage.aspx?treqid=34239):PORT). Base images should be existed in DTR for application usage (centos7, tomcat, weblogic, etc).

## Data Dependencies

Existing Mongo database must be available with the provided credentials. MVI

Existing database must be available and credentials and IP addresses known for each database that the shared services required.

# Property Configurations

The following fields need to be replaced with values from consul or MIS admin Consul Token - $TOKEN

$ENV= VAMF\_ENVIRONMENT variable Mongo db server in devInt

$CONSUL\_SERVER = consul http address Mongo db password in devInt KEYSTORE\_PASSWORD KEY\_PASSWORD

MVI\_END\_POINT – Admin needs to provide TRUSTSTORE\_PASSWORD

608 is the Vista station number in MAE 007/Dev-Int. Need to be configured for different environments.

## Global Environment Variables in Consul

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***<admin provides>***' http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/global/JWT\_SECRET

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***value***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/map2

/weblogic/ADMIN\_NAME

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***value***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/map2

/weblogic/ADMIN\_PASSWORD

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "-DUseSunHttpHandler=true -Djavax.net.ssl.trustStore=***<path to truststore - usually: /usr/java/jdk1.8.0\_131/jre/lib/security/cacerts">*** -Djavax.net.ssl.keyStore=/opt/map2/mhpro/mvicert.jks -Djavax.net.ssl. keyStorePassword=***<keystore password>*** -Dweblogic.management.configuration.SSLMBean.useservercerts=true -Dweblogic.security.SSL. ignoreHostnameVerify -Djavax.net.debug=handshake -Xms1024m -Xmx1024m -XX:CompileThreshold=8000" http://$CONSUL\_HOST/v1/kv

/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/USER\_MEM\_ARGS

### For each Vista:

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***value***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/vistalink/608/HOSTNAME

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***vistalink/608***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/vistalink/608/JNDI\_NAME

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***608***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/vistalink/608/STATION\_NUMBER

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***value***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/vistalink/608/VISTA\_ACCESS\_CODE

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***value***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/vistalink/608/VISTA\_PORT

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***changeit***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro

/v1/variables/resource/vistalink/608/VISTA\_VERIFY\_CODE

## Staff and Veteran Environment Variables in Consul

***Add for both $TYPE=staff and $TYPE=veteran***

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***admin provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/MONGO\_HOST

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d 'mhprodb' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro

/v1/variables/resource/$TYPE/MONGO\_DBNAME

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***admin provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/MONGO\_USERNAME

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***admin provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/MONGO\_PWD

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***admin provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/MONGO\_PORT

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***admin provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/MONGO\_SPECIALCHAR

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***admin provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/MONGO\_SSL

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '/\*' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/$TYPE/SPRING\_SECURITY\_FILTER\_CHAIN\_URLS

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '\*' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/$TYPE/CORS\_ALLOWED\_URL\_VALUE

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '/opt/map2/mhpro/mvilog.txt' http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/$TYPE/LOG\_MVI\_FILE\_PATH

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***MVI endpoint address***' http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/$TYPE/MVI\_END\_POINT

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d 'YES' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/$TYPE/USE\_MVI

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***mvi provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/KEYSTORE\_PASSWORD

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '/opt/map2/mhpro/cacerts.jks' http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/$TYPE/KEYSTORE\_PATH

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***mvi provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/KEY\_PASSWORD

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***mvi provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/TRUSTSTORE\_PASSWORD

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***java trust store path***' http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/$TYPE/TRUSTSTORE\_PATH

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '***admin provides***' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/JWT\_PUBLIC\_KEY

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '0 0/20 \* \* \* ?' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/CRON\_EXPRESSION

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '0 30 23 \* \* ?' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT

/mhpro/v1/variables/resource/$TYPE/PROVIDER\_CRON\_EXPRESSION

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d 't3://localhost:8080' http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/$TYPE/VISTA\_JNDI\_URL

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d 'MBAD,APPLICATION PROXY' http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/$TYPE/VISTA\_PROXY\_ID

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d 'INFO' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/$TYPE/LOG4J\_LOG\_LEVEL

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d '200MBMH' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro

/v1/variables/resource/$TYPE/SENDERID

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d 'D' http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/$TYPE/MVIPROCESSINGCODE

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT --data-binary "***p12mongocertificateinbase64***" http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/$TYPE/MONGO\_CERT

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT --data-binary "***p12mvicertificateinbase64***" http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/$TYPE/TRUST\_STORE\_CERT

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT --data-binary " ***p12mvicertificateinbase64***" http://$CONSUL\_HOST/v1/kv/appconfig

/$VAMF\_ENVIRONMENT/mhpro/v1/variables/resource/$TYPE/MVI\_CERT

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***value***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/$TYPE/ADMIN\_NAME

curl -H "X-Consul-Token: $CONSUL\_TOKEN" -s -X PUT -d "***value***" http://$CONSUL\_HOST/v1/kv/appconfig/$VAMF\_ENVIRONMENT/mhpro/v1

/variables/resource/$TYPE/ADMIN\_PASSWORD

# Application Install Instructions

## Core Installation Instructions:

For the code repository above, in order to deploy the Mhpro application containers, you will be doing the following steps over VPN or on a jump box where you access the affected services:

