Hexagon
40 E Congress St
Tucson, AZ 85701 USA
www.hexagon.com

HXGN MINEPROTECT OPERATOR ALERTNESS SYSTEM

OAS 7 and OASWeb Installation Manual

Version Control

Date	Version	Author	Description
2023-03-01	1.0	JV	Initial version

Hexagon
40 E Congress St
Tucson, AZ 85701 USA
www.hexagon.com

Contents

1. Introduction

Welcome to the HxGN MineProtect Operator Alertness System (OAS) and OASWeb Install Manual. This document is intended to serve as an installation manual for the newest HxGN OAS 7.2 software release. Upon completion of this install manual, the user will have a fully operational HxGN OAS server. For upgrading an existing server for a previous OAS software version to the 7.2 release, please refer to the HxGN MineProtect Operator Alertness System (OAS) 7.2 Server Upgrade Manual.

2. Requirements

To install a new OAS and OASWeb systems, Hexagon requires a server with Ubuntu Server version 20.04LTS.

If you need assistance to install a new Ubuntu server, please refer to “HxGN OAS 7.2 - Ubuntu Installation Manual.pdf” document, or any on-line documentation that explains how to install Ubuntu server version 20.04 LTS.

Also, the following packages are required for the version 7.2 OAS server. These packages can be acquired on the OAS Technical Release Confluence page: HxGN OAS Technical Release Notes.

Required Packages:

- freetds-0.91.tar.gz
- gfx_6.2-1.tar.gz
- growl.tar.gz
- gvenv
- gvos_extract
- json_parser-0.8.tar.gz
- libav-12.3.tar.gz
- libjpeg-turbo-2.0-4.tar.gz
- libpng16.tar.gz
- libx264-2.0.tar.gz
- proj-4.8.0.tar.gz
- SDL2-2.0.3.tar.gz
- tbstack.tar.gz

The following libraries are required if they are not already installed. If you follow the steps in the “HxGN OAS 7.2 - Ubuntu Installation Manual.pdf” document, you don’t need to install these libraries:

- mosquitto-clients_1.6.9-1_amd64.deb
- mosquitto_1.6.9-1_amd64.deb
- libmosquitto1_1.6.9-1_amd64.deb
- libwebsockets15_3.2.1-3_amd64.deb
- libdlt2_2.18.4-0.1_amd64.deb
- libev4_4.31-1_amd64.deb

For the product release files, it is recommended to use the latest release of OAS available within the Confluence page. For the purposes of this document, version 7.2-8 will be used.

Required Server Files:

- gvos-7.2-8.tar.gz
- opweb-3.10-0.3.tar.gz (optional)
- oasweb-1.1.5.tar.gz

Recommended Client Files:

- gvpc-imx6-7.2+8.sqfs
- gvpc-imx6-7.2-8.tar.gz
- gvpc-x86_64-7.2+8.sqfs
- gvpc-x86_64-7.2-8.tar.gz
- oaslv-imx8-7.2+8.sqfs
- oaslv-imx8-7.2-8.tar.gz

Please, copy all required file into the “guardvant’s” home directory: /home/guardvant/libs

If you copy the files in a different location, remember to change the commands to the libraries’ new location.

3. Configuring the HxGN OAS Server

3.1. Configuring PostgreSQL

First edit the PostgreSQL pg_hba.conf file. All instances of ***“peer”*** should be changed to ***“trust”***. Doing this will remove the requirement to enter a password every time the ***psql*** command is run.

