Chapter 1. consolez

This document describes the z/VM-zLinux consolez project and is comprised of the following sections:

1.1 Description and design

'consolez' is an open source package that gives browser access to z/VM consoles and CP commands, while limiting access to certain teams. It can save time when teams have to interact with many z/VM LPARs. Rather than having to open many 3270 console sessions, all z/VM LPARs can be accessed through the same Web user interface.

You will need a zLinux server on each z/VM LPAR and a user that can do 'passwordless' SSH. LDAP authentication is recommended but not necessary.

There's a 14 minute video describing the first prototype here:

https://youtu.be/vD_6PXeJWdE

Following is a diagram of the consolez design.

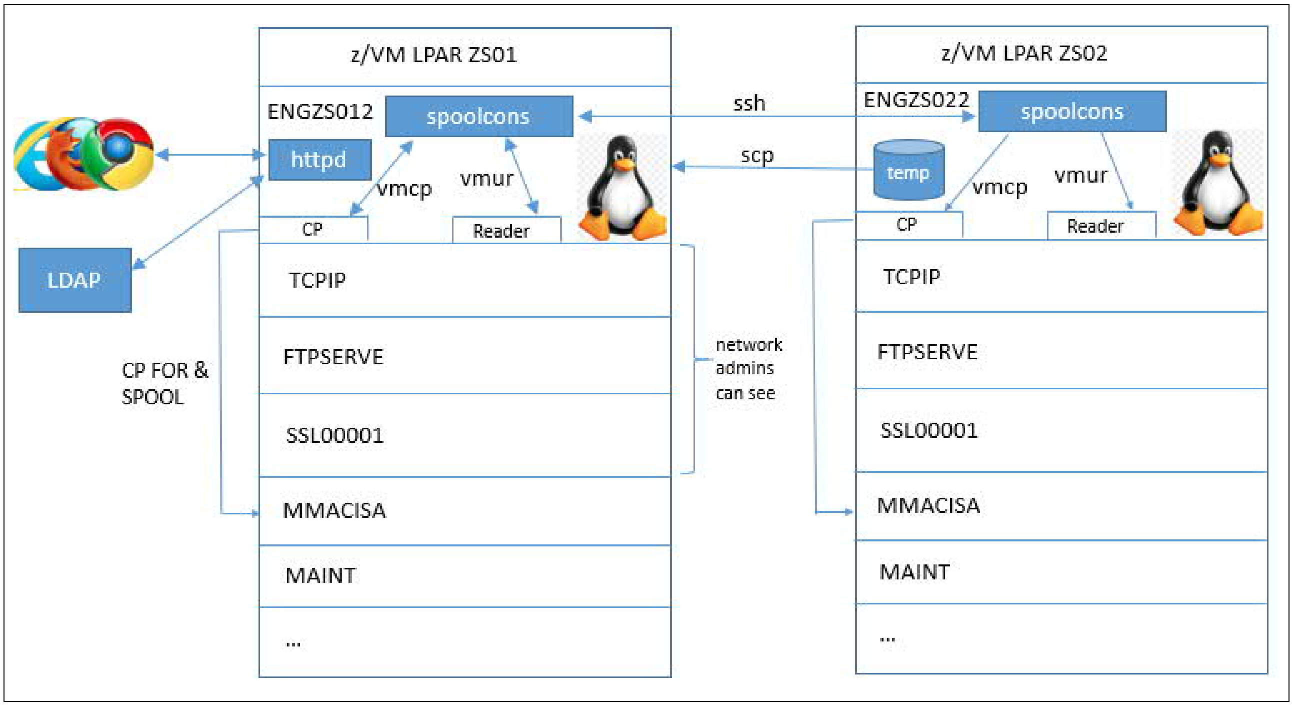


Figure 1-1 consolez architecture

The diagram shows two z/VM LPARs each with a zLinux server with elevated privilege classes. The C privilege class is needed for the **CP FOR** command. (TODO: test if just C and G are enough).

