# Postgresql 9.5 (Jay)

1. sudo apt-get update
2. sudo apt-get install postgresql postgresql-contrib
3. sudo -u postgres createuser --interactive
   * Name: fhir
   * Superuser? Y
4. sudo -u postgres createdb fhir
5. sudo -u postgres psql
   * \password fhir
   * \q
6. sudo -u postgres nano /etc/postgresql/9.5/main/pg\_hba.conf
   * add line “host all all 0.0.0.0/0 md5”
7. sudo –u postgres nano /etc/postgresql/9.5/main/postgresql.conf
   * change “#listen\_addresses = ‘localhost’” to “listen\_addresses = ‘\*’”
8. sudo /etc/init.d/postgresql restart

Steps 6 through 8 allow direct access to the postgres database. By default, postgres installation limits database connections to localhost only.

# PostGIS 2.X install (Peter)

1. sudo apt install postgis
2. sudo su – postgres
3. psql fhir -c “create extension postgis”
4. psql fhir -c “create extension fuzzystrmatch”
5. psql fhir -c “create extension postgis\_tiger\_geocoder”
6. psql fhir -c “create user synth\_ma encrypted password 'synth\_ma123'”
7. --export/import schemas: tiger\_data (ma partitions), tiger\_cb14\_500k, synth\_ma, zip (zip\_to\_zcta)

# Tomcat 8 (Jay)

1. sudo apt-get update
2. sudo apt-get install default-jdk
3. sudo groupadd tomcat
4. sudo useradd -s /bin/false -g tomcat -d /opt/tomcat tomcat
5. curl --proxy http://gatekeeper.mitre.org:80 -O http://mirror.reverse.net/pub/apache/tomcat/tomcat-8/v8.5.3/bin/apache-tomcat-8.5.3.tar.gz
6. sudo mkdir /opt/tomcat
7. sudo tar xzvf apache-tomcat-8.5.3.tar.gz -C /opt/tomcat/ --strip-components=1
8. cd /opt/tomcat/
9. sudo mkdir target
10. sudo mkdir target/lucenefiles
11. sudo chgrp -R tomcat /opt/tomcat
12. sudo chown -R tomcat /opt/tomcat
13. sudo chmod g+rwx conf
14. sudo chmod -R g+r conf
15. sudo nano /etc/systemd/system/tomcat.service

|  |
| --- |
| Contents of tomcat.service file |
| [Unit]  Description=Apache Tomcat Web Application Container  After=network.target  [Service]  Type=forking  Environment=JAVA\_HOME=/usr/lib/jvm/java-1.8.0-openjdk-amd64  Environment=CATALINA\_PID=/opt/tomcat/temp/tomcat.pid  Environment=CATALINA\_HOME=/opt/tomcat  Environment=CATALINA\_BASE=/opt/tomcat  Environment='CATALINA\_OPTS=-Xms512M -Xmx2048M -server -XX:+UseParallelGC'  Environment='JAVA\_OPTS=-Djava.awt.headless=true -Djava.security.egd=file:/dev/./urandom'  ExecStart=/opt/tomcat/bin/startup.sh  ExecStop=/opt/tomcat/bin/shutdown.sh  User=tomcat  Group=tomcat  RestartSec=10  Restart=always  [Install]  WantedBy=multi-user.target |

1. Development/Integration Only
   * sudo nano /opt/tomcat/conf/tomcat-users.xml
   * Add “<user username="fhir" password="fhir" roles="manager-gui" />” within <tomcat-users/>
2. sudo systemctl daemon-reload
3. sudo systemctl enable tomcat
4. sudo systemctl start tomcat
5. systemctl status tomcat.service

# Deploy SyntheticMass fork of HAPI-FHIR (Jay)

1. mvn package
2. mv hapi-fhir-jpaserver-example/target/hapi-fhir-jpaserver-example.war ./fhir.war
3. Copy fhir.war to Ubuntu machine /opt/tomcat/webapps/fhir.war
4. Webapp will autodeploy (if Tomcat is running)
   * Webpage: http://[server]:8080/fhir
   * FHIR REST API: <http://[server]:8080/fhir/baseDstu3>
5. If there are errors in running the war file:
   * sudo less /opt/tomcat/logs/catalina.out

# Move Tomcat 8 to Port 80 (Greg)

NOTE: End up moving tomcat to port 8081 and proxying it via apache web server on port 80