In addition to updating the method for each line in the pg_hba.conf file, it is also recommended to include the IP of the network that will be accessing the database, but with different authentication mode: “md5”. In the example file below, the IP used is for the office network and is highlighted in orange.

guardvant@<*ServerName*>:~$ sudo vi /etc/postgresql/12/main/pg_hba.conf

```
# Database administrative login by Unix domain socket
local      all             postgres                                trust
# TYPE     DATABASE        USER            ADDRESS                 METHOD
# "local" is for Unix domain socket connections only
local      All             all                                     trust
# IPv4 local connections:
host       all             all             127.0.0.1/32            trust
host       all             all             10.0.250.0/24           md5
# IPv6 local connections:
host       all             all             ::1/128                 md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
local      replication     all                                     peer
host       replication     all             120.0.0.1/32            md5
host       replication     all             ::1/128                 md5
```

The second file to update is the PostgreSQL postgresql.conf file. Make sure to uncomment out the ***"listen_address"*** line and ensure the value is set to '***'**.

guardvant@<*ServerName*>:~$ sudo vi /etc/postgresql/12/main/postgresql.conf
#-- **# CONNECTIONS AND AUTHENTICATION** **#--** **# - Connection Settings -** **#listen_addresses = 'localhost'** **# what IP address(es) to listen on;** **listen_addresses = '*'** **# what IP address(es) to listen on;** **# comma-seperated list of addresses;** **# defaults to 'localhost'; user '*' for all**

Once both files have been updated, please restart the PostgreSQL process for the changes to take effect.

guardvant@<*ServerName*>:~$ sudo /etc/init.d/postgresql reload
guardvant@<*ServerName*>:~$ sudo /etc/init.d/postgresql restart

3.2. Creating Dynamic Libraries Links

Create the dynamic library links file in the ld.so.conf.d directory. The content of the file is showed in the following table

guardvant@<*ServerName*>:~$ sudo vi /etc/ld.so.conf.d/oas.conf
 # OAS Library Locations **/opt/local/lib** **/opt/guardvant/lib**

Once the file is correct, please save and quit the file using **:wq**

3.3. Configuring MQTT

The following file must be edited for MQTT to refer to the proper ports. The port listed in this manual for the websockets protocol can be replaced with any currently unused port the user wants to use. However, whichever port is selected, must be informed to site IT to have open on the server's network.

guardvant@<*ServerName*>:~$ sudo vi /etc/mosquitto/mosquitto.conf

The format of the mosquitto.conf file will depend on whether Port 80 (HTTP) or Port 443 (HTTPS) is being used for accessing OASWeb. Please proceed to the upcoming section relevant to the port and protocol currently in use.

3.3.1. MQTT Configuration File for HTTP Site

For sites using Port 80 HTTP protocols for accessing OASWeb, the mosquitto.conf file should include the below information. As mentioned previously, the port 9001 for the websockets protocol can be replaced with any unused port the client wants to use. If using a different port, please replace the 9001 below with the chosen port being used.

```
pid_file /var/run/mosquitto.pid

persistence true
persistence_location /var/lib/mosquitto/

log_dest file /var/log/mosquitto/mosquitto.log

include_dir /etc/mosquitto/conf.d

#OAS Configuration
listener 1883
protocol mqtt
listener 9001
protocol websockets
```

3.3.2. MQTT Configuration File for HTTPS Site

For sites using Port 443 HTTPS protocols for accessing OASWeb, the mosquitto.conf file should include the below information. As mentioned previously, the port 9001 for the websockets protocol can be replaced with any unused port the user wants to use. If using a different port, please replace the 9001 below with the chosen port being used. For the ssl files, please replace the pathway, if required, with the location of the existing ssl cert files used to configure the HTTPS protocol.

```
pid_file /var/run/mosquitto.pid

persistence true
persistence_location /var/lib/mosquitto/

log_dest file /var/log/mosquitto/mosquitto.log

include_dir /etc/mosquitto/conf.d

listener 1883
protocol mqtt
listener 9001
protocol websockets
capath /etc/ssl/certs
certfile /etc/ssl/hexagonmining.pem
keyfile /etc/ssl/hexagonmining.key
```