Each zLinux server needs to be able to **ssh** and **scp** to all other servers without needing a password. Therefore, both the private and public keys need to be one user's `$HOME/.ssh/` directory along with an `authorized_keys` file. (TODO: describe the setup process).

The line command **spoolcons** does the heavy lifting. It utilizes **vmcp** to issue CP commands and **vmur** to get files from the virtual reader. It utilizes the **CP FOR** and **SPOOL** commands to instruct other virtual machines on the LPAR to spool their consoles and send it to its virtual reader.

If the console to be spooled is on a different LPAR, the **spoolcons** on the local LPAR calls the same script on the target LPAR. After spooling the console, rather than copying the spooled file to the local file system, it is copied back to the main Linux server using **scp**.

You only need to run line commands and serve Web pages from the main Linux server. You may choose to have a warm backup server available for higher availability.

1.2 Line commands

Following are the consolez line commands:

- **catcons** – print a console (symlink)
- **consfuncs** – common functions
- **cpcommand** – run one CP command
- **grepcons** – search through saved console data (symlink)
- **lscons** – list save console files (symlink)
- **rmcons** – remove console for 1 VM (symlink)
- **spoolcons** – spool a console

All commands give help with the **-h** flag. For example:

```
# spoolcons -h

Name:  spoolcons - spool z/VM console data for a virtual machine
Usage: spoolcons [OPTIONS] USERID [SYSTEMID]
Where: USERID   is the virtual machine whose console is to be spooled
       SYSTEMID is the z/VM System Identifier name (default: this LPAR)

OPTIONS:
  -h|--help            Give help
  -s|--silent          Minimal output
  -v|--verbose         Include additional output
  -V|--veryverbose     Include even more output
  -x|--debug           Print commands and arguments as they are executed
```

1.3 Web pages

Following are the Web UI scripts. The default location is `/srv/www/ldap/`.

consolez	Main page showing all data
consolez.css	Cascading style sheets
consuifuncs	Common functions
cpcmds	Page to run some/all CP commands
onecons	Web page showing one console
onelpar	Page showing one z/VM LPAR
searchcons	Search saved console data

Following are some sample screen shots of the different pages.

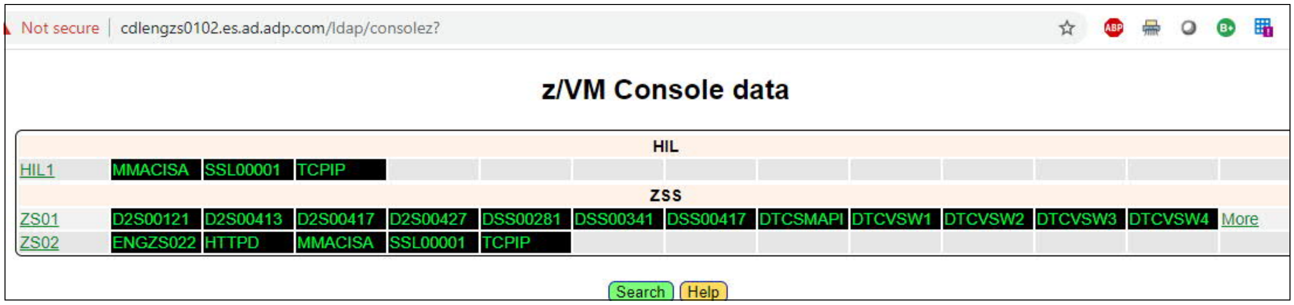
1.3.1 The consolez page

This is the main page named **consolez**. It shows one HTML table with two environments (`HIL` and `ZSS`), and three z/VM LPARs. Each of the table cells with green lettering and a black background represents a saved console file for one virtual machine. These are links which can be drilled down into the **onecons** page which shows all the console data for that virtual machine.

When there are more than 12 saved console files, the last column is set to 'More...'. That is a link as is the first column to drill down into that z/VM LPAR and show the **onelpar** page.

