Monetra®

Payment Server Configuration Guide

Configuration Guide v5.5
Updated September, 2006

The information contained herein is provided “As Is” without warranty of any kind, express or implied, including but not limited to, the implied warranties of merchantability and fitness for a particular purpose. There is no warranty that the information or the use thereof does not infringe a patent, trademark, copyright, or trade secret.

Main Street Softworks, Inc. shall not be liable for any direct, special, incidental, or consequential damages resulting from the use of any information contained herein, whether resulting from breach of contract, breach of warranty, negligence, or otherwise, even if Main Street has been advised of the possibility of such damages. Main Street reserves the right to make changes to the information contained herein at anytime without notice.

Table of Contents

1 Configuring Monetra

1.1 Pre-Configuration

After a successful installation of the Monetra engine, it is time to configure it for your processing requirements.

Four standard (text-based) configuration files (.conf) are used to configure the MONETRA daemon. These configuration files may be edited with your favorite text editor, or as of 4.0, we provide graphical configuration utilities across all major operating platforms including Linux, FreeBSD, Mac OS X and Microsoft Windows (as described below via GUI).

Note 1: These files are **key** to the operation and security compliance of your payment system. It is recommended your systems' integrator take the time to go over the documentation and verify requirements prior to setting any production values.

Note 2: To remain compliant with various card association security programs (such as CISP/PCI/PABP), certain configuration settings described within this document MUST be set to the PRESCRIBED values. All applicable settings should be marked **PCI SECURITY ITEM**.

Disclaimer: While every effort is made by our internal staff to identify and disseminate the most relevant and up-to-date security requirements to our end users, the merchant is ultimately responsible for ensuring that Monetra is configured properly and that all requirements 'throughout the production payment system' are met.

For more information, please visit the Visa website at www.visa.com/cisp/ or the newly formed Security Council at https://www.pcisecuritystandards.org/index.htm

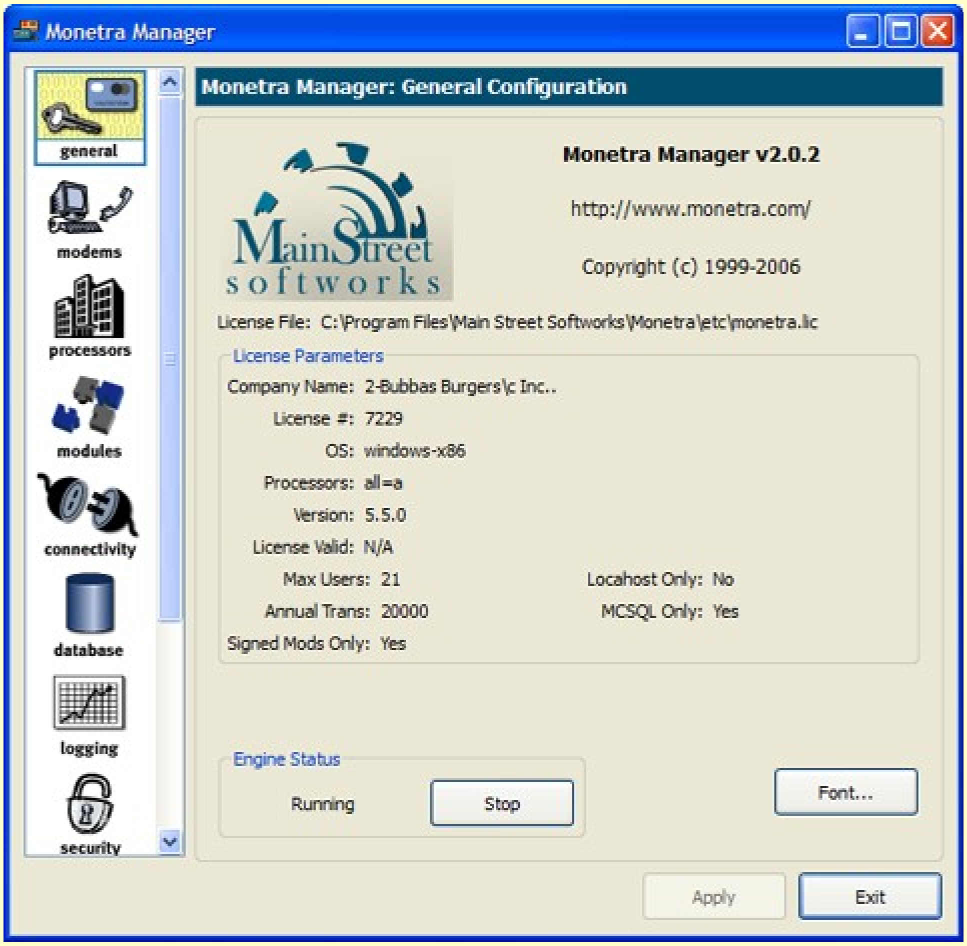
2 Configuration with Graphical User Interface (GUI)

Note: It is not mandatory to use this Manager utility. If you are more comfortable working from a command line and/or text editor, please see section 3 Configuration by Hand.

Although the Manager is quite intuitive, we are providing a full overview of each window and what functions/parameters of Monetra they configure.

For more details on specific parameters used for tuning and/or PCI/CISP security settings, please see the more detailed descriptions in section 3 Configuration by Hand.