1. Echo $UID to find out my user id
2. sudo apt-get install unzip
3. sudo apt-get install ant
4. vim /opt/tomcat/conf/server.xml
5. Change port from 8080 to 80 and save and exit
6. Restarted tomcat and now it works!
7. See port usage: sudo netstat –peanut
8. see processes for process ids: ps all

# Install htc-api (Greg)

1. pip freeze got error that pip was not installed
2. sudo apt install python-pip
3. pip install virtualenv
4. added the following lines to the end of ~/.bashrc:

$ export no\_proxy='localhost,127.0.0.1,.mitre.org'

$ export NO\_PROXY=$no\_proxy

# Proxy on/off

$ alias proxy-on="export http\_proxy='http://gatekeeper.mitre.org:80'; export https\_proxy='http://gatekeeper.mitre.org:80'; export HTTP\_PROXY=$http\_proxy; export HTTPS\_PROXY=$https\_proxy;"

$ alias proxy-off="unset http\_proxy ; unset https\_proxy ; unset HTTP\_PROXY ; unset HTTPS\_PROXY"

1. source ~/.bashrc
2. pip install virtualenv 🡪 FAILED
   1. apt-get remove python-pip python3-pip
   2. wget <https://bootstrap.pypa.io/get-pip.py> --no-check-certificate
   3. sudo –H python get-pip.py 🡪 FAILED
   4. sudo apt-get install python-pip
   5. sudo –H pip install virtualenv --proxy=gatekeeper.mitre.org:80 (indicated newer pip version available)
   6. ~~pip install --upgrade pip (skipped this the 2~~~~nd~~ ~~time because virtualenv venv was failing)~~
3. ssh-keygen –t rsa
4. more ~/.ssh/id\_rsa.pub
5. copy key
6. went to gitlab.mitre.org and added the public key using copy above
7. git clone [git@gitlab.mitre.org:HTCProject/htc-api.git](mailto:git@gitlab.mitre.org:HTCProject/htc-api.git)
8. cd htc-api/api
9. virtualenv venv 🡪 FAILED (setuptools pkg\_resources pip wheel failed with error code 2)
10. <https://github.com/pypa/pip/issues/1805>
11. Uninstalled pip and virtualenv and re-installed. Did #6 then continued at #13
12. Now get OSError: [Errno 17] File exists
13. Deleted venv directory in htc-api/api
14. virtualenv venv
15. source venv/bin/activate
16. pip install –r requirements.txt --proxy gatekeeper.mitre.org:80
    1. Command “python setup.py egg\_info” failed with error code 1 in /tmp/pip-build-Akzwsl/psycopg2/
    2. Error: You need to install postgresql-server-dev-X.Y for building a server-side extension or libpq-dev for building a client-side application
    3. sudo apt-get install postgresql 🡪 says already installed
    4. sudo apt-get install python-psycopg2
    5. sudo apt-get install libpq-dev
    6. Re-ran pip install –r requirements.txt –proxy gatekeeper.mitre.org:80
17. . htc\_run.sh
18. Hit <http://syntheticmass.mitre.org:8080/htc/api/v1> from firefox and it works fine
19. . test\_rest.sh

<returned here>

1. Moved htc-api to be under syntheticmass in /home/gquinn/

# Update htc-api (Greg)

1. Made pct\_male and pct\_female returned for counties/stats return percentages as decimalvs between 0 and 1.
2. From ~: git clone https://gregquinn2001@github.com/synthetichealth/syntheticmass.git
3. From ~: mv htc\_api syntheticmass
4. git add –a
5. git commit –m “1st push of htc-api”
6. git push

# Use local postgres (Greg)

1. Changed htc\_login.txt to point at local postgres
2. Changed mht schema in queries within htc\_api.py to synth\_ma
3. . htc\_kill.sh
4. . htc\_run.sh (make sure virtual environment activated with source command)
5. . test\_rest.sh
6. Error: relation “county\_health.chr” does not exist
   1. Sent e-mail to Peter Sylvester (cc’ed Andre and Jay) about missing table chr.
   2. Peter copied the chr table over from mht to syntheticmass postgres
7. . htc\_kill.sh
8. . htc\_run.sh (make sure virtual environment activated with source command)
9. . test\_rest.sh (all tests pass)

