This document steps through commands for verifying OpenELIS configuration for patient search from iSanté and results reporting to iSanté.ssh. These are followed by command to determine whether or not these interfaces are actually being used on the server. The commands are either Linux commands or SQL commands within the Postgres client.

[Look at OpenElis backup files on strontium](#_s18nf6ugl2wx)

[First, gain access to the server, either through the VPN, through the server’s public IP, or via the server’s console.](#_6kyn7ioox9og)

[If via arsenic, log into arsenic and fetch IP information:](#_wyeyjzesmyl9)

[If this fails, go into mysql and fetch the public IP from the database:](#_fs6l67226p1n)

[Next, log into the server, using either the VPN IP or the Public IP, respectively (or log in to the console):](#_php76z2n8qxt)

[Switch to a root shell:](#_vqt4kdcar7av)

[Check to make sure the PatientSearchService still exists as a webapp:](#_qe4i0nr2119z)

[Check the server IP:](#_a4mih8r4ebfn)

[Fetch the Postgres password from the tomcat conf file:](#_2ug4f2a5po7)

[Invoke the Postgres client:](#_xhqdwjrfitlm)

[Verify configuration: are patient search and results reporting configured?](#_y1rbo3230poz)

[If IP addresses determined above aren’t 192.168.1.50, you must update the configuration to match the correct URL or use localhost:](#_a648iuq7ku0d)

[Are the interfaces being used?](#_ekg3jt6ztae4)

[Are there any iSanté patients fetched into OE?](#_sgx73g43zk3y)

[Are any results queued up for sending?](#_4gvjoaic7dv9)

[Do any isante patients have results?](#_atxsl7n8x7aj)

[Exit from the Postgres client:](#_k36fbe52uf9g)

### Look at OpenElis backup files on strontium

I couldn't remember which one of you said they would run the query against the backup data bases to find an installation of OE which has been getting patients from iSante but below is what will give you the count. An easier way may be to just search for the part of the backup (they are in plain text) for the patient\_identity table data and see how many of the fields look like GUIDs. The backup files are on strontium /srv/www/[openelis-recv.cirg.washington.edu/receive-file](http://openelis-recv.cirg.washington.edu/receive-file)

select count(\*) from clinlims.patient\_identity pi

where pi.identity\_type\_id = (select id from clinlims.patient\_identity\_type pit where pit.identity\_type = 'GUID' );

### First, gain access to the server, either through the VPN, through the server’s public IP, or via the server’s console.

#### If via arsenic, log into arsenic and fetch IP information:

shw2@arsenic:~$ sudo ipsec status | grep 81100

isante-d175s81100[13497]: ESTABLISHED 70 minutes ago, 140.142.7.189[C=HT, CN=isante-consolidated.cirg.washington.edu]...67.47.153.206[C=HT, CN=isante-d175s81100]

isante-d175s81100{3605}: INSTALLED, TUNNEL, ESP in UDP SPIs: c0aaa097\_i ccee64c1\_o

isante-d175s81100{3605}: 140.142.7.189/32 === 172.20.1.175/32

**Note, in the above, that the public IP follows the string ‘edu]...’, the dbsite/sitecode is in the pattern ‘dxxxsyyyy’, and the VPN IP is in the pattern ‘172.20.1.xxx’.**

#### If this fails, go into mysql and fetch the public IP from the database:

mysql -u cirgadmin -p itech

mysql> select ipAddress from clinicLookup where sitecode = 81100;

+---------------+

| ipAddress |

+---------------+

| 67.47.153.206 |

+---------------+

#### Next, log into the server, using either the VPN IP or the Public IP, respectively (or log in to the console):

shw2@arsenic:~$ sudo -u cirgadmin ssh 172.20.1.175

shw2@arsenic:~$ sudo -u cirgadmin ssh 67.47.153.206

#### Switch to a root shell:

cirgadmin@isante-81100:~$ sudo sh

#### Check to make sure the PatientSearchService still exists as a webapp:

sh-3.2# ls /usr/share/tomcat5.5/webapps

haitiOpenElis haitiOpenElis.war PatientSearchService

#### Check the server IP:

sh-3.2# /sbin/ifconfig

eth0 Link encap:Ethernet HWaddr 64:31:50:4f:78:c8

inet adr:**192.168.1.50** Bcast:192.168.1.255 Masque:255.255.255.0

adr inet6: fe80::6631:50ff:fe4f:78c8/64 Scope:Lien

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:75628 errors:1 dropped:0 overruns:0 frame:1