The bottom of the page shows two buttons:

Search	Allows you to search consoles
Help	Shows this document



Not secure | cdlengzs0102.es.ad.adp.com/ldap/consolez?

z/VM Console data

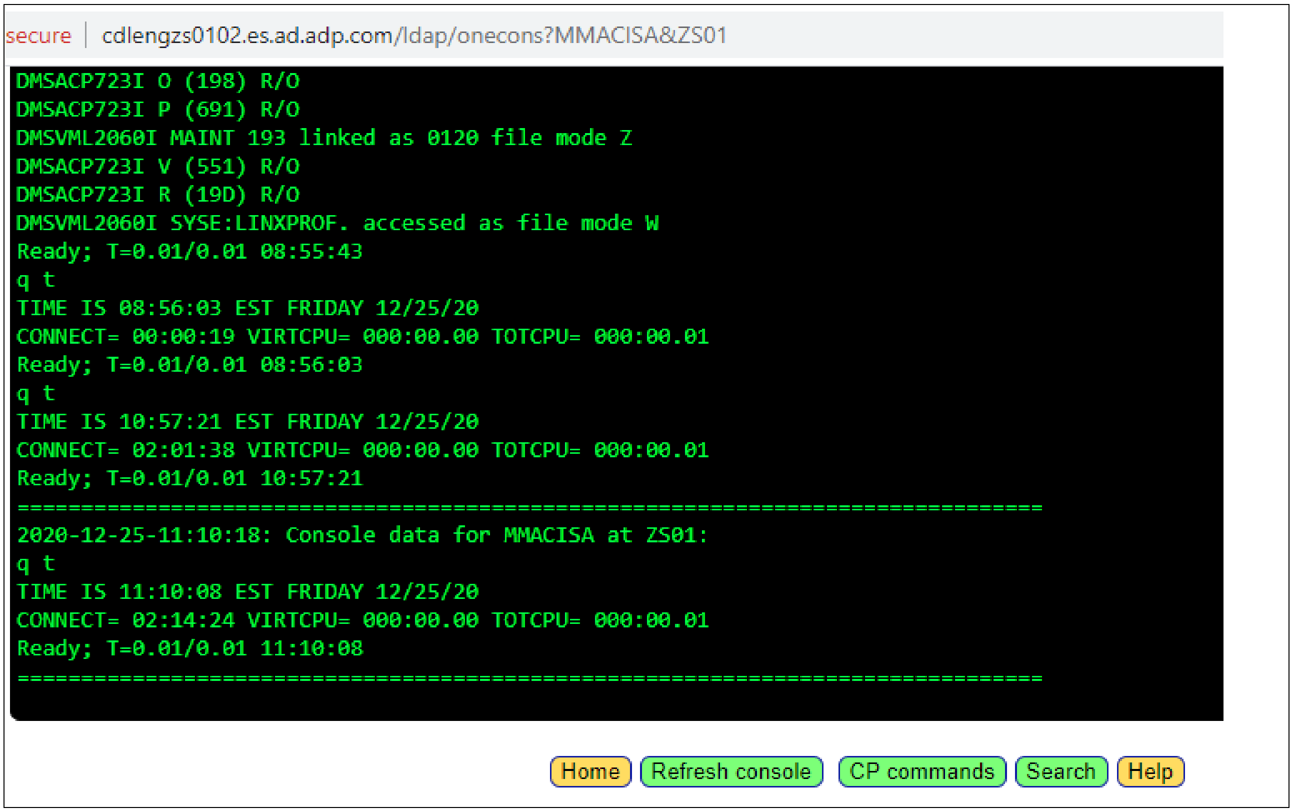
HIL													
HIL1	MMACISA	SSL00001	TCPIP										
ZSS													
ZS01	D2S00121	D2S00413	D2S00417	D2S00427	DSS00281	DSS00341	DSS00417	DTCSMAPI	DTCVSW1	DTCVSW2	DTCVSW3	DTCVSW4	More
ZS02	ENGZS022	HTTPD	MMACISA	SSL00001	TCPIP								