# Mod\_proxy (Greg)

(both fhir and api served through apache on port 80)

1. sudo apt-get install apache2 apache2-doc apache2-utils
2. sudo vim /etc/apache2/apache2.confs
3. sudo service apache2 restart
4. sudo a2dissite 000-default.conf
5. sudo vim /etc/apache2/sites-available/syntheticmass.mitre.org.conf
6. sudo mkdir –p /var/www/syntheticmass.mitre.org/public\_html
7. sudo mkdir /var/www/syntheticmass.mitre.org/logs
8. sudo a2ensite syntheticmass.mitre.org.conf
9. sudo service apache2 restart
10. vim /opt/tomcat/conf/server.xml
11. Change port from 80 to 8081 and save and exit
12. Restarted tomcat
13. sudo a2enmod proxy\_http
14. service apache2 restart
15. Edited the configuration /etc/apache2/sites-available/syntheticmass.mitre.org.conf (red lines above may be replaced by this instead):

<VirtualHost \*:80>

#ServerAdmin webmaster@syntheticmass.mitre.org

ServerName syntheticmass.mitre.org

#ServerAlias www.syntheticmass.mitre.org

#DocumentRoot /var/www/syntheticmass.mitre.org/public\_html

ErrorLog /var/www/syntheticmass.mitre.org/error.log

CustomLog /var/www/syntheticmass.mitre.org/access.log combined

ProxyPass "/api" "http://syntheticmass.mitre.org:8080/htc/api"

ProxyPassReverse "/api" "http://syntheticmass.mitre.org:8080/htc/api"

ProxyPass "/fhir" "http://syntheticmass.mitre.org:8081/fhir"

ProxyPassReverse "/fhir" "http://syntheticmass.mitre.org:8081/fhir"

</VirtualHost>

1. sudo service apache2 restart

NOTE: should switch to using AJP connector on tomcat behind proxy

# Install DIRECT reference implementation (bare metal)

<http://api.nhindirect.org/java/site/assembly/stock/3.0.1/users-guide/depl-hisp-only.html>

1. sudo apt-get install unzip
2. sudo apt-get install ant
3. export JAVA\_HOME=/usr/lib/jvm/java-8-openjdk-amd64/
4. echo "export JAVA\_HOME=$JAVA\_HOME" | sudo tee -a /etc/environment
5. Wrote, compiled, and ran simple program to check max AES encryption key length:

<https://gist.github.com/jehrhardt/5167854>

Key length is unlimited so no JCE install required for OpenJDK 8

1. cd /opt
2. sudo wget -e use\_proxy=yes -e https\_proxy=http://gatekeeper.mitre.org:80/ --no-check-certificate <https://oss.sonatype.org/content/repositories/releases/org/nhind/direct-project-stock/3.0.1/direct-project-stock-3.0.1.tar.gz>
3. sudo tar xvfz direct-project-stock-3.0.1.tar.gz
4. export DIRECT\_HOME=`pwd`/direct
5. echo "export DIRECT\_HOME=$DIRECT\_HOME" | sudo tee -a /etc/environment
6. cp /opt/direct/apache-tomcat-7.0.41/webapps/\*.war /opt/tomcat/webapps
7. cp –R /opt/direct/apache-tomcat-7.0.41/bin/nhindconfig /opt/tomcat/bin
8. /opt/tomcat/bin/shutdown.sh
9. /opt/tomcat/bin/startup.sh
10. tail –f /opt/tomcat/logs/catalina.out
11. <http://syntheticmass.mitre.org:8081/config-ui>
12. certGen.sh requires windowing

# ~~Install Unity 8~~

<http://news.softpedia.com/news/how-to-install-unity-8-on-ubuntu-16-04-lts-and-ubuntu-15-10-496949.shtml>

1. sudo apt-get update && sudo apt-get dist-upgrade
2. sudo apt-get install unity8-desktop-session-mir
3. sudo startx 🡪 failing
4. sudo apt-get --purge remove \*unity\*

