# EPV for zLINUX Plus
# Installation and Customization

*IT Cost*
*Under Control*

Supporting
**EPV for zLINUX Plus V4**

**March 2013**

**All the mentioned trademarks belong to their respective companies.**

**EPV Technologies contact details:**

EPV Technologies
Viale Angelico, 54
00195 Roma
Tel.     06 86210880
Fax.    06 86387461
E-mail: epvtech@epvtech.com
WEB:   http://www.epvtech.com

# Contents

# About this manual

This manual is intended to help anyone wanting to install and customize EPV for zLINUX Plus V4.

# Changes

Technical changes or additions to the text are indicated by a vertical line to the left of the change.

# Terminology

A "view" is an EPV report presented in an HTML page.

# 1 Overview

**E**nterprise **P**erformance **V**ision for zLINUX Plus (EPV for zLINUX)  is a product designed to provide performance analysts, capacity and systems managers with a complete vision of their companies systems and workloads.

 EPV for zLINUX allows quick identification of anomalies, performance problems and abnormal resource consumptions; it is also an efficient tool for capacity planning.
The product uses auto discovery techniques that are completely transparent to the user, aggregates and correlates the useful metrics, producing valuable information ready to use.

EPV for zLINUX uses standard metrics available in the z/VM and LINUX environment.

The product architecture is modular and very flexible. It can be installed on most of  the hardware and software platforms on the market.

All information is presented on simple HTML static pages, which can be transferred onto any platform and accessed using a "browser".
All tables on the html pages can be  exported to a Microsoft Excel spreadsheet by a simple mouse click.

EPV for zLINUX is a component of the EPV product suite.

# 2  Product components

The product components are :

➤ **EPV for zLINUX Exceptions** provides a complete vision of the most important hardware and software threshold violations helping to immediately identify problems and anomalies. The default thresholds fit well for most installations.

➤ **EPV for zLINUX Configuration** provides a detailed vision of the hardware and software configuration, including virtual machine configuration; this information is the first mandatory step in order to control and manage your systems and in general to perform all Capacity Management activities. EPV for zLINUX correlates data coming from various sources and systems, producing a global vision of shared resources such as CECs, in a completely automated process. EPV for zLINUX audits configuration changes allowing an immediate identification of possible anomalies related to them.

➤ **EPV for zLINUX Resources** provides a complete vision of the "health" condition of the critical hardware resources, especially those shared amongst different z/VM systems as processors, memory, disks and channels. Starting from an integrated enterprise view you can navigate and analyze any saturation or performance problem. The time period and resource causing the bottleneck are highlighted using a red background so they can be very easily located.

➤ **EPV for zLINUX Workloads** provides a detailed vision of your workloads, allowing a simple and guided analysis of what is going on in complex environments through drill-down capabilities. Starting from a FUNCTION view, performance and resource consumptions can be analyzed drilling down virtual machines details. In addition the top running virtual machines are also provided for each hour.

➤ **EPV for zLINUX Trends** provides daily, weekly and monthly views of performance and resource consumptions at the enterprise, system and virtual machine levels. By use of these views it is possible to understand the growth of your workloads and its impact on your systems.

# 3  Architecture

EPV for zLINUX is a PERL application based on three tiers:

- ➢ Data Load Interface;
- ➢ Correlation and aggregation engine;
- ➢ HTML pages production engine.

EPV for zLINUX provides a free light version of the EPV zParser product to read Z/VM data and store it in a SQL database which will be the input for the Data Load Interface.

The data load interface is designed to optimize performance and resource consumption during the loading phase, avoiding duplication or data loss.
It is composed of PERL language exits that extract meaningful data and store it in a transition database. These exits represent a gateway from the environment to the product; there is an exit for each kind of data to load, and many others for general purposes.

The EPV for zLINUX detail database by default contains the last 3 days, and it is designed to avoid data loss, data duplication, and  to avoid the risk of producing incorrect statistics caused by non synchronized Z/VM data collections.

The correlation and aggregation engine loads a daily SQL database, including only the metrics used during the reporting phase, aggregated at hour, day, month level.
Configuration parameters set the number of days and months to retain in the database.

The HTML pages production engine can be customized in order to:

- ➢ Produce the HTML pages for one or more days
- ➢ Report daily, weekly and monthly trends for a desired period
- ➢ Set the number of days to use when calculating averages and percentile statistics
- ➢ Perform the statistical analysis to spot statistical values outside the normal distribution
- ➢ Choose which and how many days to include in the statistics

The EPV for zLINUX code is stored inside the EPVZLINUX_VXX folder under the PRODUCTS folder. The XX stays for the version number of the product.

The HTML pages are produced in an environment, and then transferred using FTP, or  another file transfer program, to the desired server, using the appropriate conversion table when necessary (EBCDIC to ASCII).

# 4 Preliminary settings and verifications

Before you proceed with the EPV for zLINUX installation you need to perform some preliminary actions and verifications.

## 4.1 Hardware and Software requirements

The following table summarizes EPV for zLINUX Hardware and Software minimum requirements:

| Component | Requirements |
|---|---|
| **Operating System** | any Microsoft Windows OS starting from NT<br>any Unix/Linux system (special considerations apply to AIX systems, please contact EPV technical support if you need more information about this) |
| **Hardware** | Any hardware platform supported by the previous operating systems. |
| **Processors** | 4 |
| **Memory** | 8 GB |
| **Disk Space** | The space needed for database tables and HTML pages depends on the number of monitored subsystems and the number of days retained in the performance DB. |
| **Software** | Supported DBMS:<br>    MySQL Server ver. 5.0 or higher.<br>    Microsoft MS SQL Server 2005 SP4 or higher.<br>. |

**Figure 1**

## 4.2 z/VM Input records

The following z/VM Monitor records are the mandatory input to EPV for zLINUX.

| DOMAIN | RECORD | DESCRIPTION |
|---|---|---|
| 0, SYSTEM DOMAIN | 1 | System Data (per processor) |
| 0, SYSTEM DOMAIN | 2 | Processor Data (per processor) |
| 0, SYSTEM DOMAIN | 3 | Real Storage Data (Global) |
| 0, SYSTEM DOMAIN | 5 | Expanded Storage Data (per processor) |
| 0, SYSTEM DOMAIN | 16 | Cpu Utilization Data in a Logical Partition |

| 0, SYSTEM DOMAIN | 17 | Physical CPU Utilization for Logical Partition Management |
|---|---|---|
| 0, SYSTEM DOMAIN | 20 | Expanded Channel Measurement Data (Per Channel) |
| 1, MONITOR  DOMAIN | 4 | System Configuration Data |
| 1, MONITOR  DOMAIN | 5 | Processor Configuration |
| 1, MONITOR  DOMAIN | 6 | Device Configuration Data |
| 1, MONITOR DOMAIN | 7 | Memory Configuration Data |
| 1, MONITOR DOMAIN | 8 | Paging Configuration Data |
| 1, MONITOR DOMAIN | 9 | Sample Profile |
| 1, MONITOR DOMAIN | 17 | Expanded Storage Data |
| 3, STORAGE DOMAIN | 4 | Auxiliary Storage Data |
| 4, USER DOMAIN | 3 | User Activity Data |
| 4, USER DOMAIN | 4 | User Interaction Data |
| 6, I/O DOMAIN | 3 | Device Activity |

**Figure 2**

## 4.3   Additional input data

You can collect information about processes activity inside zLinux guests. Please refer to chapter 8.5 for details.

# 5  Installation

**Warning**: if you performed the EXPRESS customization as described in the "EPV Plus Products for z/OS Installation and EXPRESS Customization" the EPV for zLINUX Plus product is already installed.
No other action is required.

To install EPV for zLINUX you need to perform the following steps:

1. DBMS installation;
2. Prepare products and password folders in Windows;
3. Prepare products and password folders in Unix/Linux.

## 5.1  DBMS Installation

DBMS installation should have already been performed when installing the EPV zParser product which is a prerequisite to EPV for zLINUX. Please refer to the EPV zParser documentation.