4. HxGN OAS Software Installation

4.1. OAS software

Once all the required files are transferred onto the OAS server, begin by unpacking the library packages via the following commands.

```
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/freetds-0.91.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/gfx_6.2-1.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/growl.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/json_parser-0.8.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/libav-12.3.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/libjpeg-turbo-2.0-4.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/libpng16.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/libx264-161.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/libx264-2.0.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/proj-4.8.0.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/SDL2-2.0.3.tar.gz -C /
guardvant@<ServerName>:~$ sudo tar -zxf ~/libs/tbstack.tar.gz -C /
guardvant@<ServerName>:~$ cp ~/libs/gvos_extract ~/
guardvant@<ServerName>:~$ cp ~/libs/gvenv ~/
```

After the packages have been extracted, unpack and install the OAS server package.

```
guardvant@<ServerName>:~$ chmod 775 gvos_extract
guardvant@<ServerName>:~$ ./gvos_extract -d libs/gvos-7.2-8.tar.gz
```

```
guardvant@<ServerName>:~$ /opt/guardvant/scripts/gvos_extract libs/gvpc-imx6-7.2-8.tar.gz
libs/gvpc-x86_64-7.2-8.tar.gz libs/oaslv-imx8-7.2-8.tar.gz libs/oasweb-1.1.5.tar.gz
```

After installing of the required software packages, it is recommended to move the release files to the respective directories.

```
guardvant@<ServerName>:~$ mv ~/libs/gvpc* ~/libs/oaslv* /opt/guardvant/releases/.
guardvant@<ServerName>:~$ mv ~/libs/gvos-7.2-8.tar.gz /opt/guardvant/versions/
```

Run the command below to create the necessary links and cache to the recently extracted libraries.

```
guardvant@<ServerName>:~$ sudo ldconfig -v
```

Create the necessary link “oas” pointing to the “guardvant” directory.

```
guardvant@<ServerName>:~$ sudo ln -s /opt/guardvant /opt/oas
```

4.2. Creating the Client Services and Database

To create the services and database, the following two parameters are needed.

- **<ServerName>** - Should match the server host name being installed. Also, you can use the client's name.
- **<ClientName>** - Should match the name or initials of the client. For example, if the mine name is Hexagon Mine, the <ClientName> could be hexagon or hxgn.

In this manual, the server's name is "hxgnoasdemo", and the client's name is "hexagon".
The following command creates the server and database.

guardvant@<*ServerName*>:~$ /opt/oas/scripts/gvinstall create **<ServerName> <ClientName>**
Example: guardvant@hxgnoasdemo:~$ /opt/oas/scripts/gvinstall create hxgnoasdemo hexagon

OAS version 7.2-10 or older have a problem when running the gvinstall script. The following error shows up, but if the error is about the Activation.json file this is a known issue. Please continue with the installation process. Refer to the following figure and table.

```
guardvant@test:~$ /opt/guardvant/scripts/gvinstall create test hexagon
2023-05-16 22:32:07.055:info creating GvDatabase with driver=postgres,database=hexagon,host=localhost,name=postgres,user=gv_admin,password=guard4640
2023-05-16 22:32:07.120:info migrated schema for hexagon
/opt/guardvant/versions/current/db/dict/Activation.json: invalid table
Creation of the database failed. Check the condition of PostgreSQL and reattempt the command manually.

The failing command is:
/opt/guardvant/bin/gv_model load -v -D
             postgres://localhost/hexagon,user=gv_admin,password=guard4640,create
             /opt/guardvant/versions/current/db/*.json
             /opt/guardvant/versions/current/db/*/*.json
             /opt/guardvant/opweb/current/db/*.json

created test for hexagon on test
guardvant@test:~$
```

guardvant@<*ServerName*>:~$ /opt/oas/scripts/gvinstall create **<ServerName> <ClientName>**
guardvant@test:~$ /opt/oas/scripts/gvinstall create test hexagon **2023-05-16 22:32:07.055:info creating GvDatabase with driver=postgres,database=hexagon,host=localhost,name=postgres,user=gv_admin,password=guard4640** **2023-05-16 22:32:07.120:info migrated schema for hexagon** **/opt/guardvant/versions/current/db/dict/Activation.json: invalid table** **Creation of the database failed. Check the condition of PostgreSQL and reattempt the command manually.** **The failing command is:** **/opt/guardvant/bin/gv_model load -v -D** **postgres://localhost/hexagon,user=gv_admin,password=guard4640,create** **/opt/guardvant/versions/current/db/*.json** **/opt/guardvant/versions/current/db/*/*.json** **/opt/guardvant/opweb/current/db/*.json** **created test for hexagon on test**