# Support synthetic statistics

1. Open pgadmin II and connect to syntheticmass.mitre.org on port 5432
2. Open fhir database
3. Select table county\_stats and copy SQL pane contents
4. Open query window by clicking on SQL Query icon in toolbar
5. Paste “Create TABLE county\_stats” that you just copied from SQL Pane
6. Update table name in 4 places to have “synth\_” in front of table name so you end up with synth\_county\_stats (CREATE TABLE, CONSTRAINT, ALTER TABLE, GRANT ALL ON TABLE)
7. Execute query
8. Repeat steps 3-7 for cousub\_stats to create synth\_cousub\_stats
9. Updated ~/syntheticmass/htc-api/api/htc\_api.py.
   1. For each rest api method which queried statistics data, created a new one with a synth prefix that goes against the synth\_ prefixed table
10. Did a backup of the data from the non synth tables and then opened that file in the pgadmin query tool.
11. Searched and replace the table name with the synth\_ prefixed one
12. Execute queries
13. Changed the data in county mode
14. *< Need a script to reset all data in synth\_ to zeros >*

# Deploy synthetic mass web site

1. sudo su
2. export https\_proxy='http://gatekeeper.mitre.org:80'
3. export HTTPS\_PROXY=$https\_proxy
4. cd ~/syntheticmass
5. git pull
6. sudo apt-get install nodejs
7. sudo apt-get install npm
8. ln -s /usr/bin/nodejs /usr/bin/node (https://github.com/sass/node-sass/issues/1601)
9. npm install (as root)
10. sudo npm run build
11. cp -R build/\* /var/www/syntheticmass.mitre.org/public\_html
12. sudo vim /etc/apache2/sites-available/syntheticmass.mitre.org.conf
13. uncommented out “DocumentRoot /var/www/syntheticmass.mitre.org/public.html”
14. Added after DocumentRoot within VirtualHost:

<Directory “/”>

DirectoryIndex index.html

</Directory>

1. Added “Header set Access-Control-Allow-Origin “\*”
2. saved and closed
3. sudo service apache2 reload

# SSL Enable Apache Web Server

<https://www.digitalocean.com/community/tutorials/how-to-create-a-ssl-certificate-on-apache-for-ubuntu-14-04>

1. sudo a2enmod ssl
2. sudo service apache2 restart
3. sudo mkdir /etc/apache2/ssl
4. sudo openssl
5. req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/apache2/ssl/apache.key -out /etc/apache2/ssl/apache.crt
   1. US
   2. Massachusetts
   3. Bedford
   4. MITRE
   5. Synthetic Mass
   6. syntheticmass.mitre.org
   7. < no e-mail address >
6. sudo nano /etc/apache2/sites-available/default-ssl.conf
7. Changed ServerAdmin
8. Added ServerName and ServerAlias
9. Updated DocumentRoot to be /var/www/syntheticmass.mitre.org/public\_html
10. Added Directory element with DirectoryIndex of index.html nested in it
11. Added ProxyPass and ProxyPassReverse for /api and /fhir:

ProxyPass "/api" <http://syntheticmass.mitre.org:8080/htc/api>

ProxyPassReverse "/api" "http://syntheticmass.mitre.org:8080/htc/api"

ProxyPass "/fhir" "http://syntheticmass.mitre.org:8081/fhir"

ProxyPassReverse "/fhir" "http://syntheticmass.mitre.org:8081/fhir"

1. Write out and close nano
2. sudo a2ensite ssl-syntheticmass.mitre.org.conf
3. sudo service apache2 restart
4. sudo a2dissite syntheticmass.mitre.org.conf
5. sudo service apache2 reload
6. sudo nano /etc/apache2/apache2.conf
   1. Remove “indexes” from options for Directory /var/www
   2. Save and close
7. sudo service apache2 restart

# Block traffic except to port 443 and allow loopback traffic

1. sudo iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
2. sudo iptables -A INPUT -p tcp --dport ssh -j ACCEPT
3. sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
4. sudo iptables -A INPUT -j DROP
5. sudo iptables -I INPUT 1 -i lo -j ACCEPT
6. sudo iptables -I INPUT 5 -m limit --limit 5/min -j LOG --log-prefix "iptables denied: " --log-level 7
7. sudo iptables -I INPUT 1 -p tcp --dport 443 -j ACCEPT
8. sudo nano webpack.config.js
   1. Changed API base URL to use https instead of http (API\_HOST for production case)
   2. Save and close
9. npm run build
10. rm -r /var/www/syntheticmass.mitre.org/public\_html/\*
11. cp -R build/\* /var/www/syntheticmass.mitre.org/public\_html/
12. sudo service apache2 reload
13. sudo sh -c "iptables-save > /etc/iptables.rules"
14. sudo nano /etc/network/interfaces
    1. Edit and add to end:

pre-up iptables-restore < /etc/iptables.rules

* 1. Save and close

# Install Apache James

<http://api.nhindirect.org/java/site/assembly/stock/3.0.1/users-guide/depl-hisp-only.html>

