MVS Operations

What is it?

This tool was written to make it easier perform common operational tasks with MVS images, such as tk4- , running under Hercules. It does this by providing a web interface for common tasks such as: viewing the MVS console, viewing and downloading printout for individual jobs and submitting jobs via the card reader.
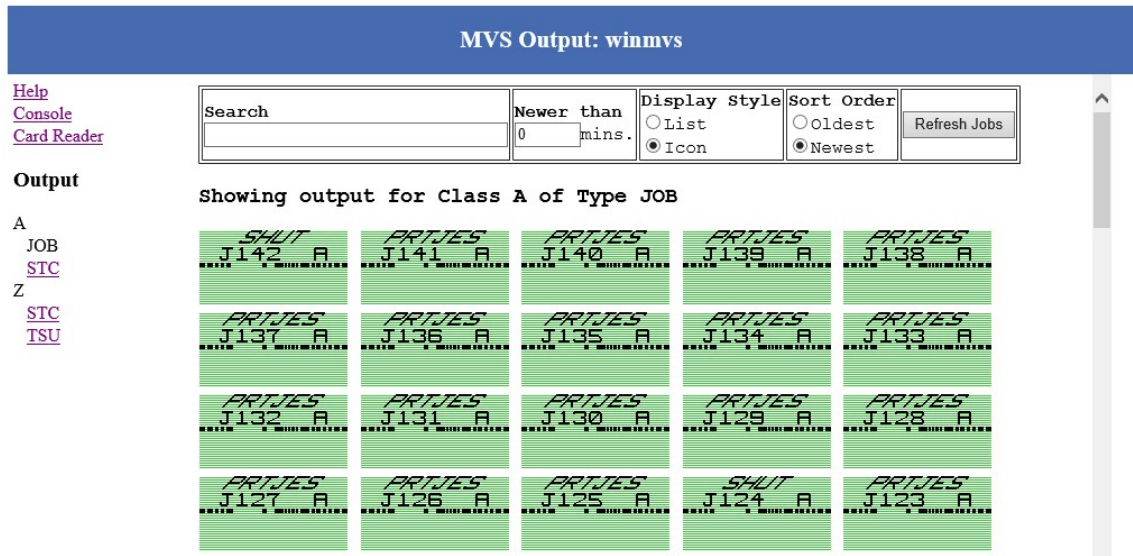
This suite has been tested against tk4- on LINUX (including Raspberry pi) and Windows. Currently there is a running example hosted at https://mvs.gadsby.me.uk

Example Screen Shots

i)   The Main Entry screen showing the potential selection of different MVS images and the entry point to the Console, Card Reader and Output panels.

ii)   The Output screen showing the print output for JOBs in Class A. Clicking on a job icon will open the output from that job.

iii) Output as shown on the screen for a job. Navigation is by page number or by selecting from a page range. The actual printout can be downloaded via the links at the top of the page.

iv) PDF download from the job output screen showing the "green bar" line paper listing. This is available in portrait (2 up) or landscape mode.

v)   The Card Reader screen showing how pre-defined Code Template jobs can be selected using buttons on the left with the MVS user and password entry shown. Card decks can also be selected from the user's local system

MVS Reader: winmvs

Help   Console   Output

Code Templates

Local JCL
asm
asmhasp2
c
cob
iebgener
pas
shutdown

Enter Job Below

Submit Job   Clear cards   %user%= herc01   %pass%= ●●●●●●

```
//AHELLO    JOB USER=%user%,PASSWORD=%pass%
//HELLO  EXEC ASMFCLG
//ASM.SYSUT1 DD UNIT=SYSDA
//ASM.SYSUT2 DD UNIT=SYSDA
//ASM.SYSUT3 DD UNIT=SYSDA
//ASM.SYSGO DD UNIT=SYSDA
//ASM.SYSIN DD *
HELLOA   CSECT
         USING HELLOA,15
         SAVE (14,12)
         WTO 'HELLO WORLD! FROM ASSEMBLER'
         RETURN (14,12),RC=0
         END
/*
```

vi)   The MVS Console showing the scrolling console log. Optionally, this allows entry of MVS commands using the Hercules console interface, command entry is via the command entry bar at the base of the screen.

