

Load Testing Framework

IdP and LDAP

Notes taken from Shibboleth v3 work. Grinders can be recycled, but Salt commands will have to be changed by someone who understands our Salt setup better.

ssh to grinder-master

grinder-master static IP: 54.244.88.132

Updating nodes(run on 54.244.88.132):

```
mv file.crt /var/www/dist/  
salt "*" cmd.run "curl -o http://54.244.88.132/dist/file.crt /path/to/file.crt"
```

e.g.

```
cp tests/saml2sso_idp/test.properties /var/www/dist/  
salt -G "roles:idp" cmd.run "curl -o /opt/shibboleth-idp/conf/logback.xml http://54.244.88.132/dist/logback.xml"
```

Interesting Paths:

```
/opt/jetty/  
/opt/shibboleth-idp/  
/opt/grinder-framework/
```

Starting jetty:

```
salt -G "roles:idp" cmd.run "cd /opt/shibboleth-idp/jetty-base/ && nohup java -jar /opt/jetty/start.jar >> /dev/null &"
```

Stopping jetty:

```
salt -G "roles:idp" cmd.run "kill \$(ps aux | grep '[j]etty' | awk '{print \$2}')
```

Restarting jetty:

```
salt -G "roles:idp" cmd.run "kill \$(ps aux | grep '[j]etty' | awk '{print \$2}') && sleep 10 && cd /opt/shibboleth-idp/jetty-base/ && nohup java -jar /opt/jetty/start.jar >> /dev/null &"
```

Checking jetty:

```
salt -G "roles:idp" cmd.run "ps -ef | grep jetty"
```

Starting The Grinder Console:

Either figure out **X-Windows** and

```
ssh -X 54.244.88.132  
cd /opt/grinder-framework/  
./bin/console.sh
```

or

```
ssh 54.244.88.132
vncserver :1
```

Connect to grinder:5901 using VNC client
Open terminal in Ubuntu

```
cd /opt/grinder-framework/  
./bin/console.sh
```

Changing test parameters:

Host new config file at 54.244.88.132/var/www/dist/

```
salt -G "roles:grinder" cmd.run "curl -o /opt/grinder-framework/tests/saml2sso_idp/test.properties http://54.244.88.132/dist/test.properties && curl -o /opt/grinder-framework/tests/saml2sso_idp/test.py http://54.244.88.132/dist/test.py"
```

Initializing agents:

```
salt -G "roles:grinder" cmd.run "cd /opt/grinder-framework && ./bin/agent.sh saml2sso_idp &"
```

Checking on agents:

```
salt -G "roles:grinder" cmd.run "ps -ef | grep agent"
```

Killing agents:

```
salt -G "roles:grinder" cmd.run "kill \$(ps aux | grep '[a]gent.sh' | awk '{print \$2}')
```

The modified load testing script can either randomly choose one of 500000 users or do repeated SSO as a single user to emulate slightly more realistic load; just change test.py.