## 5.2  Preparing products and password folders in Windows

Products and password folders should have already been prepared when installing the EPV zParser product which is a prerequisite to EPV for zLINUX. Please refer to the EPV zParser documentation.

If for any reason you need to update the EPV for zLINUX product, in Windows systems you have to copy the supplied /PRODUCTS/EPVZLINUX_VXX folder (where XX stays for the version number of the product) from the EPV Installation CD to the PRODUCTS folder in EPV zParser installation folder.

From here on the *"$EPVPATH"* variable should be substituted with the path where the installation software was copied.

**WARNING:** those folders should not be copied to the disk drive root folder, so we recommend to create a folder in the root (e.g. $EPVPATH=/EPVROOT).

If you need to update the license key please copy the LICENSE_EPVZLINUX.EPV file in the $EPVPATH/PASSWORD folder.

## 5.3   Preparing products and password folders in Unix/Linux

Products and password folders should have already been prepared when installing the EPV zParser product which is a prerequisite to EPV for zLINUX. Please refer to the EPV zParser documentation.

If for any reason you need to update the EPV for zLINUX product, in Unix/Linux systems you have to copy the supplied /PRODUCTS/EPVZLINUX_VXX folder (where XX stays for the version number of the product) from the EPV Installation CD to the PRODUCTS folder in EPV zParser installation folder.

From here on the *"$EPVPATH"* variable should be substituted with the path where the installation software was copied.

If you need to update the license key please copy the LICENSE_EPVZOS.EPV file in the *$EPVPATH*/PASSWORD folder.

*As discussed in Chapter 7 of the EPV zParser Installation and Customization manual, you can create a user profile in Windows by using the EPV Customization GUI and then move the user profile to a Unix/Linux system where the EPV products have to run.*
*Only in that case in order to run the EPV Customization GUI you have also to copy the supplied /PASSWORD, /PRODUCTS,  /SETUP, /TOOLS, /DOCUMENTS and /USERPROFILE folders, and all the included subfolders, from the EPV Installation CD to a folder in a Windows system (e.g. /EPVROOT/).*
*All the provided products licenses have also to be copied in the PASSWORD folder. These licenses will only be used to run the EPV Customization GUI.*

## 5.4   HTML publishing folders

On the system that will host the HTML pages you have to verify that the following folders  exist under *$EPVPATH*[1]:
**JAVA**        folder containing JAVA code, scripts and  sheets for the user interface;
**IMG**          folder containing images used by EPV (EPV logo, Microsoft Excel icon, etc.);
**LNXDOC**    folder containing the pages of the help system;
**LNXHTML**  folder containing all the HTML pages produced (empty at installation)

The following HTML pages have also to exist under *$EPVPATH*: **START.HTML**, **HOME.HTML**, **ABOUT.HTML**, **CONTRIBUTORS.HTML, EPVLTREE.HTML**, **LRESEARCH.HTML**, **LRESI.HTML** and **LRESO.HTML**.

If any of these folder and files doesn't not exist you must copy what is missing from the ./PRODUCTS/EPVUI folder of the EPV Installation CD to *$EPVPATH*.

---

[1] It depends on the type of installation you did (express or advanced) and on the installed products.

# 6 Manual Customization (mandatory)

If for any reason you don't want to use the EPV Customization GUI you can perform a manual customization following these steps.

## 6.1 Customizing the DBs

All the procedures you need are placed under the *$EPVPATH* /TOOLS/ZLINUX folder and are classified by DB engine and platform installation. A unique procedure (RUNALL.BAT for Windows systems and RUNALL.sh for Unix/Linux systems) is provided which calls all the others in the right sequence.

To run the procedure, open a command prompt or shell and run:

- for Windows systems (MS SQL Server): RUNALL.BAT *youruser yourpassword instance,*
  where *youruser* and *yourpassword* are those defined in your DBMS and *instance* is the MS SQL Server instance;
- for Windows systems (MySQL): RUNALL.BAT *youruser yourpassword,*
  where *youruser* and *yourpassword* are those defined in your DBMS;
- for Unix and Linux systems (MySQL): ./RUNALL.sh *youruser yourpassword.*
  where *youruser* and *yourpassword* are those defined in your DBMS.

## 6.2 Customizing the SETTINGS parameters

In order to create the HTML reports under a desired location please open the SETTINGS.PL file placed under the *$EPVPATH*/USERPROFILE/*$Profilename*/COMMON folder (where *$Profilename* is the name of your user profile set at EPV zParser installation) and customize the $HTMDIR variable specifying the preferred HTML path.

## 6.3 Verify the MIPS table

The EPV for zLINUX version 4 folder contains more different MIPS tables than in the past. They are named MIPSLTxx.PL, MIPSATxx.PL, MIPSPTxx.PL, MIPSARxx.PL, MIPSLRxx.PL, MIPSHRxx.PL and MIPSPRxx.PL, where xx is the z/OS level (e.g xx=19 means z/OS version 1.9) and they are based on IBM LSPR benchmarks.

The **MIPSLTxx** tables contain the low I/O rate (LoIO) estimated MIPS (for the IBM hardware models 2064 and higher only).
The **MIPSATxx** tables contain the estimated average MIPS (for all the IBM machines).
The **MIPSPTxx** tables contain the Performance Capacity Index (PCI) estimated MIPS (it is currently available only for z/OS 1.9 benchmarks).

The **MIPSARxx** tables contain the average Relative Nest Intensity  (RNI) estimated MIPS (e.g. xx=13 for z/OS 1.13).
The **MIPSLRxx** tables contain the low Relative Nest Intensity  (RNI) estimated MIPS (e.g. xx=13 for z/OS 1.13).
The **MIPSHRxx** tables contain the high Relative Nest Intensity  (RNI) estimated MIPS (e.g. xx=13 for z/OS 1.13).
The **MIPSPRxx** tables contain the Performance Capacity Index (PCI) estimated MIPS (e.g. xx=13 for z/OS 1.13).

EPV for zLINUX uses the TABMIPS.PL file saved under the $EPVPATH/USERPROFILE*/$Profilename*/EPVZLINUX folder  (where *$Profilename* is the name of your user profile set at EPV zParser installation); by default it contains the values in the MIPSHR13 table. If you want to use other values you can copy any of the above tables in TABMIPS[2].

Customers **CAN MODIFY** TABMIPS table values but they **SHOULD NOT** modify the MIPSARxx, MIPSLRxx, MIPSHRxx and MIPSPRxx tables.

By default the IFL pool capacity is taken directly from the TABMIPS table without considering the MP effect added by other processors (GCP, zAAP and zIIP).

However, MIPS tables and EPV estimates have to be considered as "average" values because they do not take into consideration the number of active LPARs and their configuration. We strongly advice customers to use the IBM zPCR tool in order to get good estimates.
This is the reason why the exit UEXMIPS  is also provided in the *$EPVPATH*/USERPROFILE/*$Profilename*/EPVZLINUX/USEREXIT  folder  allowing customers to set their trusted MIPS values (possibly estimated using zPCR) for IFL pools capacity.

To avoid the risk of using obsolete MIPS values, the EPV process will terminate if it finds a new machine model whose MIPS values set in the exits have not been updated.

---

[2] Verify in the chosen table if all the machines in your site are available. EPV Technologies will provide an updated version of the tables if you verify that some of your machines are missing.

Installation and Customization

# 7 Scheduling

Scheduling the EPV for zLINUX provided procedures (manually or automatically) allows you to process data and produce the HTML reports daily.
All the .BAT procedures mentioned here are available as .sh to allow EPV for zLINUX to run on Unix/Linux systems.

## 7.1 Collecting data once a day

When your environment is set to collect data once a day the whole EPV process is run by the ALLPHASES.BAT procedure, created during EPV zParser installation and scheduled daily, which is located in the *$EPVPATH*/USERPROFILE*/$Profilename*/COMMON/PROCS folder.
Inside that procedure you should find (or add if missing) a CALL to the NIGHTBATCH.BAT procedure located under the *$EPVPATH*/USERPROFILE*/$Profilename*/EPVZPARSER/AGENT/PROCS folder, and inside this procedure you should find a CALL to the NIGHTBATCH_ZLINUX.BAT.