<http://stackoverflow.com/questions/12626027/chained-client-certificates>

<http://api.nhindirect.org/java/site/agent/1.5.1/users-guide/dev-cert-gen.html>

1. mkdir direct\_certs
2. cd direct\_certs
3. sudo openssl genrsa -out rootCA.key 2048
4. sudo openssl req -x509 -new -nodes -key rootCA.key -sha256 -days 1024 -out rootCA.pem
   1. US
   2. Massachusetts
   3. Bedford
   4. MITRE
   5. Synthetic Mass
   6. Syntheticmass.mitre.org Root CA
   7. [admin@syntheticmass.mitre.org](mailto:admin@syntheticmass.mitre.org)
5. sudo openssl genrsa -out device.key 2048
6. sudo openssl req -new -key device.key -out device.csr
   1. Country: US
   2. State: Massachusetts
   3. Locality: Bedford
   4. Org: MITRE
   5. Org unit: Synthetic Mass
   6. Common name: direct.syntheticmass.mitre.org
   7. Email: [admin@direct.syntheticmass.mitre.org](mailto:admin@direct.syntheticmass.mitre.org)
   8. Challenge Password: Password\_1
   9. Company name: MITRE
7. sudo openssl x509 -req -in device.csr -CA rootCA.pem -CAkey rootCA.key -CAcreateserial -out device.crt -days 500 -sha256
8. sudo openssl x509 -in rootCA.pem -out rootCA.der -outform DER
9. sudo openssl rsa -in rootCA.key -out rootCAkey.der -outform DER
10. sudo openssl x509 -in device.crt -out org.der -outform DER
11. sudo openssl rsa -in device.key -out orgkey.der -outform DER
12. sudo openssl pkcs12 -export -out org.p12 -inkey device.key -in device.crt -certfile rootCA.pem
    1. Export password: Password\_1
13. Temporarily Open config port 8081 (http)
    1. sudo iptables -I INPUT 1 -p tcp --dport 8081 -j ACCEPT
14. Use SFTP server (Bitvise SSH client) to copy ~/direct\_certs to desktop on windows machine so they can be accessed from browser
15. <http://syntheticmass.mitre.org:8081/config-ui>
    1. Create New Domain
       1. Domain Name: direct.syntheticmass.mitre.org
       2. Postmaster e-mail: [postmaster@direct.syntheticmass.mitre.org](mailto:postmaster@direct.syntheticmass.mitre.org)
       3. Status: ENABLED
       4. Click Add
    2. Click “Anchors” tab
       1. Browse to the certificate in <Desktop>\direct\_certs
       2. Choose rootCA.der
       3. Check off Incoming and Outgoing
       4. Choose Status ENABLED
       5. Click Add Anchor
    3. Click “Certificates” in top toolbar
       1. Click Browse and go to <Desktop>\direct\_certs
       2. Choose org.p12
       3. Status: ENABLED
       4. Click Add Certificate.
16. cd /opt/direct/james-2.3.2
17. sudo sh bin/setdomain.sh direct.syntheticmass.mitre.org
18. Delete temporary access to port 8081
    1. sudo iptables -L -v --line-numbers
    2. sudo iptables -D INPUT 1
19. sudo -E sh bin/run.sh &
20. telnet localhost 4555
    1. root
    2. root
    3. adduser gquinn gquinn
    4. quit
21. sudo iptables -L -v --line-numbers
22. sudo iptables -I INPUT 4 -p tcp --dport 25 -j ACCEPT
23. sudo sh -c "iptables-save > /etc/iptables.rules"

Only allows SMTP traffic – no POP or IMAP.