TX packets:87187 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 lg file transmission:1000

RX bytes:15436986 (14.7 MiB) TX bytes:72367900 (69.0 MiB)

Interruption:16 Mémoire:f4000000-f4012100

lo Link encap:Boucle locale

inet adr:127.0.0.1 Masque:255.0.0.0

adr inet6: ::1/128 Scope:Hôte

UP LOOPBACK RUNNING MTU:16436 Metric:1

RX packets:54489 errors:0 dropped:0 overruns:0 frame:0

TX packets:54489 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 lg file transmission:0

RX bytes:21510627 (20.5 MiB) TX bytes:21510627 (20.5 MiB)

#### Fetch the Postgres password from the tomcat conf file:

sh-3.2# more /usr/share/tomcat5.5/conf/Catalina/localhost/haitiOpenElis.xml

<Context path="haitiOpenElis" >

<Resource name="jdbc/LimsDS" auth="Container"

type="javax.sql.DataSource" driverClassName="org.postgresql.Driver"

url="jdbc:postgresql://localhost:5432/clinlims"

username="clinlims" password="PASSWORD LOCATED HERE" maxActive="20" maxIdle="10"

maxWait="-1"/>

</Context>

#### Invoke the Postgres client:

sh-3.2# psql -h localhost -U clinlims clinlims

### Verify configuration: are patient search and results reporting configured?

clinlims=> select name, value from clinlims.site\_information

where name in ('resultReporting', 'resultReportingURL', 'patientSearchURL', 'patientSearchLogOnUser', 'patientSearchPassword', 'useExternalPatientSource');

name | value

--------------------------+---------------------------------------

patientSearchLogOnUser | iSanteSvcUser

patientSearchPassword | cq:qzhuxpVz80oD

useExternalPatientSource | true

resultReporting | true

patientSearchURL | https://192.168.1.50/PatientSearchService/iSante/services/patients

resultReportingURL | https://192.168.1.50/isante/receiveLabResults.php

(6 lignes)

#### If IP addresses determined above aren’t 192.168.1.50, you must update the configuration to match the correct URL or use localhost:

clinlims=> update clinlims.site\_information

set value = ‘https://WW.XX.YY.ZZ/PatientSearchService/iSante/services/patients’

where name = ‘patientSearchURL’;

clinlims=> update clinlims.site\_information

set value = 'https://WW.XX.YY.ZZ/iSante/receiveLabResults.php'

where name = 'resultReportingURL';

**Or:**

clinlims=> update clinlims.site\_information

set value = ‘https://localhost/PatientSearchService/iSante/services/patients’

where name = ‘patientSearchURL’;

clinlims=> update clinlims.site\_information

set value = 'https://localhost/iSante/receiveLabResults.php'

where name = 'resultReportingURL';

### Are the interfaces being used?

#### Are there any iSanté patients fetched into OE?

clinlims=> select count(\*) from clinlims.patient\_identity

where identity\_type\_id = ( select id from clinlims.patient\_identity\_type where identity\_type = 'GUID' );

count

-------

475

(1 ligne)

#### Are any results queued up for sending?

clinlims=> select count(\*) from clinlims.report\_external\_export

where type = (select id from clinlims.report\_queue\_type where name = 'Results' );

count

-------

0

(1 ligne)

#### Do any isante patients have results?

clinlims=> select pi.identity\_data as GUID, t.description as test, r.lastupdated from clinlims.patient p

join clinlims.patient\_identity pi on pi.patient\_id = p.id

join clinlims.sample\_human sh on sh.patient\_id = p.id

join clinlims.sample s on sh.samp\_id = s.id

join clinlims.sample\_item si on si.samp\_id = s.id

join clinlims.analysis a on a.sampitem\_id = si.id

join clinlims.result r on r.analysis\_id = a.id

join clinlims.test t on t.id = a.test\_id

where pi.identity\_type\_id =

(select id from clinlims.patient\_identity\_type where identity\_type = 'GUID' );

#### Exit from the Postgres client:

clinlims=> \q