Figure 1-2 consolez main page

1.3.2 The onecons page

This is the page that displays all the saved console data for a single z/VM user ID. At the bottom of the page, the following buttons are shown:

Home	Returns you to the **consolez** page
Refresh console	Spools the console and refreshes the page
CP commands	Allows you to enter some or all CP commands
Search	Allows you to search all saved console data
Help	Shows this document



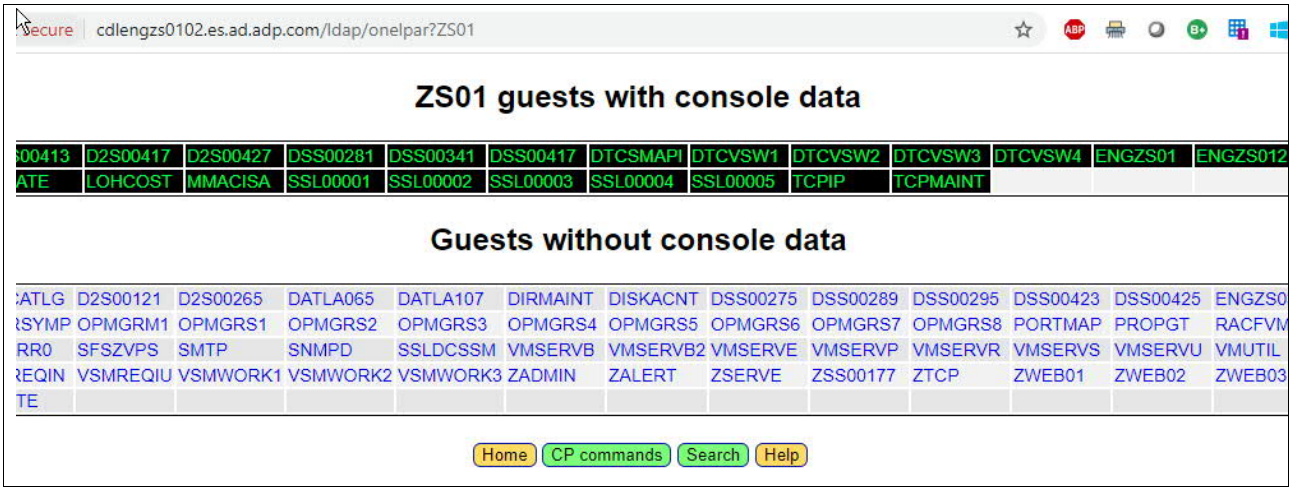
```
DMSACP723I O (198) R/O
DMSACP723I P (691) R/O
DMSVML2060I MAINT 193 linked as 0120 file mode Z
DMSACP723I V (551) R/O
DMSACP723I R (19D) R/O
DMSVML2060I SYSE:LINXPROF. accessed as file mode W
Ready; T=0.01/0.01 08:55:43
q t
TIME IS 08:56:03 EST FRIDAY 12/25/20
CONNECT= 00:00:19 VIRTCPU= 000:00.00 TOTCPU= 000:00.01
Ready; T=0.01/0.01 08:56:03
q t
TIME IS 10:57:21 EST FRIDAY 12/25/20
CONNECT= 02:01:38 VIRTCPU= 000:00.00 TOTCPU= 000:00.01
Ready; T=0.01/0.01 10:57:21
================================================================================
2020-12-25-11:10:18: Console data for MMACISA at ZS01:
q t
TIME IS 11:10:08 EST FRIDAY 12/25/20
CONNECT= 02:14:24 VIRTCPU= 000:00.00 TOTCPU= 000:00.01
Ready; T=0.01/0.01 11:10:08
================================================================================
```

Figure 1-3 Web page to show one saved console

1.3.3 The onelpar page

This is the page that shows all the virtual machines on one z/VM LPAR that have saved console data. Network administrators will only see the first HTML table. Full administrators will see a second table which shows virtual machines that are logged on, but do not have any saved console data. They can be drilled down into, their consoles will be spooled, and they will be moved from the lower into the upper table.