Once the database has been created, some of the server scripts need to be updated.

4.3. Updating Server Scripts

4.3.1. GVOS Script

After creating the database, the first script to update will be the gvos script created in the previous step. In this file, make sure to comment out the line that defines the GV_ADDRESS variable as well as update the GV_VERSION variable to the current release. In this manual the version is 7.2-8.
Also, make sure you add the $postgresql service next to the $network required service.

guardvant@*<ServerName>*:~$ sudo vi /etc/init.d/**<ServerName>**

```
#!/bin/bash
### BEGIN INIT INFO
# Provides: gvos
# Required-Start: $network postgresql
# Required-Stop:
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Guardvant Application Server
### END INIT INFO

[ -d /opt/oas ] && OAS_ROOT=/opt/oas || OAS_ROOT=/opt/guardvant
client=<ClientName>
server=<ServerName>
iface=none
this_ip=none
virtual_ip=none

export GV_VERSION=7.2-8
export GV_BACKUP=none
export GV_BACKUP_PORT=4600
#export GV_ADDRESS=<ServerName>
export GV_DOCROOT=$OAS_ROOT/clients/$client
```

4.3.2. GVUPLOADS Script

There is a known issue in the gvuploads script locate in the /etc/init.d/ directory. To fix this issue edit the file following the procedure below.
You need to replace the $GV_ROOT reference to $OAS_ROOT.

guardvant@*<ServerName>*:~$ sudo vi /etc/init.d/gvuploads

```
#!/bin/sh
### BEGIN INIT INFO
# Provides: gvuploads
# Required-Start: gvos
# Required-Stop:
# Default-Start: 2 3 4 5
```

```
# Default-Stop: 0 1 6
# Short-Description: Guardvant Video Uploads Manager
### END INIT INFO

[ -d /opt/oas -a ! -h /opt/oas ] && export OAS_ROOT=/opt/oas || export
OAS_ROOT=/opt/guardvant

case "$1" in
   start)
      $OAS_ROOT/scripts/gvuploads.run start
      ;;
   stop)
      $OAS_ROOT/scripts/gvuploads.run stop
      ;;
   restart)
      $OAS_ROOT/scripts/gvuploads.run restart
      ;;
esac
```

4.3.3. RSYNC File

The next file to be updated is the rsync configuration file created for the server. The line defining the address variable needs to either be deleted or commented out. It is always recommended to comment the line rather than deleting it.
Also, in the events section, the path variable must be corrected from:
/opt/guardvant/<ClientName>/events
To:
/opt/guardvant/clients/<ClientName>/events

guardvant@*<ServerName>*:~$ sudo vi /etc/rsync/**<ServerName>.conf**

```
max connections = 200
#address = <ServerName>
log file = /var/log/rsync_<ServerName>.log
timeout = 30

[releases]
      path = /opt/guardvant/releases
      read only = yes
      list = yes
      uid = guardvant
      gid = staff

[events]
      path = /opt/guardvant/clients/<CLIENTNAME>/events
      read only = no
      list = yes
```

```
        uid = guardvant
        gid = staff

[snapshots]
        path = /opt/guardvant/clients/<CLIENTNAME>/snapshots
        read only = no
        list = yes
        uid = guardvant
        gid = staff
```

4.3.4. NGINX Configuration

One of the new features of the version 7 release of gvos includes support for the new OAS online GUI, OASWeb. This section will outline the multiple options for configuring NGINX with the version 7 release.