MVS Console: winmvs

Help   Pause   Show current console   Update Interval 5   seconds

```
HHC01603I * ZZZzz /,'.-'`'      -.  ;-;;.    **       ****       ||   ||
HHC01603I *      |,4- ) )-,_.,( ( ''-'   **       *****         ||   ||
HHC01603I *      '---''(_/--' `-')_)      **       ** **        ||   ||  |||||||||||
HHC01603I *                               **       **  **      ||||||||||  Update 08
HHC01603I *       The MVS 3.8j            **       **   **     ||||||||||
HHC01603I *    Tur(n)key System           **       **    **          ||
HHC01603I *                            ******     ****   ***       |||||
HHC01603I *
HHC01603I *        TK3 created by Volker Bandke      vbandke@bsp-gmbh.com
HHC01603I *        TK4- update by Juergen Winkelmann  winkelmann@id.ethz.ch
HHC01603I *              see TK4-.CREDITS for complete credits
HHC01603I *
HHC02264I Script 5: file scripts/tk4-.rc processing ended
HHC01040I 0:000C COMM: client mvs, ip 127.0.0.1 connected to device 3505
HHC01206I 0:000C Card: client mvs, ip 127.0.0.1 disconnected from device 3505
10.38.02 JOB  126  $HASP100 PRTJES   ON READER1
10.38.02 JOB  126  $HASP373 PRTJES    STARTED - INIT  1 - CLASS A - SYS TK4-
10.38.02 JOB  126  IEF403I PRTJES - STARTED - TIME=10.38.02
10.38.02 JOB  126  $HASP375 PRTJES    ESTIMATED CARDS EXCEEDED
10.38.02 JOB  126  IEF404I PRTJES - ENDED - TIME=10.38.02
10.38.02 JOB  126  $HASP395 PRTJES    ENDED
10.38.02           $HASP309    INIT  1 INACTIVE ******** C=A
10.38.02 JOB  126  $HASP150 PRTJES    ON PRINTER1         45 LINES
10.38.02           $HASP160 PRINTER1 INACTIVE - CLASS=A
```

D T   ×   Enter
Ready

How does it Work?

There are two components to the MVS Operations toolkit.

- **hercprt** which collects the output from the MVS printer devices and splits them into individual jobs into a directory tree on the server. To collect the output the print devices defined in the Hercules configuration file need changing from the normal file definition to a sockdev definition. E.g.

```
#
# unit record devices
# Comment out old print devices
#0002 3211 prt/prt002.txt ${TK4CRLF}
#000E 1403 prt/prt00e.txt ${TK4CRLF}
#000F 1403 prt/prt00f.txt ${TK4CRLF}
# replace with socket definitions such as
0002 3211 localhost:3202 sockdev
000E 1403 localhost:3214 sockdev
000F 1403 localhost:3215 sockdev
```

  The **hercprt** script is normally invoked in the background before Hercules is started. See the section **HERCPRT Manual Page** for further details on the available options.

- A collection of php scripts used by a webserver, such as Apache, which interface to the standard Hercules console file, card reader socket and accesses the individual print files, within the directory tree, created by **hercprt**.

  These web scripts are controlled by a configuration file **lib/config.php** which is tailored at install time to set the appropriate values to match the Hercules configuration file.

Security

By default there is NO security in place within MVS Ops so any user who can access the webserver hosting this application can see all output, submit any jobs [they do need to know a valid MVS user/password] and even enter MVS console commands.

User authentication can be turned on at configuration time, see the Adding User Authentication section.

Cookies are used to hold settings of key state e.g. MVS image selected, print style, etc., between sessions.

Pre-Requisites

To get MVS Operations working you need the following components installed:

- A working Hercules MVS instance. Tk4- is ideal. You will need to be able to change the Hercules configuration file to update the printer device entries.
- A working web server with **php** and the **php-gd** and **php-curl** libraries installed (on Windows WAMPSERVER http://www.wampserver.com/ has been tested and works well). You will need to be able to add files to the html directory on this webserver or create a new site to host the web content.
- A working **perl** installation (on Windows Strawberry Perl http://strawberryperl.com/ has been tested and works well).
- Sufficient disk space to hold the split print jobs that is accessible for **hercprt** to write to and the webserver to read from.