There are similar action buttons at the bottom of the page.



ZS01 guests with console data

00413	D2S00417	D2S00427	DSS00281	DSS00341	DSS00417	DTCSMAPI	DTCVSW1	DTCVSW2	DTCVSW3	DTCVSW4	ENGZS01	ENGZS012
ATE	LOHCOST	MMACISA	SSL00001	SSL00002	SSL00003	SSL00004	SSL00005	TCPIP	TCPMAINT			

Guests without console data

ATLG	D2S00121	D2S00265	DATLA065	DATLA107	DIRMAINT	DISKACNT	DSS00275	DSS00289	DSS00295	DSS00423	DSS00425	ENGZS0
SYMP	OPMGRM1	OPMGRS1	OPMGRS2	OPMGRS3	OPMGRS4	OPMGRS5	OPMGRS6	OPMGRS7	OPMGRS8	PORTMAP	PROPGT	RACFVM
RR0	SFSZVPS	SMTP	SNMPD	SSLDCSSM	VMSERVB	VMSERVB2	VMSERVE	VMSERVP	VMSERVR	VMSERVS	VMSERVU	VMUTIL
EQIN	VSMREQIU	VSMWORK1	VSMWORK2	VSMWORK3	ZADMIN	ZALERT	ZSERVE	ZSS00177	ZTCP	ZWEB01	ZWEB02	ZWEB03
TE												

Figure 1-4 Web page to show consoles for one z/VM LPAR

1.3.4 The cpcmds page

This page enables users to run CP commands. Again, there are two views. Full administrators will see the view on the left which enables them to run any CP command. Network administrators will see the view on the right which allows them to run a small subset of CP commands related to the status of the network.

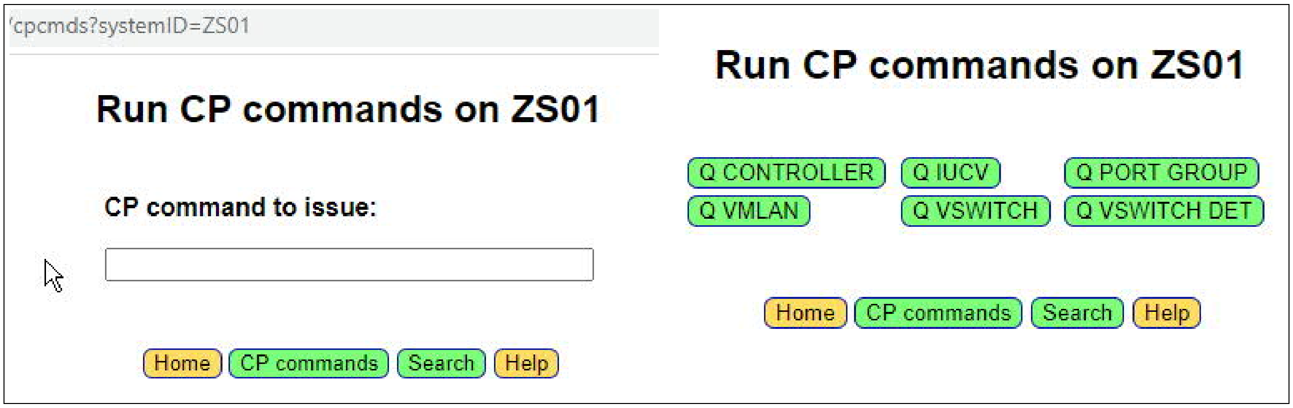
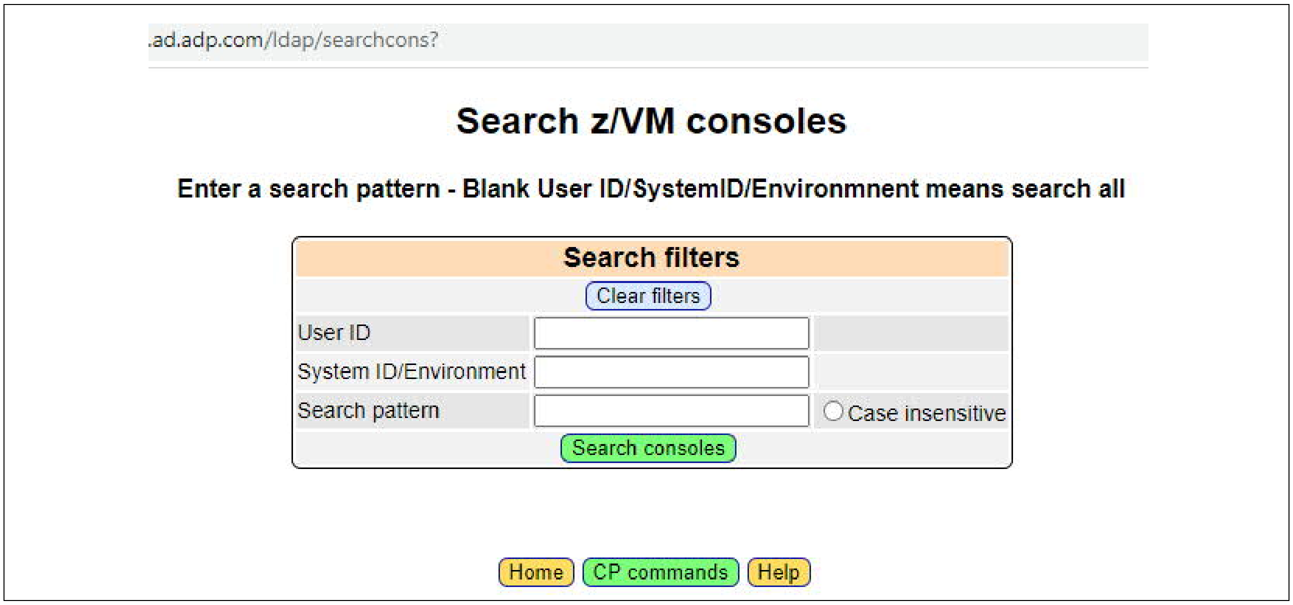