## 7.2 Collecting data in continuous mode using EPV agents

When your environment is set to collect data in a continuous mode using the EPV agents, the daily consolidation process is run by the POSTZPARSER.BAT procedure, created during EPV zParser installation, which is located in the *$EPVPATH*/USERPROFILE/*$Profilename*/COMMON/PROCS folder.

Inside that procedure you should find (or add if missing) a CALL to the NIGHTBATCH_ZLINUX.BAT procedure which is located under the *$EPVPATH*/USERPROFILE*/$Profilename*/EPVZLINUX/PROCS folder.

# 8 Manual Customization (optional)

In the following some optional steps, to fit specific user needs, are described.

## 8.1 Customizing the CONFIG parameters

The default values are valid for most sites. However you can customize parameters and thresholds settings in the *$EPVPATH*/USERPROFILE*/$Profilename*/EPVZLINUX/CONFIG.PL file (where *$Profilename* is the name of your user profile set at EPV zParser installation) as desired.

Attachment A provides a short description of all the EPV parameters and thresholds, their default values and their meaning.

## 8.2 Thresholds and Exceptions customization

EPV for zLINUX provides a set of base thresholds to control both resource utilization and application performance. Each base threshold is a single value controlling all the occurrences of a specific metric. Base thresholds can be customized by modifying the default values provided in the CONFIG.PL file of the *path/*USERPROFILE*/$Profilename*/EPVZLINUX folder.
While the threshold value in the CONFIG.PL file is generally valid, there are situations where a different threshold value is needed for a specific system or for a particular hour of the day.
This is the reason why advanced thresholds have been introduced. The current implementation is based on a specific exit associated to each threshold. By changing the provided sample exit, customers can set as many different threshold values, for each controlled metric, as needed.
The name of all these user exits follows this naming convention: **T** *exception name*.
Each user exit contains an example of CASE statement with all the criteria variables which can be used to modify the threshold value.

Attachment B contains the default values for each base threshold and the name of each advanced threshold file.

Every time a base or advanced threshold is violated an exception is generated.
All generated exceptions are reported by default in the HTML group.

At the moment two different exception groups are defined: HTML and SYSTEM.
For each defined group except for the HTML group a text file is produced, (groupname.txt), containing a list of the exceptions associated with that specific group.

Through the **AGROUPS.PL** file, located in the *$EPVPATH*/USERPROFILE*/$Profilename*/EPVZLINUX/USEREXIT folder you can define as many groups as you need and assign exceptions (using an ALERT code) to groups.

Customizing the **AFILTERS.PL** file in the *$$$$EPVPATH*/USERPROFILE*/$Profilename*/EPVZLINUX/USEREXIT folder you can exclude alerts or hours you do not want to consider: for example, you could consider warnings coming from the TEST environment not worth to generate exceptions.

## 8.3 Exclusion of incorrect values from statistical analysis

EPV for zLINUX provides user exits to exclude values caused by loops or other anomalies which could partially invalidate trend statistical analysis. The name of these user exits follows this naming convention: **UESxxxxx.PL** where **xxxxx** are the last five characters of each statistical exception name.
Each file contains an example CASE statement with all the criteria variables which can be used to eliminate the incorrect value.
Attachment D contains the name of all the statistical exceptions controlled by EPV for zLINUX and the name of each user exit with a short description.

## 8.4 Function characterization

The **UEXFUNC.PL** file located in the *$EPVPATH*/USERPROFILE*/$Profilename*/EPVZLINUX/USEREXIT folder contains the identification logic to automatically characterize the function.
You can modify this logic to add your customizations setting the FUNCTION variable which must not be longer than 20 characters and must not contain blanks.
By default the FUNCTION variable is set to user account value.
The settings take place during the EPV load.

## 8.5 Collect process activity inside zLinux guest

To collect process activity information from inside zLinux guests you have to:
- Schedule a shell script running the ps command (see an example in Attachment C) on each zLinux guest you want to control;
- Make the produced output files accessible to EPV;
- Set the DETAPROC parameter to Y; It is located located in the CONFIG.PL member which is located in the *$EPVPATH*/USERPROFILE*/$Profilename*/EPVZLINUX folder;
- Modify the LNXDATA parameter in the CONFIG.PL member to point to the ps output files;
- Associate a VM system and user name to each zLinux system by customizing the UEXPROC member which is located in the *$EPVPATH*/USERPROFILE*/$Profilename*/EPVZLINUX/USEREXIT folder.

## 8.6 Exclusion of logical volumes from response time analysis

It is possible to exclude some logical volume from the response time analysis (i.e. work areas, public devices, etc.).
In order to accomplish this task you need to define the disks you want to exclude in the **UEXDISKS.PL** file located in the *$EPVPATH/*USERPROFILE*/$Profilename/*EPVZLINUX/USEREXIT folder.

## 8.7 Setting SHIFTS

All Trend views provide HTML tables for different shifts. The file named **SHIFT.PL** located in the *$EPVPATH/*USERPROFILE*/$Profilename/*EPVZLINUX/USEREXIT folder contains our default shift values which should be modified with your standard company values. When migrating to a new version, it is very important to control your shift definitions in the **SHIFT.PL** file before loading new data. The easiest way is by copying the **SHIFT.PL** file from your old folder to the new one.
EPV for zLINUX summarizes the shifts when producing the HTML pages.

## 8.8 Managing configuration changes

It is possible to enable or disable the metrics controlled by EPV in order to report about configuration changes. You can customize the **CHANGES** member located in the *$EPVPATH/*USERPROFILE*/$Profilename/*EPVZLINUX/USEREXIT folder by modifying the CHANGE column.

## 8.9 Publishing on the IBM HTTP Server on zSeries

To publish the EPV for zLINUX HTML pages on the IBM HTTP Server on zSeries, the following steps are needed:
   a) Verify that all files and directories are in uppercase except for the .class suffix.
   b) Transfer all the files in the IMG directory in binary mode.
   c) Transfer the following files in the JAVA directory in binary mode:

   *.JS
   *.class
   *.CSS

Transfer all the other files in ascii mode using the following quote command[3]:
       **quote site sbdataconn=(IBM-1047,ISO8859-1)**

---

[3] IBM-1047 has to be eventually substituted with the DefaultFsCp value set in httpd.conf (if different from IBM-1047).

IMPORTANT NOTE:
Beyond these general rules the correct FTP mode (binary or ascii) depends on how customers set their web server. Depending on the **addtype** statement in **httpd.conf** the different file types have to be transferred in binary or ascii mode.

Normally .JS files are used without converting to ebcdic (the conversion needs to be done again when transferring the .JS to the client; it takes some time and is not good for performance).
However if the .JS files are defined as ebcdic they need to be transferred in ascii.

The .CSS should in any case be defined using an **8bit addtype** and transferred in binary mode to work properly.

## 8.10 Exporting reports in Microsoft Excel

It is possible to load an EPV for zLINUX view or an entire EPV for zLINUX page to a Microsoft Excel spreadsheet.
This functionality is based on ActiveX. In order to use it you need to set the security of your browser at a low level, or better insert the EPV for zLINUX website in the list of the trusted sites.

The advanced export option permits users to save the view or entire page at a specified location and perform the execution of an automatic macro. The following  steps need to be performed :

a) **Set an environment variable:** on the PC where you perform the advanced export you need to set an environment variable: the variable name **must be** EPVXCL and it must contain a valid path pointing to a directory where you want to store your Excel files. The value set in the variable must end with a backslash.

b) **Excel file naming convention:** each EPV for z/OS view has an associated standard name; you can find this name looking at the folder name of the opened work sheet. In order to use the advanced export functionality you need to save the sheet in the path defined before, with the same name appearing in the folder.

c) **Main macro name:** in order to automatically activate a macro you must name the macro EPVEXEC. The advanced export function will search for an Excel file in the defined path with the name of the chosen view, if it exists, it will load the data and execute the EPVEXEC macro (if defined).