Installation Steps

Before you start it may be useful to fill in this table for your configuration as these values will be required throughout the steps below:

Name	Usage *Example*	Your Value
SITE NAME	Identifies the MVS image selected *MYMVS*	
DESCRIPTION	Long name for site *My TK4- system on LINUX*	
CONSOLE	Hercules Console log file */users/mvs/log/3033.log*	
READER	Hercules reader socket *localhost:3505*	
HERCCONS	Hercules console command socket *localhost:8038*	
PRTDIR	Directory for **hercprt** to place output */usr/tmp/PRT*	
JCLDIR	Directory containing JCL templates */users/mvs/jcl*	
TAPEDIR	Directory to upload tape images */usr/tmp/TAPE*	

i)   Extract the bin, jcl and www directories from the download file which can be found at
       mvs.gadsby.me.uk into a local directory.

LINUX

ii)   Copy **hercprt** file from the downloaded bin directory to ***/usr/local/bin*** , or somewhere else on your
       PATH, and make sure it is executable e.g. **chmod 755 /usr/local/bin/hercprt**.

iii)   Edit your Hercules **mvs** start script to insert the following bold lines BEFORE the hercules start line:

```
export HERCULES_RC=scripts/ipl.rc
# start print splitter ready for MVS startup
/usr/local/bin/hercprt -f conf/tk4-.cnf -o /usr/tmp/PRT
nohup $force_arch hercules $DAEMON -f conf/tk4-.cnf >log/3033.log &
```

       Change references to conf/tk4-.cnf to match the config file passed to Hercules and */usr/tmp/PRT*
       (PRTDIR in the table above) to the directory where you want the burst output files to be written.

Windows

ii)   Place **hercprt** from the downloaded bin directory into a location that is accessible from your mvs.bat
       script.

iii)   Edit your **mvs.bat** start script to insert the following line BEFORE the hercules start line (change
       conf\tk4-.cnf below to match the config file passed to Hercules):

```
SET TK4CRLF=CRLF
# start print splitter ready for MVS startup
start perl \users\mvs\bin\hercprt -f conf\tk4-.cnf -o \temp\PRT
.\hercules\windows\%ARCH%\hercules %DAEMON% -f conf\tk4-.cnf >log/3033.log
```

       Change references to conf\tk4-.cnf to match the config file passed to Hercules and *\temp\PRT*
       (PRTDIR in the table above) to the directory where you want the burst output files to be written.

General

iv)   Edit your Hercules config file, e.g. *conf/tk4-.cnf* , and change the lines for print devices of type 3211 and 1403 used for JES2 job output classes e.g. 002, 00E and 00F to:

```
0002 3211 localhost:3202 sockdev
000E 1403 localhost:3214 sockdev
000F 1403 localhost:3215 sockdev
```

Where the numbers after localhost are arbitrary TCP socket numbers.

v)   Start Hercules using the modified start scripts and confirm that print files are generated in the PRTDIR. The structure under PRTDIR should look something like Z/STC A/JOB etc. If not then:
   a. Review the Hercules log for any errors relating to the changed devices,
   b. Add the -d flag to the hercprt line above and restart Hercules. You'll get more diagnostic information that should help diagnose the issue.

vi)   Link or copy the **www** directory contents from the download into your webserver html directory or similar.