OASWeb – Single Site over Port 80

guardvant@<*ServerName*>:~$ sudo vi /etc/nginx/sites_available/default

```
server {
        listen 80;
        server_name localhost;
        root /opt/guardvant/web/current;
        location /<ClientName>/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location /gvos/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location / {
                try_files $uri $uri/ @rules;
        }
        location @rules {
                rewrite ^(.*)$ /index.html break;
        }
        location /sites.json {
                alias /opt/guardvant/web/sites.json;
        }
}
```

OASWeb – Single Site over Port 443

guardvant@<*ServerName*>:~$ sudo vi /etc/nginx/sites_available/default

```
server {
        listen 80;
        server_name localhost;
        location /<ClientName>/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location /gvos/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
}
server {
        listen 443;
        ssl    on;
        ssl_certificate    /etc/ssl/hexagonmining.pem;
        ssl_certificate_key    /etc/ssl/hexagonmining.key;
        server_name <ClientName>.hexagonmining.com;
        index index.html;
        root /opt/guardvant/web/current;
        location /gvos/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location /<ClientName>/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location / {
                try_files $uri $uri/ @rules;
        }
        location @rules {
                rewrite ^(.*)$ /index.html break;
        }
        location /sites.json {
                alias /opt/guardvant/web/sites.json;
        }
}
```

OASWeb – Multi-Site over Port 80 (Usually Cloud Servers)

guardvant@<*ServerName*>:~$ sudo vi /etc/nginx/sites_available/default

```
server {
        listen 80;
        server_name localhost;
        root /opt/guardvant/web/current;
        location /<ClientName>/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location /gvos/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location / {
                try_files $uri $uri/ @rules;
        }
        location @rules {
                rewrite ^(.*)$ /index.html break;
        }
        location /sites.json {
                alias /opt/guardvant/web/sites.json;
        }
        location /<Client_2>/ {
                include proxy_params;
                proxy_pass https://<Client_2_IP>/gvos/;
        }
        location /<Client_3>/ {
                include proxy_params;
                proxy_pass https://<Client_3_IP>/gvos/;
        }
}
```

OASWeb – Multi-Site over Port 443

guardvant@<*ServerName*>:~$ sudo vi /etc/nginx/sites_available/default

```
server {
        listen 80;
        server_name localhost;
        location /<ClientName>/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location /gvos/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
}
server {
        listen 443;
        ssl    on;
        ssl_certificate    /etc/ssl/hexagonmining.pem;
        ssl_certificate_key    /etc/ssl/hexagonmining.key;
        server_name <ClientName>.hexagonmining.com;
        index index.html;
        root /opt/guardvant/web/current;
        location /gvos/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location /<ClientName>/ {
                include proxy_params;
                proxy_pass http://localhost:5000;
        }
        location / {
                try_files $uri $uri/ @rules;
        }
        location @rules {
                rewrite ^(.*)$ /index.html break;
        }
        location /sites.json {
                alias /opt/guardvant/web/sites.json;
        }
        location /<Client_2>/ {
                include proxy_params;
                proxy_pass https://<Client_2_IP>/gvos/;
        }
        location /<Client_3>/ {
                include proxy_params;
                proxy_pass https://<Client_3_IP>/gvos/;
        }
}
```

guardvant@<*ServerName*>:~$ sudo /etc/init.d/nginx reload

guardvant@<*ServerName*>:~$ sudo /etc/init.d/nginx restart

4.4. Client Specific Configurations

4.4.1. Loading Database

If available, copy any client specific json configuration files to the client's db directory.
If you need the basic JSON files to install a new server, you can ask for them from your Hexagon contact.

```
guardvant@<ServerName>:~$ scp <REMOTE_PC>:*.json /opt/oas/clients/<ClientName>/db/
```

Once the client's json files are in the /opt/oas/clients/<ClientName>/db/ directory, the configurations for the software release as well as the client specific set-up can be loaded into the database using the gv_admin script.

```
guardvant@<ServerName>:~$ /opt/oas/scripts/gv_admin load <ServerName>
```

This script will generate a diff file of all configurations that are being altered from default. It is highly recommended to review this file and ensure all updates are desired prior to applying the changes.