If you cannot modify the security of your browser, EPV for zLINUX allows you to use two different export functionalities.
If you right click on a table cell you will get the standard browser menu and the users will be able to use the Export to Excel function available in latest IE versions[4]. In this case the view exported to a Microsoft Excel spreadsheet appears without the standard EPV format.

---

[4] Similar function can be obtained using other browsers through add-on and plugins.

---

Another way allows you to export the entire page without using ActiveX. In this case the user should follow these instructions:

a) **copy and open an Excel template:** on the installation CD you can find the EPV_EXPORT Excel file located inside the HTM\TOOLS directory. Copy this file to your PC and open it.

b) **get the html page URI:** clicking on the right mouse button, you should select the GET PAGE URI item; this function opens a new HTML page that show the URI you want to export.

c) **export page:** you should copy the HTML page URI and paste that in the input box that appears in the above Excel template

## 8.11 Customizing the user interface

EPV for zLINUX pages are produced following HTML standards. It is possible to customize the appearance of pages (font, colours, etc.); for a detailed description of the utilized styles see the "EPV V10 - User Interface".

## 8.12 Customizing the SORT feature

EPV for zLINUX allows to sort each report inside the HTML pages; to avoid performance problem when you activates this feature, EPV provides two variables inside the **EPV_CONFIG.TXT** file in the JAVA directory (see 5.1.2). The **SORTMIN** and **SORTMAX** variables allows to sort only the tables with a number of records in that range. Moreover to sort correctly the numeric data inside each table, you should verify the **FMT** variable located in **EPV_CONFIG.TXT** file. The value of this variable Shoul be equal to the one assigned to the **FMT** parameter in the **CONFIG.PL** member in the *$EPVPATH/*USERPROFILE*/$Profilename/*EPVZLINUX folder.

## 8.13 Automatic deletion of old pages in Windows environment

The deletion of old EPV HTML pages can be performed by a program named RemoveOldFile.exe that runs in the WINDOWS environments[5]. This program must be scheduled daily on the server hosting the EPV HTML pages; it will automatically delete directories and pages older than a user defined number of days.
This program can be found on the installation CD under $EPVPATH/TOOLS and must be customized to fit your needs.
A full explanation of the parameters that must be modified is provided in the program comments.

## 8.14 Customizing the Recovery procedure

By default EPV for zLINUX loads the detail data in the LDETA database for the last 3 days, maintains the hourly summarized data for the last 60 days and uses this data to create the daily reports for the last two days. In cases where you want to automatically recover data older than 60 days and create the daily html pages for them without the need to modify manually the

---

[5] If the HTML pages are published in z/OS (USS), UNIX or Linux, a user function based on the rm command should be implemented.

CONFIG.PL parameters ($LDETDAYS, $DETAIL, $RPTEPVS, $RPTEPVE), you can customize the RECOVER.PL member located under the *$EPVPATH*/USERPROFILE/$Profilename/COMMON folder. The recovery procedure dynamically changes the above CONFIG.PL parameters internally in the program to permit the loading and creation of the html pages without modifying the original CONFIG.PL parameters.

The following parameters need to be customized:

- $RECOVERY = To activate the recovery procedure set to YES (default is NO);
- $MINDURTM = Minimum duration in seconds accepted with the same SYSTEM and DATE by the recovery procedure (default is 3600);
- $MAXRPTDAYS = Maximum number of HTML reporting days (default is 2);
- $STGDB = Parser Staging Database – used only when collecting data in continuous mode (default is BLANK).

**WARNING:** the recovery procedure doesn't change any of the trend configuration parameters. In order to maintain the recovered data into the HTML trend reports and in the EPV trend database, you need to verify that the trend parameters in the CONFIG.PL member cover the period you want to maintain.

.

# 9 Using the EPV Customization GUI

The EPV Customization GUI has been designed to run on a Windows system, however you can create a profile which can be used to run the EPV products on Unix or Linux.

To start working with the EPV Customization GUI, you must enter the SETUP folder and run the EpvInst.exe program.

## 9.1 ADVANCED customization

You will get the windows in Figure 3 where you have the possibility to choose the customization mode.

To proceed with the ADVANCED customization you have to select ADVANCED and press CONTINUE (for the EXPRESS customization please refer to the EPV Plus Products for z/OS Installation and EXPRESS Customization manual).
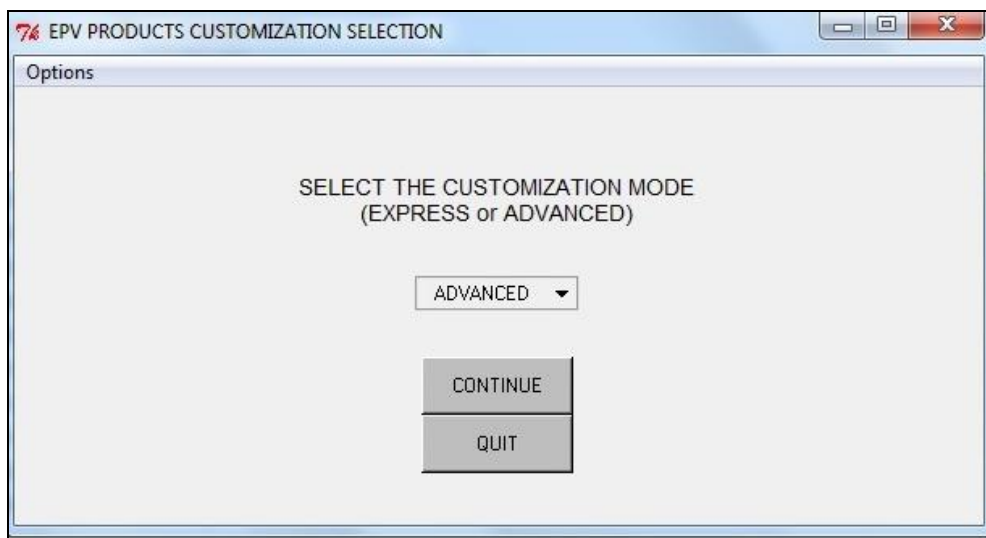


**Figure 1**

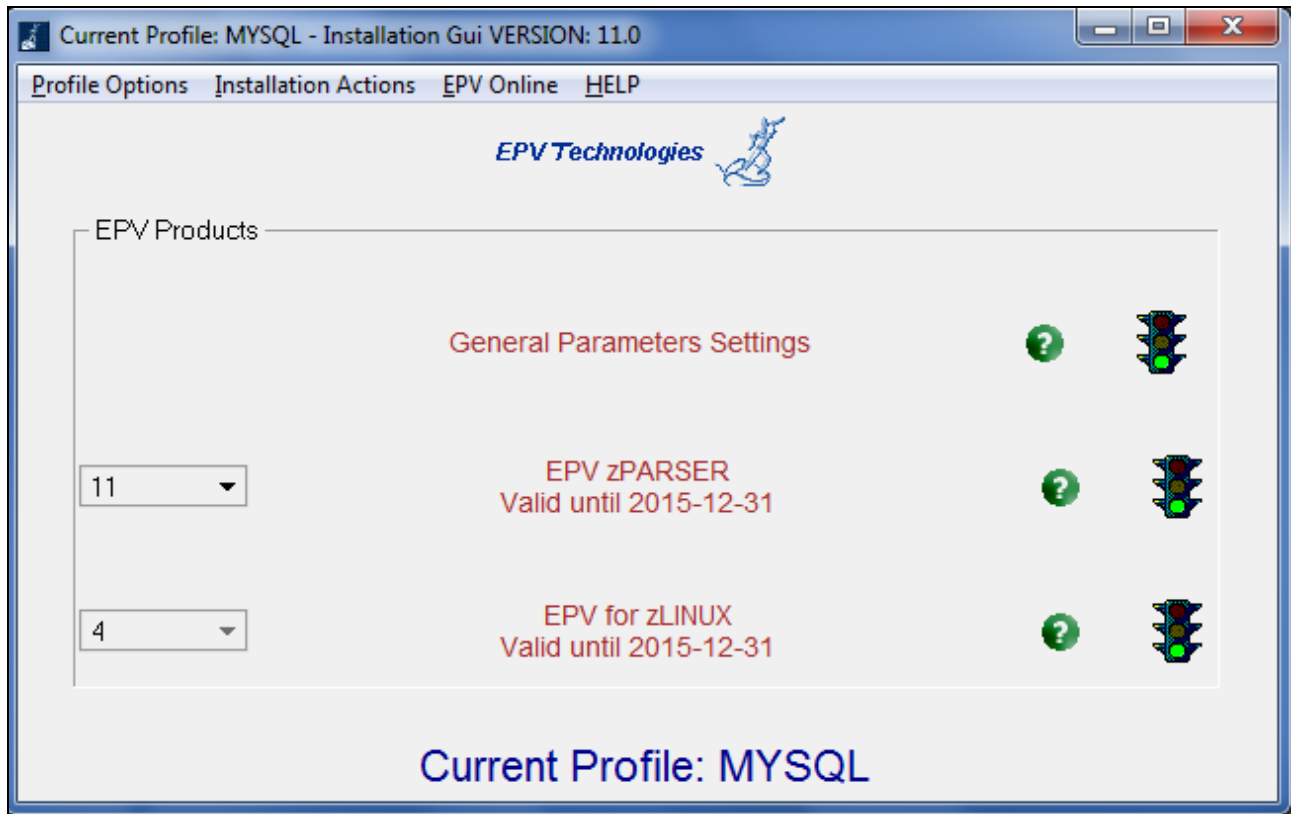In the next window you are asked to select a profile.