Make sure that the copied/linked directories and files are readable and NOT writeable by the webserver.

vii)   Ensure that the example jobs in the **jcl** directory can be read by the webserver php scripts. This is JCLDIR

viii)   Edit *www/lib/config.php* to reflect the actual MVS image(s) that you have and the locations of the directories, reader and console input end points from the table above. E.g.:

```
"lnxmvs" => array(
        "Description" => "Linux MVS image",
        "CONSOLE"      => "/home/mvs/log/3033.log", # console log file for MVS instance
        "READER"       => "localhost:3505",       # socket for Hercules RDR for this instance
        "HERCCONS"     => "localhost:8038",       # Hercules HTTP_PORT function to submit MVS command
        "PRTDIR"       => "/usr/tmp/prt",         # directory of output files split by hercprt (-o flag)
        "JCLDIR"       => "/home/mvs/jcl"         # JCL sample directory
            ),
```
Remove any entries for any MVS images that are not installed.

ix)   Setup is now completed so test the following facilities to confirm that all is well.

TEST: Connect to your webserver, go to the start page for the created site and you should see the start page like that shown in image (i) in this document. If you have defined multiple sites you should see a site selector.

If not check lib/config.php contains at least one complete entry in the sites array.

TEST: Select the console link, you should see the current console dialogue displayed. It should also scroll every 5 seconds if more console output has been written.

If not check the file CONSOLE matches the Hercules configuration file.

TEST: In the command line entry box enter "D T" (no quotes) and <enter>, within 5 seconds you should get the time displayed.

If you get a 500 error you've probably not installed php-curl.

If nothing happens verify that HERCCONS in lib/config.php matches the Hercules definition in tk4-.cnf, or similar.

TEST: Select the reader link, enter a valid MVS **user** and **password**. Select the ASM jcl box on the left and an assembler job deck should be displayed. Hit **submit** and confirm that the job is submitted.

If you get a connection refused check that the READER socket is correct in your **config.php** and that MVS is running.

If no code samples are displayed check that the JCL directory in config.php is correct and that the files can be read by the webserver user.

TEST: Select the output link, click on a class entry (A/JOB, Z/STC etc) and output jobs should be shown as clickable icons. Check that the jobs open and display. Test the download lst and pdf links to confirm that output can be downloaded).

If no icons are displayed, just boxes, it is highly likely that **php-gd** has not been installed.

If there are no classes or job types displayed check that **config.php** PRTDIR and the -o option on hercprt command in the mvs start script are the same and that a directory tree of output files is actually present within that directory. Also, these files must be readable by the webserver user.
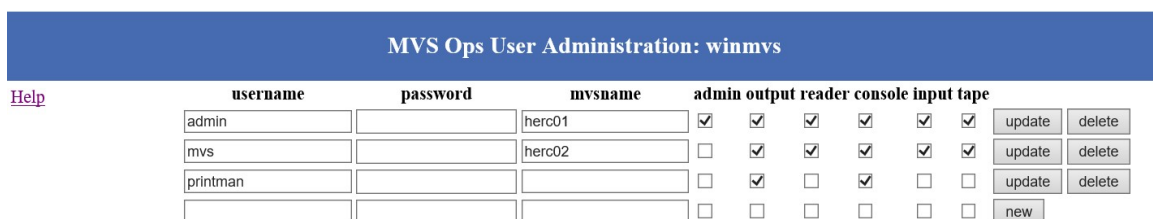
Adding User Authentication

It is possible to turn on basic user authentication to prevent unauthorised users accessing the various screens. To enable this feature, you'll need a MySql instance available within your LAMP/WAMP environment along with the ability to create a new database and user within that instance.

To enable this feature perform the following steps:

i)   Connect to your MySql data base as user, such as "root" which sufficient privileges to create a database and user. Review the ***createdb.sql*** script provided as part of the mvs_ops installation and change the "**mvs_ops_user**" username and "**mvs_secret**" password to suitable local values, you can also change the "**mvsops**" database name if required. Run the ***createdb.sql*** script and ensure that there are no errors.

ii)   Copy ***www/lib/db.php.SAMPLE*** to ***www/lib/db.php*** and set the values to match those set in step (i).

iii)   Edit **www/lib/config.php** and change the line

          define("SECURED", false);

to

          define("SECURED", true);

iv)   Start a new MVS Ops web session and you should see the start screen with a user name and login entry on the top right hand side. Enter the pre-configured username of "**admin**" password "**admin123**".

You will be shown as logged in as "**admin**". Now select the Admin link on the left hand of the screen, this will take you to a screen to modify user attributes, add new users, etc.. See the help screen for more information.