4.4.2. Site Time Zone

It is recommended to verify the time zone on the server is correct for the physical location of the client. To verify the current time zone, the "date" command can be used.

```
guardvant@<ServerName>:~$ date
Tue Nov 01 12:34:56 UTC 2022
```

To update the date, the link pointing to the current time zone needs to be updated. The following process is required to update the server's time zone.

```
guardvant@<ServerName>:~$ cd /etc
guardvant@<ServerName>:~$ ls -l /etc/localtime
guardvant@<ServerName>:~$ sudo rm localtime
guardvant@<ServerName>:~$ sudo ln -s /usr/share/zoneinfo/<Region>/<City> localtime
```

4.4.3. Sites.json file

The last step is creating the file sites.json in the /opt/guardvant/web/ directory.

The format of this file will be dependent on whether Port 80 (HTTP) or Port 443 (HTTPS) is being used for accessing OASWeb. Please proceed to the upcoming section relevant to the port and protocol currently in use.

The "<Client>" variable will be what is displayed as the client's name on the OASWeb site's tab. The "<SiteURL>" variable should be replaced with the IP Address or URL of the OASWeb interface. Both are highlighted in orange text to signify they need to be replaced with the site-specific information.

Use the following command to create the file.

```
guardvant@<ServerName>:~$ vi /opt/oas/web/sites.json
```

HTTP Site

For sites using Port 80 HTTP protocol for accessing OASWeb, the following formatting is required:

```
{
    “<client>”:{“type”:”Site”,”name”:”<Tab Name>”,”mqtt”:”ws://<Server_IP>:9001”,”url”:”<Server_IP>”}
}
```

HTTPS Site

For sites using Port 443 HTTPS protocols for accessing OpWeb/OASWeb, the following formatting is required.

```
{
    “<client>”:{“type”:”Site”,”name”:”<Tab Name>”,”mqtt”:”wss://<Server_IP>:9001”,”url”:” <Server_IP>”}
}
```

Multi-Site

One of the new features of OASWeb is the capability to run multiple OAS clients’ OASWeb interfaces within the same browser page separated by individual tabs. Each client included in this file also needs to be configured in the nginx configuration file. The formatting of how to configure that file is outlined in the upcoming section “*Updating Server Scripts: NGINX Configuration*”. To configure a multi-site sites.json file, the following formatting is required.

```
{
  “<client_1>”:{“type”:”Site”,”name”:”<Tab Name_1>”,”mqtt”:”wss:// <Server_IP>:9001”,”url”:” <Server_IP>”}
  “<client_2>”:{“type”:”Site”,”name”:”<Tab Name_2 >”}
  “<client_3>”:{“type”:”Site”,”name”:”<Tab Name_3 >”}
}
```

Once the configurations have been updated, please reboot the server.

```
guardvant@<ServerName>:~$ sudo reboot
```

YOUR SERVER IS READY.

About Hexagon

Hexagon is a global leader in digital reality solutions, combining sensor, software and autonomous technologies. We are putting data to work to boost efficiency, productivity, quality and safety across industrial, manufacturing, infrastructure, public sector, and mobility applications.

Our technologies are shaping production and people- related ecosystems to become increasingly connected and autonomous – ensuring a scalable, sustainable future.

Hexagon's Mining division solves surface and underground mine challenges with proven technologies for planning, operations, and safety.

Hexagon (Nasdaq Stockholm: HEXA B) has approximately 22,000 employees in 50 countries and net sales of approximately 5.1bn USD. Learn more at hexagon.com and follow us @HexagonAB.

Visit us at hexagonmining.com