**Figure 2**

Click OK and choose an existing profile (normally created during the EPV zParser installation).

## 9.2 Customizing the DBs

After selecting the user profile you have to run the Create DataBases and Install Stored Procs options (in this order) provided under the Installation actions menu on the top of the window, as shown in the image below, to create the databases needed by the EPV for zLINUX product:

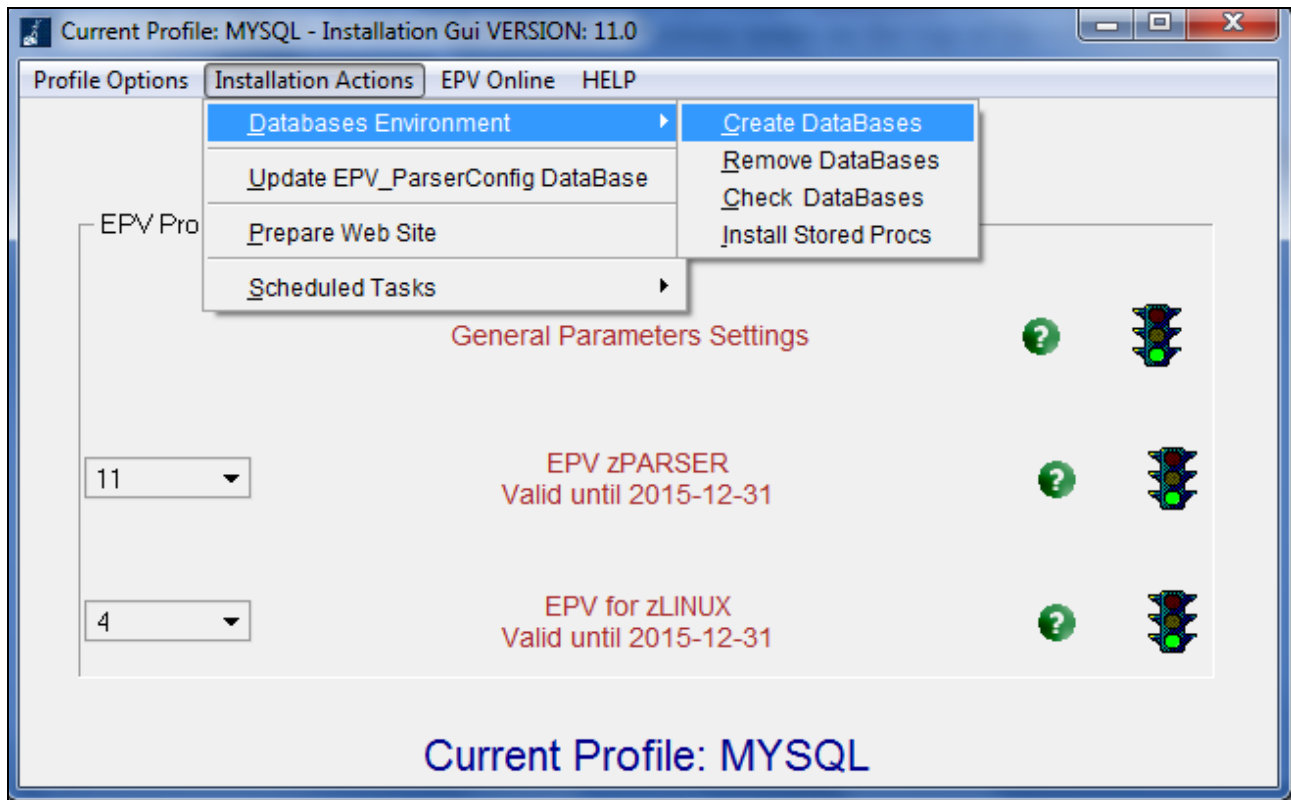

**Figure 5**

**Warning:** this window does not appear when installing EPV Plus Products in Unix/Linux.
In this case the DBs have to be allocated after moving the profile in the Unix/Linux system by performing the steps described in Chapter 9.4.

## 9.3   Customizing the product's parameters

If you have a valid license for the EPV for zLINUX product you first need to select the correct version from the combo box located at the left side of the main panel and then choose the EPV for z/LINUX entry.
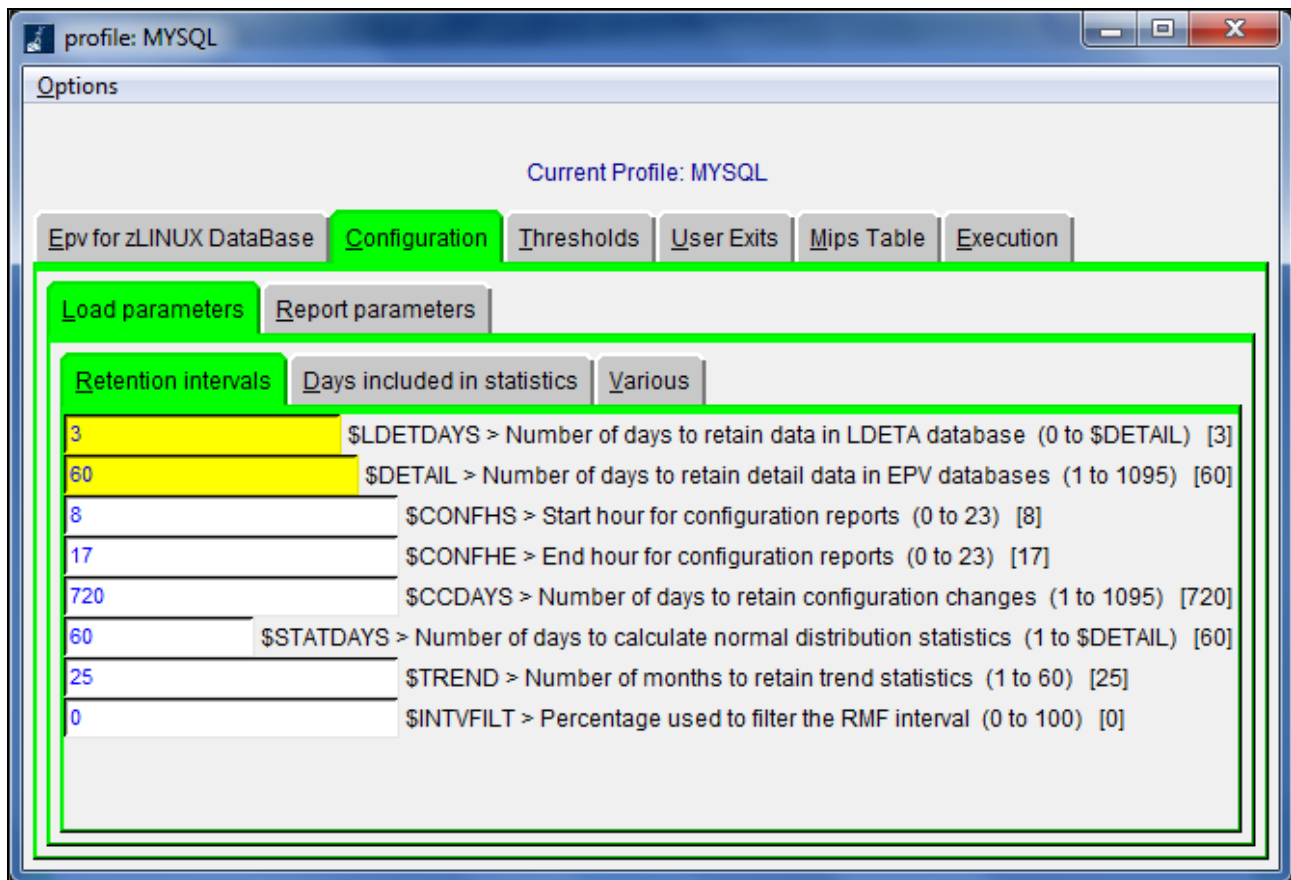