1. Clone the Mhpro repository on the v1.0.11 tag
2. git clone <https://DNS.URL/scm/mhpro/mhpro-resources.git>

(MHPRO Resources)

1. git clone <https://DNS.URL/scm/mhpro/mhpro-web.git>
2. Clone the Automation Repository – Follow the automation instructions
3. <https://DNS.URL/projects/DEV/repos/automation/browse>
4. <https://DNS.URL/projects/DEV/repos/automation/browse/jobs>
5. Follow the automation instructions to create Jenkins jobs from step #2 above.
6. Go to Jenkins and verify that the three jobs have been created.
7. Create Nexus repository for the application – “mhpro”.

(MHPRO UI)

1. Create Docker repository in DTR (mhpro-web-veteran, mhpro-web-provider, mhpro-resource-staff, mhpro-resource-veteran).
2. Registering MHPRO services in Consul (mhpro-resource-staff, mhpro-resource-veteran, mhpro-web-provider, mhpro-web-veteran).
3. Execute the –build-job in Jenkins and this will kick off the build image job.
4. If there are no error, kick off –deploy-job and this job will create a container.

## Database Pre-population

Scripts have been developed that will initialize the collections needed by Mental Health Checkup in MongoDB. The scripts can be located in the Mental Health Checkup repositories in Stash at <https://DNS.URL/projects/MHPRO/repos/mhpro-resources/browse/Dockerfile?at=refs%2Fheads%2Frelease%2F1.0>. Once you navigate to this repository, the scripts and data files can be found in the “data” folder. To initialize the collections:

1. Transfer the file named loadMongoData.js MongoDB Linux server.
2. Change your current directory to where the file was transferred.
3. Mongo shell along with a JavaScript file will be used to load the data in to MongoDB. There are 5 pieces of information needed in order to execute the script:
   1. mongoHost – the server name for MongoDB
   2. mongoPort – the port number the Mongo daemon is listening
   3. mongoDbName – the Mental Health checkup database name ‘mhprodb’
   4. userId – the user id created in the database installation script
   5. userPwd – the password for the user id created
4. Execute the script replacing the appropriate parameters in italics below with the values collected above:

### mongodb\_initialize

mongo --ssl --sslAllowInvalidCertificates -u *userId* -p "*userPwd*" *mongoHost*:*mongoPort*/ *mongoDbName* loadMongoData.js

Where loadMongoData.js is here: [https://DNS.URL/projects/MHPRO/repos/mhpro-resources/browse/data/loadMongoData.js?](https://DNS.URL/projects/MHPRO/repos/mhpro-resources/browse/data/loadMongoData.js?at=refs%2Fheads%2Frelease%2F1.0) [at=refs%2Fheads%2Frelease%2F1.0](https://DNS.URL/projects/MHPRO/repos/mhpro-resources/browse/data/loadMongoData.js?at=refs%2Fheads%2Frelease%2F1.0)

# Additional Installation Instructions Documentation

Overall deployment procedure document: [https://DNS.URL/pages/viewpage.action?](https://DNS.URL/pages/viewpage.action?title=NextGen%2Bpipeline%2Bdeployment%2Bof%2Ba%2Bmicro-service&amp;spaceKey=LP) [title=NextGen+pipeline+deployment+of+a+micro-service&spaceKey=LP](https://DNS.URL/pages/viewpage.action?title=NextGen%2Bpipeline%2Bdeployment%2Bof%2Ba%2Bmicro-service&amp;spaceKey=LP)

NextGen Platform SDD: [https://DNS.URLpages/viewpage.action?title=System+Design+Document+%28SDD%29+-](https://DNS.URL/pages/viewpage.action?title=System%2BDesign%2BDocument%2B%28SDD%29%2B-%2BNextGen&amp;spaceKey=WBM)

[+NextGen&spaceKey=WBM](https://DNS.URL/pages/viewpage.action?title=System%2BDesign%2BDocument%2B%28SDD%29%2B-%2BNextGen&amp;spaceKey=WBM)

# Launchpad Script

N/A. Currently, there is no Launchpad in NG $ENV or NG Dev-Int. IAM work is in progress.

# Validation

1. Validate MentalHealthCheckup Veteran
2. https://apigatewayhost/mhc-vet/
3. Select SSOe login
4. Enter credentials
5. MHPRO application should be displayed.
6. Validate MentalHealthCheckUp Provider
7. https://apigatewayhost/mhc-provider/
8. Select SSOi login
9. Enter credentials
10. Select provider station
11. MHPRO application should be displayed
12. Delete the container.
13. Delete the image.
14. Delete the container.
15. Delete the image.
16. Back out the configurations.
17. Back out the data base changes.
18. Run the Jenkins jobs with production code base.

# Back-Out Procedures

## Back-Out for First Time Deployment (No Existing Container)

1. Delete the container.
2. Delete the image.

## Back-Out Deployment with Existing Container

1. Delete the container.
2. Delete the image.
3. Back out the configurations.
4. Back out the data base changes.
5. Run the Jenkins jobs with production code base.