Troubleshooting

Q:	In the printer listing only filenames are displayed instead of icons on the first page.
A:	Check that Display Icon is selected. If so, it is likely that the **php-gd** library has not been installed. On LINUX try doing *apt-get install php5-gd* or *apt-get install php7.2-gd* (depending on your version of php) and restart your webserver.
Q:	On the console screen I'm getting an error 500 when I try and submit a command.
A:	This can be caused if the php curl library has not been installed. On LINUX try doing *apt-get install php-curl* and restarting your web server.
Q:	My print jobs don't appear to be split.
A:	Have you changed the default banner line for your job separator page within JES2? The **hercprt** script looks for lines matching :
	****A **START** JOB 190 JES2JOB GENERATE OS/VS2 HASP **ROOM** 1234 6.04.49 PM 26 MAY 17 PRINTER1 SYS TK4- JOB 190 **START** A********
	The items in **bold** must match with END replacing START at the end of jobs and CONT replacing START if a restart is made mid print. The regex used, by default, is based on:
	/^****[A-Z0-9] *START.*ROOM.*START *[A-Z0-9]****$/
Q:	I'm running on LINUX and cannot get any card reader jobs submitted and there's no printout being generated.
A:	The sockdev can be problematic on later LINUX releases – run a "dmesg" and see if you have any messages related to SYN / SYN Cookies. If you see SYN related error messages, then try turning on SYN cookie handling:
	echo "1" > /proc/sys/net/ipv4/tcp_syncookies
	you'll need to have a root session to do this. After the change restart Hercules.
Q:	Several banner start and end pages are contained in some printouts. Why is this happening? I'd prefer to get each printout in a separate file?
A:	Long running jobs in MVS often create new print files on a regular basis, e.g. the MF1 job creates a summary of activity every 15 minutes. By default, **hercprt** appends these subsequent pages to the original job output. If you wish to change this behaviour use the "**-m**" flag when starting **hercprt**, this causes **hercprt** to create a new file with the file count being appended to the job number _1, _2, _3 etc. for each discrete printout.
Q:	I'm stuck / I've found a bug! How can I get more help?
A:	Send an email to mvs@gadsby.me.uk and I'll do my best to help you get going. Please note I cannot help you get your basic webserver or Hercules MVS image up and running.

Q: If I leave the web session screens untouched for a period of time I get a "You don't have sufficient privileges to access the xxxx." Message when I come back to the system. I'm running on LINUX.

A: /usr/lib/php/sessionclean is being run by a cron entry in /etc/cron.d/php at 9 and 39 minutes past the hour, this deletes session files that have not been updated in the last session.gc_maxlifetime seconds, by default that is set to 1440 in the php.ini [24 minutes]. Once the session file held in /var/lib/php/sessions is removed the session is effectively dead and MVSops removes your access to the low level screens - the main screen will revert to a login.

The solution is at the LINUX level, either:

i)   change the session.gc_maxlifetime=1440 in your php.ini (/etc/php/7.3/apache2/php.ini) to a higher value, e.g 28800 is 8 hours, or
ii)   change the /etc/cron.d/php to run sessionclean less frequently - maybe once a day at 03:09AM [09 03 * * *].

However, if this is a public facing system long uncleared sessions are potentially a security exposure - if this is the case setting session.gc_maxlifetime to around 2 hours is a reasonable compromise, but only you can decide.

hercprt - read printer output from Hercules/MVS on a socket or FIFO and split jobs into separate files

Synopsis
 Collect files from Hercules/MVS as they are being produced and split them into separate files
 that can be viewed using other processes e.g. web browser. The functionality mimics a human
 operator using the start and end banner pages to split the print stream.

 hercprt can read data from multiple printers, it does this by creating a process for each printer
 port. Sockets are used rather than pipes so no data buffering occurs and each line can be processed
 as it is printed by MVS.