**Figure 6**

Through this panel you can customize all the parameters needed by EPV for zLINUX, change the default threshold settings, modify the user exits, decide what kind of  benchmark to use to evaluate the MIPS of your environment and so on.
For a detailed description of all this options please refer to Attachment A, B, C and D at the end of this manual or to the detailed description done in the previous chapters for the manual installation.

Finally save the customizations you have done by clicking on the 'Options' menu and choosing the 'Save and Return' option; it will close the panel and bring you again to the main window.

## 9.4 Additional customization steps for Unix and Linux systems

If you need to move a new or modified profile to a Unix or Linux system you have to perform the following additional customization steps:

a) export a variable named *$EPVPATH* that contains your EPV path installation root folder; this can be accomplished by issuing the following command:

export EPVPATH=/home/epv/EPVROOT/

or by putting it in the EPV user automatic logon script (e.g. bashrc or /etc/profile).

b) change the permission of all folders and files, inside it, to at least 755;

c) remove all the CR (Carriage Return) inside all the .sh files in the profile; you can do that by using the standard DOS2UNIX utility; we suggest to run the following command from inside your *$profilename:* **find . -type f -name '*.sh' -exec dos2unix '{}' \;**

d) if not already done, you have to allocate the z/OS DBs by executing the RUNALL.sh script, located in *$EPVPATH*/TOOLS/EPVZOS/MYSQL_PROCS/UNIX, providing MySQL user name and password (e.g. ./RUNALL.sh youruser yourpassword).

## 9.5 EPV online (only when running in Windows systems)

The EPV online menu permits you to load data, create reports and view all the log files interactively.

> **Warning**: be aware that to run the EPV Plus Products on a regularly basis you have to put the appropriate procedures in your daily scheduling (see Chapter 7).

You should run the phases in the same order in which they appear in the following menu:
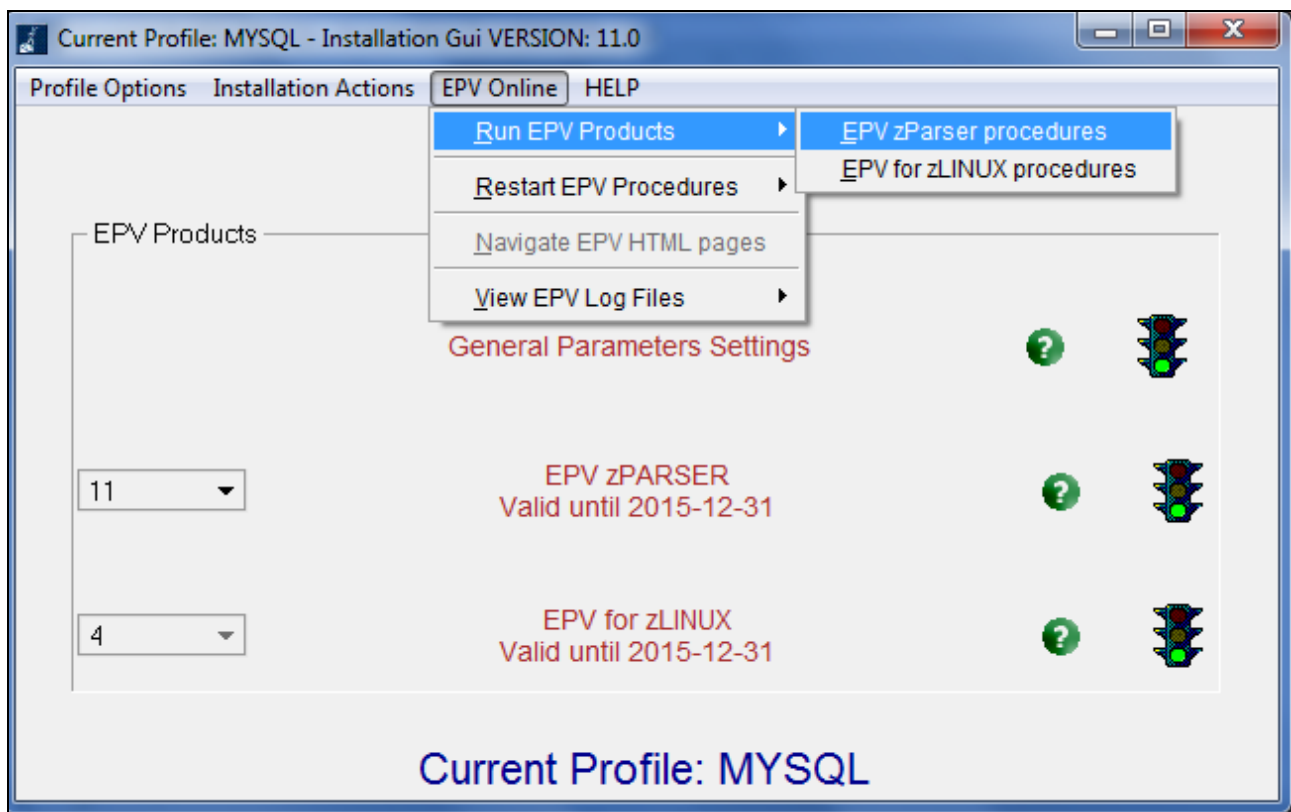


**Figure 7**

For each step a dedicated window will be opened. From this window it is possible to check the return code of each step performed and view the relative detail log created by the EPV Plus Products.

This feature is very useful to perform spot loading of old data or to produce html pages for old dates without changing the user profile settings.

# 10 Using the EPV Customization GUI

For any technical problem with or question about EPV for zLINUX write an email to:

epv.support@epvtech.com

For any other issue about EPV for zLINUX please write an email to:

epv.info@epvtech.com

# 11 Customer support

For any technical problem with or question about EPV for zLINUX write an email to:

epv.support@epvtech.com

For any other issue about EPV for zLINUX please write an email to:

epv.info@epvtech.com

# Attachment A – Config parameters

Default values are included in square brackets.

