VirtualMachine

Notes on setting up the Tika Virtual Machine

See TIKA-1331

Install software (this has been updated for Ubuntu)

1. sudo apt update
2. gpg
    a. sudo apt install gnupg
3. java
    a. `wget -qO - https://adoptopenjdk.jfrog.io/adoptopenjdk/api/gpg/key/public | sudo apt-key add -`
    b. `sudo apt-get install -y software-properties-common`
    c. `sudo add-apt-repository --yes https://adoptopenjdk.jfrog.io/adoptopenjdk/deb/`
    d. `sudo apt-get install adoptopenjdk-8-hotspot`
    e. `sudo apt-get install adoptopenjdk-11-hotspot`
    f. `sudo apt-get install adoptopenjdk-14-hotspot`
4. `sudo apt-get install fontconfig (https://github.com/AdoptOpenJDK/openjdk-build/issues/693 via Dominik Stadler)`
5. `sudo apt install ttf-dejavu (same as above)`
6. sudo apt-get install groovy
7. sudo apt-get install maven
8. sudo apt-get install subversion
9. sudo apt-get install git
10. sudo apt-get install file
11. installed docker following: https://docs.docker.com/engine/install/ubuntu/

Datasette

On 12 November 2020, I ran tika-eval's new FileProfile on the corpus. This includes file type detection by Tika and by 'file', digests and file sizes.

We configured the reverse proxy for /datasette:

ProxyPreserveHost On
ProxyPass /datasette http://0.0.0.0:8001
ProxyPassReverse /datasette http://0.0.0.0:8001

The .db is in /data1/publish. cd to that directory and then:

```
docker run --name datasette -d -p 8001:8001 -v `pwd`:/mnt datasetteproject/datasette datasette -p 8001 -h 0.0.0.0 /mnt/file_profiles.db --config sql_time_limit_ms:120000 --config max_returned_rows:100000 --config base_url:/datasette/
```

HTTPD

/etc/apache2

public directories are symlinks in /usr/share/corpora

Everything below here needs to be updated for Ubuntu

config/admin stuff

```
nano /etc/ssh/sshd_config
```

- disallow root to log in
- add allowed users to AllowUsers

Install, configure and run fail2ban. Thanks to tecmint.

```
yum install fail2ban
```

```
vi /etc/fail2ban/jail.conf
```

```
systemctl start fail2ban
```

opening port 9998 for TIKA-1301

To open port 9998:
`firewall-cmd --zone=public --add-port=9998/tcp --permanent`

`firewall-cmd --reload`

permission management

1. `adduser <user> sudo`

2. `passwd <user>`

3. `groupadd <admingroup>`

4. `usermod -g <admingroup> <user>`

5. `modify /etc/ssh/sshd_config to add user to AllowUsers`

6. `systemctl restart sshd.service` --restart sshd on RHEL 7

mkdirs

/public/corpora/govdocs1 ...

prep govdocs1

1. `cp zipfilelist /public/corpora/govdocs1/archive`

2. `wget -i zipfilelist.txt`

3. (go get some coffee)

4. `cd govdocs1/scripts`

5. `groovy unzip.groovy 0`

6. (go get some more coffee)

7. `groovy rmBugged.groovy`

prep nsfpolardata

1. `scp -r <user>@nsfpolardata.dyndns.org:/usr/local/ndeploy/data/AcadisCrawl .`

2. (go get some coffee)

3. `scp -r <user>@nsfpolardata.dyndns.org:/usr/local/ndeploy/data/AcadisCrawl2 .`

4. (go get some coffee)

5. `scp -r <user>@nsfpolardata.dyndns.org:/home/mattmann/polar-data/nutch_trunk/runtime/local/bin/crawlId .`

6. (go get some coffee)

7. `cd /data1/public/archives/nsf-polar-data/`

8. `export NUTCH_OPTS="-Xmx8192m -XX:MaxPermSize=8192m"`

9. `./bin/nutch dump -outputDir out -segment /data1/public/archives/nsf-polar-data/acadis/AcadisCrawl/segments/`

10. `./bin/nutch dump -outputDir out2 -segment /data1/public/archives/nsf-polar-data/acadis/AcadisCrawl2/segments/`

11. `./bin/nutch dump -outputDir out3 -segment /data1/public/archives/nsf-polar-data/nasa-amd/crawlId/segments/`

add more disc

From Rackspace website, add block storage volume and attach it to server.

1. `mkfs.ext3 /dev/xvdb`

2. `mkdir /data1`

3. `mount /dev/xvdb /data1`

4. `nano /etc/fstab`
add these lines (e.g.):
`/dev/xvdb /data1 ext3 defaults 1 2`

`/dev/xvdc /data2 ext3 defaults 1 2`

4b. When you wreck the fstab file and can't log into your system after a hard reboot and you are in recovery mode:

`blkid` to figure out the drive types

`mount -t ext3 /dev/xvdb1 /mnt` to mount the system

4c. Before you hit 4b, try `mount -fav` to see if there are any errors in your fstab file.

httpd

config file in the usual place : /etc/httpd/conf/httpd.conf

1. Set robots.txt to disallow all: /var/www/html

2. Link data dir(s) under: /var/www/html

3. Configure config file to allow links and to show directories

4. Show long file names...add to config file: `IndexOptions FancyIndexing SuppressDescription NameWidth=*`

5. start: apachectl start

pdftotext

Downloads from: https://www.xpdfreader.com/download.html

Current version: 4.00; Released: 2017 Aug 10

1. Downloaded 64-bit Linux `XpdfReader`; executed: `XpdfReader-linux64-4.00.01.run`; unpacked and cp xpdf to /usr/local/bin

2. Downloaded 64-bit Linux `Xpdf tools`; unpacked and cp bin64/* to /usr/local/bin

3. Downloaded language support packages: Arabic, Chinese/simplified, Chinese/traditional, Cyrillic, Greek, Hebrew, Japanese, Korean, Latin2, Thai and Turkish; unzipped them all, cat all add-to-xpdfrc >> tmp_xpdfrc and cp all to /usr/local/share/xpdf

4. cat xpdf-tools-linux-4.00/doc/sample-xpdfrc tmp_xpdfrc >> /usr/local/etc/xpdfrc

NOTE: We found that *pdftotext* was not correctly reading the *xpdfrc* file in this location. We found no differences in extracted text when we removed the *xpdfrc* file and when we had it there. We did find a difference, especially in CJK PDFs, when we specified the *xpdfrc* file from the commandline with the *-cfg* option.

ffmpeg

1. `sudo yum install epel-release`

2. `sudo yum localinstall --nogpgcheck https://download1.rpmfusion.org/free/el/rpmfusion-free-release-7.noarch.rpm https://download1.rpmfusion.org/nonfree/el/rpmfusion-nonfree-release-7.noarch.rpm`

3. `sudo yum install ffmpeg ffmpeg-devel`

Other data

See ApacheTikaHtmlEncodingStudy for a description of gathering data for TIKA-2038. See CommonCrawl3 for a description of refreshing data for TIKA-2750. See ComparisonTikaAndPDFToText201811 for a comparison of text extracted from PDFs by Apache Tika/Apache PDFBox and pdftotext.

Security

Review bad logins or bad httpd requests:

```
sudo tail /var/log/secure
sudo vi /var/log/httpd/access_log*
Block ips
sudo vi /etc/hosts.deny
ALL: xxx.xx.xx.xxx
sudo systemctl restart sshd
```