Figure 1-5 Page to issue CP commands - full and network administrator’s views

1.3.5 The searchcons page

This page allows all users to search all saved console data. In the first field, a specific user ID or a pattern can be entered to limit the search to certain virtual machines. Similarly in the second field a specific z/VM LPAR or pattern, or a single environment can be entered to limit the search to those z/VM LPARs. If these fields are left blank, all user IDs and LPARs are searched.

The last field is the pattern to search for. There is a radio button to the right if text case is not to be considered.



.ad.adp.com/ldap/searchcons?

Search z/VM consoles

Enter a search pattern - Blank User ID/SystemID/Environmnent means search all

Figure 1-6 Web page to search console data

1.4 Install and customize Apache

To prepare for a zoom RPM install, perform the following steps:

- Start an SSH session to the RHEL server as root.
- Install the Apache Web server and other co-requisite packages:

```
# yum -y install httpd httpd-manual tree vim bind-utils mod_ldap mod_ssl mlocate postfix
      ...
```

1.4.1 Configure Apache

Apache need only be installed and configured on one (or two if you want a backup) zLinux server. This section describes how to configure Apache on a RHEL 8 system. Other Linux distributions will differ slightly.

You should probably use encrypted (https) communication. To do so, you'll need a certificate.

Create certificate signing requests

You must create a certificate signing request to have your certificate signed. All of the files created are named server.* so the steps will work on any server.