| | |
|---|---|
| **$LDETA** | Epv for zLINUX DataBase $MDETA > Detail DB name |
| **$LCONF** | Epv for zLINUX DataBase $MCONF > Configuration DB name |
| **$LRESA** | Epv for zLINUX DataBase $MRESA > Resource analysis DB name |
| **$LWKLA** | Epv for zLINUX DataBase $MWKLA > Workload analysis DB name |
| **$LSTAT** | Epv for zLINUX DataBase $MSTAT > Statistic DB name |
| **$LTRND** | Epv for zLINUX DataBase $MTRND > Trend DB name |
| **$LPROC** | Epv for zLINUX DataBase $USRZOS > Process DB name |
| **$LWORK1** | Epv for zLINUX DataBase $ZWORK1 > First work DB name |
| **$LWORK2** | Epv for zLINUX DataBase $ZWORK2 > Second work DB name |
| **$LWORK3** | Epv for zLINUX DataBase $ZWORK3 > Third work DB name |
| **$LWORK4** | Epv for zLINUX DataBase $ZWORK4 > Fourth work DB name |
| **$LWORK5** | Epv for zLINUX DataBase $ZWORK5 > Fifth work DB name |
| **$LWORK6** | Epv for zLINUX DataBase $ZWORK6 > Sixth work DB name |
| **$LDETDAYS** | Retention intervals $LDETDAYS > Number of days to retain data in LDETA database (0 to $DETAIL) [3] |
| **$DETAIL** | Retention intervals $DETAIL > Number of days to retain detail data in EPV databases (1 to 1095) [60] |
| **$CONFHS** | Retention intervals $CONFHS > Start hour for configuration reports (0 to 23) [8] |
| **$CONFHE** | Retention intervals $CONFHE > End hour for configuration reports (0 to 23) [17] |
| **$CCDAYS** | Retention intervals $CCDAYS > Number of days to retain configuration changes (1 to 1095) [720] |
| **$STATDAYS** | Retention intervals $STATDAYS > Number of days to calculate normal distribution statistics (1 to $DETAIL) [60] |
| **$TREND** | Retention intervals $TREND > Number of months to retain trend statistics (1 to 60) [25] |
| **$INTVFILT** | Retention intervals $INTVFILT > Percentage used to filter the RMF interval (0 to 100) [0] |
| **$MONDAY** | Days included in statistics $MONDAY > Include Monday in statistical analysis (YES or NO) [YES] |

| $TUEDAY | Days included in statistics $TUEDAY > Include Tuesday in statistical analysis  (YES or NO)  [YES] |
|---|---|
| $WEDDAY | Days included in statistics $WEDDAY > Include Wednesday in statistical analysis  (YES or NO)  [YES] |
| $THUDAY | Days included in statistics $THUDAY > Include Thursday in statistical analysis  (YES or NO)  [YES] |
| $FRIDAY | Days included in statistics $FRIDAY > Include Friday in statistical analysis  (YES or NO)  [YES] |
| $SATDAY | Days included in statistics $SATDAY > Include Saturday in statistical analysis  (YES or NO)   [NO] |
| $SUNDAY | Days included in statistics $SUNDAY > Include Sunday in statistical analysis  (YES or NO)   [NO] |
| $HOLIDAY | Days included in statistics $HOLIDAY > Include Holiday in statistical analysis  (YES or NO)   [NO] |
| $LOADSFM | Various $LOADSMF > Load type settings  (DEFAULT or FORCE)  [DEFAULT] |
| $CHECKMIP | Various $CHECKMIP > Control MIPS values when CEC is updated  (YES or NO)  [Y] |
| $LOADSMF | Various $LOADSMF > Load type settings  (DEFAULT or FORCE)  [DEFAULT] |
| $STAT | Various $STAT > Activate normal distribution  (YES or NO)  [YES] |
| $STD | Various $STD > Number of std to test for normal distribution(only when STAT is Y)  (>0)  [3] |
| $DETAPROC | Various $DETAPROC > Load process data  (YES or NO)  [NO] |
| $LNXDATA | Various $LNXDATA > Path of the linux process data directory |
| $RPTEPVS | Time intervals $RPTEPVS > Start reporting date  (DEFAULT or yyyy-mm-dd)  [DEFAULT] |
| $RPTEPVE | Time intervals $RPTEPVE > End reporting date  (DEFAULT or yyyy-mm-dd)  [DEFAULT] |
| $CCHTM | Time intervals $CCHTM > Number of days to report configuration changes  (1 to 365)  [360] |
| $CCRPT | Time intervals $CCRPT > Default to report the configuration changes of last cchtm days or date to start from specific date  (DEFAULT or yyyy-mm-dd)  [DEFAULT] |
| $NAVIGATE | Time intervals $NAVIGATE > Number of days to navigate in main HTML page  (1 to $DETAIL)  [10] |
| $TRENDDAY | Time intervals $TRENDDAY > Number of days in day trend vision  (1 to $DETAIL)  [60] |
| $TRENDMON | Time intervals $TRENDMON > Number of months in month trend vision  (1 to $TREND)  [25] |

| $TOPDISK | Top statistics $TOPDISK > Number of top DISKS for each pcuid in resource vision  (1 to 50)  [10] |
|---|---|
| $TOPUSER | Top statistics $TOPUSER > Number of top VM user for each hour in workload vision  (1 to 50)  [20]  [10] |
| $RDAYMON | Days included in trend day $RDAYMON > Include Monday in trend day reports  (YES or NO)  [YES] |
| $RDAYTUE | Days included in trend day $RDAYTUE > Include Tuesday in trend day reports  (YES or NO)  [YES] |
| $RDAYWED | Days included in trend day $RDAYWED > Include Wednesday in trend day reports  (YES or NO)  [YES] |
| $RDAYTHU | Days included in trend day $RDAYTHU > Include Thursday in trend day reports  (YES or NO)  [YES] |
| $RDAYFRI | Days included in trend day $RDAYFRI > Include Friday in trend day reports  (YES or NO)  [YES] |
| $RDAYSAT | Days included in trend day $RDAYSAT > Include Saturday in trend day reports  (YES or NO)  [NO] |
| $RDAYSUN | Days included in trend day $RDAYSUN > Include Sunday in trend day reports  (YES or NO)  [NO] |
| $RDAYHOL | Days included in trend day $RDAYHOL > Include Holiday in trend day reports  (YES or NO)  [NO] |
| $RMONMON | Days included in trend month $RMONMON > Include Monday in trend month reports  (YES or NO)  [YES] |
| $RMONTUE | Days included in trend month $RMONTUE > Include Tuesday in trend month reports  (YES or NO)  [YES] |
| $RMONWED | Days included in trend month $RMONWED > Include Wednesday in trend month reports  (YES or NO)  [YES] |
| $RMONTHU | Days included in trend month $RMONTHU > Include Thursday in trend month reports  (YES or NO)  [YES] |
| $RMONFRI | Days included in trend month $RMONFRI > Include Friday in trend month reports  (YES or NO)  [YES] |
| $RMONSAT | Days included in trend month $RMONSAT > Include Saturday in trend month reports  (YES or NO)  [YES] |
| $RMONSUN | Days included in trend month $RMONSUN > Include Sunday in trend month reports  (YES or NO)  [YES] |
| $RMONHOL | Days included in trend month $RMONHOL > Include Holiday in trend month reports  (YES or NO)  [YES] |
| $DBDEC | Various $DBDEC > Decimal point to store in EPV databases  (1 to 3)  [3] |
| $DEBUG | Various $DEBUG > Debug detail level  (0 to 3)  [2] |
| $FMT | Various $FMT > number format (E/S/U) Europe/Switzerland/Usa  (E, S or U)  [E] |

| $DBCHECK | Various $DBCHECK > Create page statistics for EPV databases  (YES or NO)   [NO] |
|---|---|
| $CECAVA | General Cpu thresholds $CECAVA > Thershold for %CEC available  (0 to 100)  [5] |
| $IFLBUSY | General Cpu thresholds $IFLBUSY > Threshold for %IFLBUSY used  (0 to 100)  [95] |
| $UVMCPUS | General Cpu thresholds $UVMCPUS > Thresholds for Total CPU time used for each VMACHINE  (>0) [1800] |
| $Z/VMRESP | General Cpu thresholds $Z/VMRESP > Thresholds for Average Response Time for each SYSTEM(ms) (>0)  [10] |
| $TVRATIO | General Cpu thresholds $TVRATIO > Thresholds for CPU ratio of Z/VM usage and VMACHINE usage (>0)  [1.5] |
| $Z/VMPGIN | Disk thresholds $Z/VMPGIN > Thresholds for SYSTEM Page-In rate  (0 to 100)  [100] |
| $UVMPGIN | Disk thresholds $UVMPGIN > Thresholds for VMACHINE Page-In rate  (0 to 100)   [10] |
| $Z/VMPGSP | Disk thresholds $Z/VMPGSP > Thresholds for SYSTEM page utilization  (0 to 100)   [50] |
| $DSKRESP | Disk thresholds $DSKRESP > Thresholds for Average Response Time for each disk(ms)  (>0)   [30] |
| $DSKIOSQ | Disk thresholds $DSKIOSQ > Thresholds for Average IOSQ Time for each disk(ms)  (>0)   [5] |
| $DSKPEND | Disk thresholds $DSKPEND > Thresholds for Average Pending Time for each disk(ms)  (>0)   [2] |
| $ZLNX_MAX_TASK | Execution $ZLNX_MAX_TASK > Number of tasks to process in parallel mode  (1 to 6)   [2] |