# Build and Deploy HAPI-FHIR JPA Server

1. cd ~
2. git clone [git@gitlab.mitre.org:synthea/hapi-fhir.git](mailto:git@gitlab.mitre.org:synthea/hapi-fhir.git)
3. sudo apt install maven
4. If not already retrieved, get sonatype certificate using your laptop:
   1. Using firefox browse to <https://oss.sonatype.org/>
   2. Click on lock icon and click on right arrow then more information
   3. Click View Certificate
   4. Click on Details tab
   5. Click on Export… button
   6. Save certificate as \*.crt to downloads in user directory (c:\users\gquinn\Downloads\-.sonatype.org.crt)
5. Use SFTP (Bitvise) to copy -.sonatype.org.crt to home directory (~) on syntheticmass.mitre.org
6. sudo keytool -importcert -file -.sonatype.org.crt -keystore /usr/lib/jvm/java-8-openjdk-amd64/jre/lib/security/cacerts
   1. changeit
   2. y
7. sudo nano /etc/maven/settings.xml
   1. Uncomment out proxy and create one to gatekeeper.mitre.org on port 80 for http
   2. Save and close (Ctrl-O, Ctrl-X)
8. cd ~/hapi-fhir
9. mvn install (failed on hapi-fhir-jpaserver-example)
10. cd hapi-fhir-jpaserver-example
11. mvn package
12. sudo mv /opt/tomcat/webapps/fhir.war /opt/tomcat/webapps/fhir.bkp
13. sudo mv target/hapi-fhir-jpaserver-example.war /opt/tomcat/webapps/fhir.war
14. cd /opt/tomcat
15. sudo bin/shutdown.sh (wait a minute)
16. sudo bin/startup.sh (wait a few minutes)

# Redirect port 80 to port 443

1. sudo nano /etc/apache2/sites-available/syntheticmass.mitre.org.conf
2. Edit to look like:

<VirtualHost \*:80>

ServerName syntheticmass.mitre.org

ServerAlias www.syntheticmass.mitre.org

Redirect permanent / https://syntheticmass.mitre.org/

</VirtualHost>

* 1. Save and close

1. sudo a2ensite syntheticmass.mitre.org.conf
2. sudo service apache2 reload
3. rm /opt/tomcat/webapps/config-service.war (no longer needed and was endless looping on startup so just removed it for now)

# Fix conformance retrieval

1. cd ~/hapi-fhir/hapi-fhir-jpaserver-example
2. nano src/main/java/ca/uhn/fhir/jpa/demo/FhirTesterConfig.java
   1. update .withBaseUrl and replace ${serverBase} with <https://syntheticmass.mitre.org/fhir>”
   2. Write and close
3. Go to “Rebuilding HAPI-FHIR server on Ubuntu Server” below

# Allow tomcat to call back to itself (going through apache web server)

1. sudo keytool -alias apachewebserver -importcert -file /etc/apache2/ssl/apache.crt -keystore /usr/lib/jvm/java-8-openjdk-amd64/jre/lib/security/cacerts
   1. changeit
   2. y
2. sudo /opt/tomcat/bin/shutdown.sh
3. sudo /opt/tomcat/bin/startup.sh

# Rebuilding HAPI-FHIR server on Ubuntu Server

1. If changes are in a dependent project used by hapi-fhir-jpaserver-example (e.g., hapi-fhir-base or hapi-fhir-jpaserver-base)
   1. cd ~/hapi-fhir/< project name e.g. hapi-fhir-base >
   2. make changes to source files with src directory of that project
   3. mvn install
2. cd ~/hapi-fhir/hapi-fhir-jpaserver-example
3. Make changes to source files within hapi-fhir-jpaserver-example
4. mvn package
5. sudo cp target/hapi-fhir-jpaserver-example.war /opt/tomcat/webapps/fhir.war
6. sudo tail -f /opt/tomcat/logs/catalina.out
7. wait for deployment to be finished (about 46 seconds) by watching for log message

# Zero Statistics

1. psql -U fhir -W -h localhost -d fhir -c "update synth\_ma.synth\_county\_stats set pop\_male = 0, pop\_female = 0, pop = 0"
   1. fhir
2. psql -U fhir -W -h localhost -d fhir -c "update synth\_ma.synth\_cousub\_stats set pop\_male = 0, pop\_female = 0, pop = 0"
   1. fhir

# Deploy Web Site Updates

1. sudo su
2. cd ~/syntheticmass
3. export https\_proxy='http://gatekeeper.mitre.org:80'
4. export HTTPS\_PROXY=$htts\_proxy
5. git pull
6. exit
7. cd ~syntheticmass/site
8. sudo npm run build
9. cp -R build/\* /var/www/syntheticmass.mitre.org/public\_html
10. sudo service apache2 reload