 The output files are written into a directory structure based on Job Class, Job Type, Job Number.
 For example, JOB 97 printed to class A from a TSO user will be in outdir/A/JOB/97_0.lst and
 Started Task 251 printing to class B will be in outdir/B/STC/251_0.lst.
 If a job has multiple steps then output will appear with the numeric suffix incremented once per
 segment if the -m flag is used otherwise all output will appear in the _0.lst file.

Usage
 Modify the Hercules .cnf file for each printer that you wish to process to write to a sockdev.

 e.g. For TCP sockets (this is the recommend method)
        0002 3211 localhost:3202 sockdev
        000E 3211 localhost:3214 sockdev
        000F 3211 localhost:3215 sockdev

 For local UNIX Domain sockets
        0002 3211 /tmp/prt0002 sockdev
        000E 3211 /tmp/prt000E sockdev
        000F 1403 /tmp/prt000F sockdev

 Once Hercules is started start hercprt and it will spawn off a process for each defined printer socket

        hercprt localhost:3202 localhost:3214 localhost:3215
 or
        hercprt /tmp/prt0002 /tmp/prt000E /tmp/prt000F
 or
        hercprt -f herc_config_file

Options
 hercprt [-d] [-o dir] [-e log] [-u name][-w count][-f config_file]
        [host:port || hostsocket || input_file]....

 -o dir Output directory to use for writing files e.g. -o /var/spool/mvs
        DEFAULT: /user/mvs/prt/burst

 -e log Change where errors are written to
        DEFAULT: STDERR

 -u name Changes the effective user to that of the user specified this means output files
        can be created so they are owned by another user e.g. www-data
        NOTE: only valid if effective user id is root to start with

 -w count Wait for the connection to the printer port for up to count minutes
        NOTE: This allows hercprt to be started before Hercules and wait for printers to come online
        DEFAULT: 5 to give time for Hercules to start, if set to zero will only try once with no pause

 -m   Set this flag if you want a multipart printed job to be split into separate files
        DEFAULT: output from multiple steps are sent to the same output file

 -f config_file
        Read the printer definitions directly from the Hercules config file
        Only 3211 and 1403 SOCKDEV entries are matched and processed

 -d   Turn debug on. If set, will send debug information to print to stdout

 tcp socket spec, UNIX domain socket (file socket) or inputfile
        One or more of these may be specified separated by spaces

        host:port   TCP socket on host
        hostsocket   UNIX Domain soket on this host (special file of type socket)
        input_file   A regular file containing data that will be processed immediately

Example
  For the UNIX Domain Hercules config file above (setting uid to web server)
          hercprt -o /var/spool/prt -u www-data /tmp/prt0002 /tmp/prt000E /tmp/prt000F

  For the TCP example above
          hercprt -o /var/spool/prt localhost:3202 localhost:3214 localhost:3215

  To make use of the existing Hercules config file and print some debug information
          hercprt -d -f conf/tk4-.cnf -o /usr/tmp/prt

JCL Sample Jobs

Several sample JCL jobs are supplied. These jobs, which have been tested on tk4-, are briefly described in the table below.

Job Name	Description
asm	Assembler version of hello, world
asmhasp2	Rebuild of the JES2 sub-system. This replaces the running JES2 image and is picked up on the next IPL. This generates a lot of output (4045 pages).
c	C version of hello, world
cob	COBOL version of hello, world
iebgener	Prints a file from a JES2 source PDS onto the A output class using IEBGENER
pas	PASCAL version of hello, world. Note how the program entry on line 7 needs OUTPUT defining to produce any output.
shutdown	Starts a full shutdown of the system. This must be run from a suitably privileged user such as HERC01.
tape	Printing a dataset from a tape

The CARDREADER tab on MVS Ops can fill in the USER and PASSWORD fields from the submission screen. If you want to make use of this then code your JOB card like this:

        //NAME   JOB USER=%user%, PASSWORD=%pass%

The %user% and %pass% text will be replaced with the values set at job submission time.

If user authentication is turned on then the %user% will be filled from the mvsname provided, if any.

To Do and Known ~~Features~~ Bugs

The CSS used is very crude. In particular, listing output to screen does not always correctly support the iPhone/iPad.

Automate the installation and configuration.