# Attachment B – Base and EXCEPTIONS thresholds

| BASE THRESHOLD | ADVANCED THRESHOLD USER EXIT | DESCRIPTION | DEFAULT VALUE |
|---|---|---|---|
| CECAVA | TCECAVA | PERCENT OF CEC AVAILABLE | 5 |
| IFLBUSY | TIFLBUSY | PERCENT OF IFL SYSTEM BUSY | 95 |
| UVMCPUS | TUVMCPUS | VMACHINE TOTAL CPU TIME IN SECONDS | 1800 |
| Z/VMRESP | TZ/VMRESP | AVERAGE SYSTEM RESPONSE TIME IN MILLISECONDS | 10 |
| TVRATIO | TTVRATIO | CPU RATIO OF Z/VM USAGE AND VMACHINE USAGE | 1.5 |
| Z/VMPGIN | TZ/VMPGIN | SYSTEM PAGE-IN RATE | 100 |
| UVMPGIN | TUVMPGIN | VMACHINE PAGE-IN RATE | 10 |
| Z/VMPGSP | TZ/VMPGSP | PERCENT OF SYSTEM PAGE UTILIZATION | 50 |
| DSKRESP | TDSKRESP | AVERAGE DISK RESPONSE TIME IN MILLISECONDS | 30 |
| DSKIOSQ | TDSKIOSQ | AVERAGE DISK IOSQ TIME IN MILLISECONDS | 5 |
| DSKPEND | TDSKPEND | AVERAGE DISK PENDING TIME IN MILLISECONDS | 2 |

# Attachment C – Collecting process activity

Two scripts must be scheduled in the zLinux crontab of each system you want to collect process activity data and executed regularly:

- epvzlinux_collector.sh, to collect process activity measurements; normally scheduled at 5/10 minutes interval;
- epvzlinux_archive.sh, to send the collected measurements to the server where EPV for z/LINUX runs; normally scheduled once a day just after midnight.

An example of both scripts is provided here. It is also included in the TOOLS folder of the installation CD.

## 1) epvzlinux_collector.sh script

```
#!/bin/bash
#
###########################################################
# NOTE: please set this parameters
###########################################################

# full path of data directory to store process activity measurements
CURRENT_DIR=/home/epvuser/PRC/CURRENT

# interval value - it must corresponds to what you set in system crontab
INTRVL=300

###########################################################
# WARNING: Do not modify below this line
###########################################################

HOST=`hostname`
DATE=`date +%y%h%d`
DATETIME=`date +%d/%m/%y:%H:%M:%S`
FILENAME=$CURRENT_DIR/$HOST-$DATE".zlnxprc"

CPU=`cat /proc/cpuinfo | grep 'processor\>' | wc -l`

echo "***ps*** $HOST $DATETIME INT=$INTRVL CPU=$CPU" >> $FILENAME
echo "***header*** ruser comm ppid pid time vsz" >> $FILENAME

ps -Ao "ruser=fullusername" -o "comm ppid pid time vsz" | awk ' $5 != "00:00:00" && $1 != "fullusername" {printf ("%s;%s;%s;%s;%s;%s\n",$1,$2,$3,$4,$5,$6) }' >> $FILENAME
```

**The script scheduling has to be defined in crontab as follows:**

*/5 * * * * /home/epvuser/epvzlinux_collector.sh > /home/epvuser/PRC/CURRENT/zlinux_prc.log 2>&1

**This entry will run the collector script each day every 5 minutes.**

## 2) epvzlinux_archive.sh script

```
#!/bin/bash
#
#######################################################
# NOTE: please set this parameters
#######################################################

# full path of data directory where process activity measurements are collected
CURRENT_DIR=/home/epvuser/PRC/CURRENT

# full path of data directory where process activity measurements are archived
ARCHIVE_DIR=/home/epvuser/PRC/ARCHIVE

# days to keep files in archive directory
DAYS_TO_MAINTAIN=7

# remote host address
REMOTE_HOST=192.168.1.1

# remote username
REMOTE_USR=yourusr

# remote password
REMOTE_PWD=yourpwd

# remote directory
REMOTE_DATA=remote_archive


#######################################################
# WARNING: Do not modify below this line
#######################################################

function proc_ftp {
file_toftp=$1
file_local=$2
ftp -n $REMOTE_HOST <<ENDFTP 2> $CURRENT_DIR/ftplog.log
user $REMOTE_USR $REMOTE_PWD
```

```
prompt off
hash on
cd $REMOTE_DATA
lcd $CURRENT_DIR
put $file_local $file_toftp
quit
ENDFTP
}
cd $CURRENT_DIR

HOST=`hostname`
filegroup=(`ls | grep "\.zlnxprc"`)

for file in ${filegroup[@]}; do

zlnxfile=$CURRENT_DIR/$file
zlnxfile_arch=$ARCHIVE_DIR/$file

filedate_raw=`expr match "$file" '.*\([\-][0-9][0-9][A-Z][a-z][a-z][0-9][0-9][\.]*\)'`
filedate=`expr match "$filedate_raw" '.*\([0-9][0-9][A-Z][a-z][a-z][0-9][0-9]*\)'`

fyearet=${filedate:0:2}
fmonlet=${filedate:2:3}
fday=${filedate:5:2}

case "$fmonlet" in
Jan ) fmon=01;;
Feb ) fmon=02;;
Mar ) fmon=03;;
Apr ) fmon=04;;
May ) fmon=05;;
Jun ) fmon=06;;
Jul ) fmon=07;;
Aug ) fmon=08;;
Sep ) fmon=09;;
Oct ) fmon=10;;
Nov ) fmon=11;;
Dec ) fmon=12;;
esac

let"fyear=($fyearet+2000)"
mday=`date +%d`
mon=`date +%m`
year=`date +%Y`

currdate_fmt=$year$mon$mday
```

*filedate_fmt=$fyear$fmon$fday*

*if [[ $filedate_fmt < $currdate_fmt ]]; then*
*zlnxfile_toftp="$HOST.zlnxprc"*
*proc_ftp $zlnxfile_toftp $file*
*mv $zlnxfile $zlnxfile_arch*

*fi*

*done*

*find $ARCHIVE_DIR \( -name '*.zlnxprc' \) -mtime $DAYS_TO_MAINTAIN -exec rm {} \;*

*exit*

**The script scheduling has to be defined in crontab as follows:**
0 0 * * * /home/epvuser/zlinux_prc_archive.sh > /home/epvuser/PRC/CURRENT/zlinux_prc_tran.log
2>&1

**This entry will run the archive script every day at midnight.**

# Attachment D - Statistical User Exits

| ALERTS NAME | USER EXIT NAME | DESCRIPTION |
|---|---|---|
| STDFUMIP | UESFUMIP | ABNORMAL FUNCTIONS MIPS CONSUMPTION |
| STDSYMIP | UESSYMIP | ABNORMAL SYSTEM MIPS CONSUMPTION |
| STDVMMIP | UESVMMIP | ABNORMAL VMACHINE MIPS CONSUMPTION |

# Related documentation

The following manuals complement the information provided in this manual:

- *EPV zParser V11 Installation and Customization*
- *EPV for zLINUX Plus V4  Database Layout*
- *EPV for zLINUX V4 List of Views*
- *EPV for zLINUX V4 Preparing Input for a Demo*
- *EPV V10 User Interface*
- *EPV Plus V11 Operations Guide*