- Change directory to /etc/httpd and create three directories for the key, the certificate signing request and the certificate.

```
# cd /etc/httpd
# mkdir ssl.key ssl.csr ssl.crt
```

- Change directory to ssl.key.

```
# cd ssl.key
```

- Create a private key file server.key with the following **openssl** command.

```
# openssl genrsa -out server.key 3072
Generating RSA private key, 2048 bit long modulus
.+++
....................................................+++
e is 65537 (0x10001)
```

- Change directory to /etc/httpd/ssl.csr/. This is where certificate-signing requests (CSR) are kept.

```
# cd /etc/httpd/ssl.csr
```

- Create a CSR in the file apache.csr with the following **openssl** command. Answer all the questions. The most important field is the Common Name - it must be the fully-qualified host name. Do not include the optional challenge password or company name.

```
# openssl req -new -key ../ssl.key/server.key -out server.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
```

```
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

> Type the contents of the CSR file on to your clipboard.

```
# cat server.csr
-----BEGIN CERTIFICATE REQUEST-----
...
-----END CERTIFICATE REQUEST-----
```

> Copy the contents to your clipboard.

You should now have the certificate signing request (CSR) to send to a Certificate Authority.

Create the certificate

TODO Is there a free certificate creation site?

For now, you have to find a way to get a certificate created from your CSR.

Copy the certificate to the zLinux server

The certificate can now be copied back to the zLinux server.

> Copy the file from your workstation with Putty **pscp** or WinSCP. For example.

```
C:\> pscp.exe certnew.cer <yourServer>:
```

> Go back to the SSH session on the Linux server as root. Change to your home directory.

```
# cd ~macisaam
```

> Copy the certificate file certnew.cer to the /etc/httpd/ssl.crt/ directory.

```
# cp certnew.cer /etc/httpd/ssl.crt
```

> Convert the certificate chain file (.p7b) into a certificate with the following **openssl** command. This puts the certificate chain into the certificate file.

```
# cd /etc/httpd/ssl.crt
# openssl pkcs7 -print_certs -in certnew.cer -out server.cer
```

> Show the contents of the file. You should see a number of certificates.

```
# cat server.cer
subject=/C=US/ST=New Jersey/L=Roseland/O=ADP/OU=vmlinux/CN=cdlenghil102.es.ad.adp.com...
issuer=/DC=com/DC=ADP/DC=AD/DC=ES/CN=ADP Internal Issuing CA 01
-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE-----

subject=/DC=com/DC=ADP/DC=AD/DC=ES/CN=ADP Internal Issuing CA 01
issuer=/CN=ADP Internal Policy CA 01
-----BEGIN CERTIFICATE-----
...
```

You now have a valid, signed certificate for your server.

Configure Apache to use the certificate

Now the certificate can be referenced by the Apache configuration files.

– Change to the /etc/httpd/conf directory. Make a backup copy of the httpd.conf file.

```
# cd /etc/httpd/conf
# cp httpd.conf httpd.conf.orig
```

– Edit the httpd.conf file. The default document root on RHEL is under /www/, not under /srv/ as with SLES. This can easily be changed in the httpd.conf file.

```
# vi httpd.conf
...
# change /www to /srv
   <Directory "/srv/www">
       AllowOverride None
       Require all granted
   </Directory>
   <Directory "/srv/www/html">
       Options Indexes FollowSymLinks
       AllowOverride None
       Require all granted
   </Directory>
   ....
   <IfModule alias_module>
       ScriptAlias /cgi-bin/ "/srv/www/cgi-bin/"
   </IfModule>
   <Directory "/srv/www/cgi-bin">
       AllowOverride None
       # Options None
       Options FollowSymLinks
       Require all granted
   </Directory>

   # Start ----- local customizations ------
   # use https
   SSLEngine on
   SSLCertificateKeyFile /etc/httpd/ssl.key/server.key
   SSLCertificateFile /etc/httpd/ssl.crt/server.cer

   # ldap/ directory is for other LDAP-protected scripts
   ScriptAlias /ldap/ /srv/www/ldap/
   <Directory /srv/www/ldap/>
       AllowOverride None
       Options +ExecCGI -Includes
       AuthType Basic
       AuthName "System z self-service - Enter your LDAP credentials"
       AuthBasicProvider ldap
       AuthLDAPURL
   ldap://<your.ldap.example.com>:389/ou=people,ou=linux_systems,dc=example,dc=com?uid
       Require ldap-filter objectClass=posixAccount
   </Directory>
   ...
```

Enable cgi-bin scripts to run

It seems Apache on RHEL does not allow cgi-bin/ scripts to be run out of the box.

I found this comment on the Internet and chuckled:

> “{/var,}/run is vaporized by systemd on boot. It's a wonderful and entirely unexpected surprise with such low value.”

Here is work-around that seems to work:

> Set Apache to run as zadmin:

```
# cd /usr/lib/tmpfiles.d
# cp httpd.conf httpd.conf.orig
# vi httpd.conf
...
# diff httpd.conf httpd.conf.orig
1c1
< d /run/httpd   710 zadmin vmlinux
---
> d /run/httpd   710 root apache
```

> Make zadmin the owner of the /var/run/httpd/ directory.

```
# chown zadmin /var/run/httpd
```

This should the ownership of /var/run/httpd/ at boot time.

- Start Apache.

```
# service httpd start
...
```

–

```
# systemctl enable apache2
...
```

Your should now have Apache configured.

1.5 Install and customize consolez

The consolez source code is currently available at:

https://sites.google.com/site/mike99mac/consolez-0.94.tgz

An RPM is now available at:

https://sites.google.com/site/mike99mac/consolez-0-94.s390x.rpm

If you download both, you have a choice on how to install. I would recommend the latter.

1. Copy the tar file to root directory. (**Note**: this has not been tested)

```
# cd /
# tar xzf consolez-0.94.tgz
...
```

2. Install the RPM as root.

```
# rpm -i consolez-0-94.s390x.rpm
...
```

Eventually, the code will reside on github as I’m told that is “where it’s at” :))

https://github.com/mike99mac/consolez

To customize consolez, perform the following tasks.

– Create the `/etc/consolez.conf` file from the sample provided.

```
# cd /etc
# cp consolez.conf.sample consolez.conf
```

– Edit the file, setting `consolezDir`, `consolesUser` and a description of your environments, z/VM LPARs, and zLinux engineering servers.

```
# vi consolez.conf
#
# Configuration file for consolez - /etc/consolez.conf
#
# Variable 'consolesUser' must be set to a user that can SSH without passwords
# Variable 'consolezDir' can be set but if not will default to /srv/consolez
#
consolezDir = "/data/consolez"             # directory where console data is stored
consolezUser="zadmin"                      # user with passwordless SSH consolez
will run as
#
# Remaining values define your organizations environments, Linux 'engineering
servers' and z/VM systems
#
# For example, to define two environments each with two z/VM LPARs:
# environment ENV1
# hostname1 zvm1
# hostname2 zvm2
#
# environment ENV2
# hostname3 zvm3
# hostname4 zvm4
#
```

consolz should now be configured.

1.6 Use consolez

You should now be able to use consolez.

– If you have /usr/local/sbin/ in your PATH, you should be able to issue a command to show help.

```
# which spoolcons
/usr/local/sbin/spoolcons
# spoolcons -h

Name:  spoolcons - spool z/VM console data for a virtual machine
Usage: spoolcons [OPTIONS] USERID [SYSTEMID]
Where: USERID   is the virtual machine whose console is to be spooled
       SYSTEMID is the z/VM System Identifier name (default: this LPAR)

OPTIONS:
  -h|--help            Give help
  -s|--silent          Minimal output
  -v|--verbose         Include additional output
  -V|--veryverbose     Include even more output
  -x|--debug           Print commands and arguments as they are executed
```

– Now try issuing a **spoolcons** command

```
# spoolcons tcpip
...
```

– If that succeeded you should be able to see that one console file is saved with the **lscons** command.

```
# lscons
<ENV>:<LPAR> TCPIP
```

– Show the contents of that spool file.

```
# catcons TCPIP
...
```

– Search through the file for a pattern.

```
# grepcons tcpip : time
...
```

– View the consolez Web page through a browser

```
https://<your.zlinux.example.com>/ldap/consolez
```

You should see the Web page and be able to naviagate the handful of Web pages.