2.1 Manager- General



Monetra Manager v2.0.2

http://www.monetra.com/

Copyright (c) 1999-2006

License File: C:\Program Files\Main Street Softworks\Monetra\etc\monetra.lic

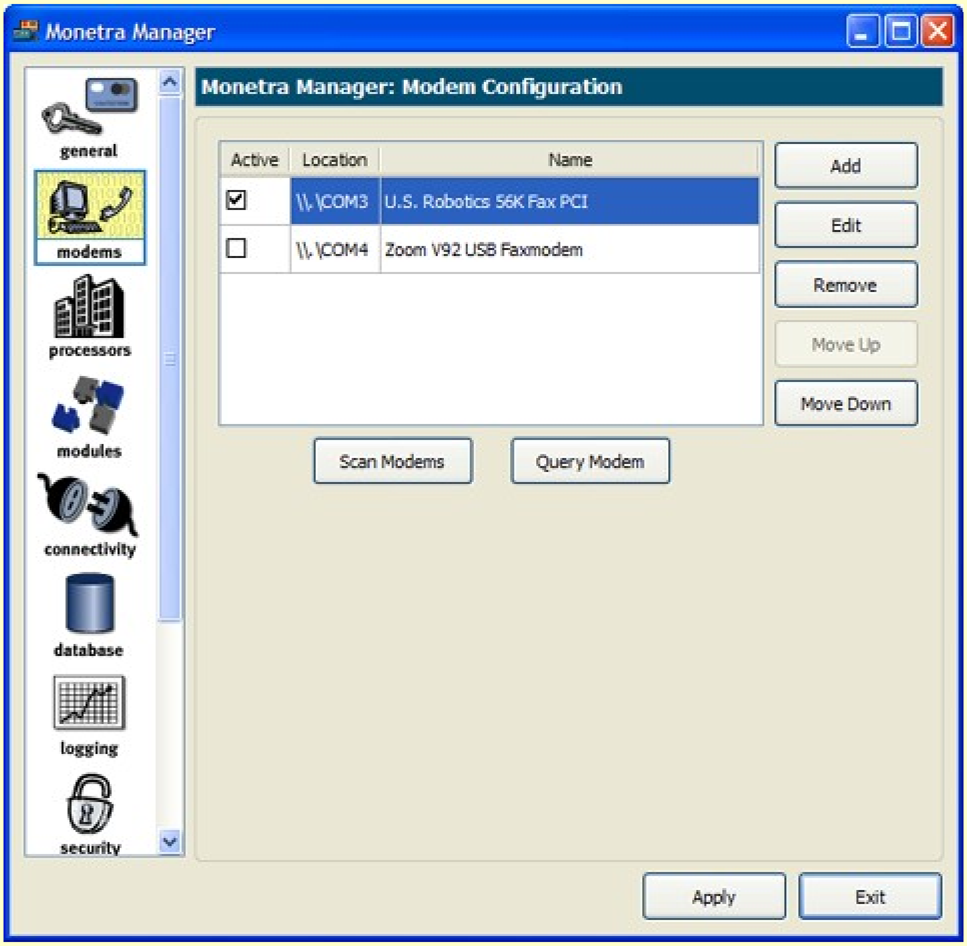
License Parameters

Company Name: 2-Bubbas Burgers\c Inc..
License #: 7229
OS: windows-x86
Processors: all=a
Version: 5.5.0
License Valid: N/A
Max Users: 21
Localhost Only: No
Annual Trans: 20000
MCSQL Only: Yes
Signed Mods Only: Yes

The *General* configuration window is used to Start and Stop the Monetra Server, alongside a window to view current license details.

License Parameters: displays data from the installed license key block .
Engine Status: used to Stop or Start the Payment Server.

2.2 Manager- Modems



Monetra Manager: Modem Configuration

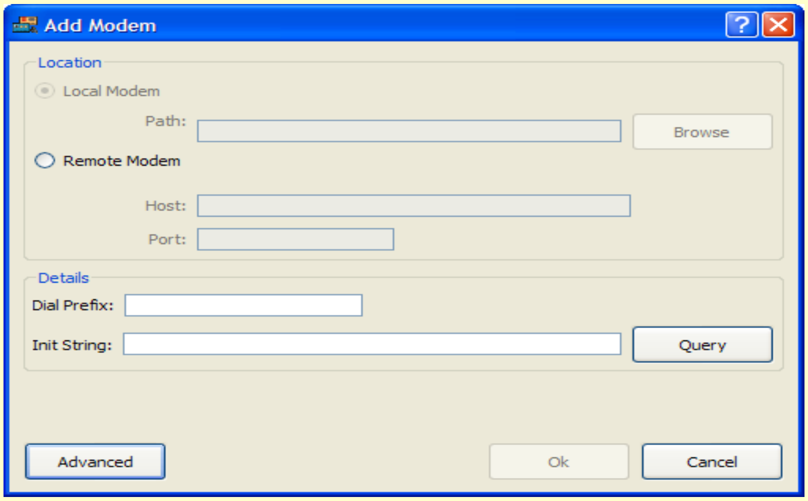
Active	Location	Name
☑	\\.\COM3	U.S. Robotics 56K Fax PCI
□	\\.\COM4	Zoom V92 USB Faxmodem

The *Modem Configuration* window is used to configure one or more dial-up (POTS) modem(s) for use with the Monetra Payment Server.

Table:
Col-1: Check-BOX: 'Active'- denotes an active modem that can be used by the server in stand alone, or pooled (in multi-mode)
Col- 2: Location: Denotes location of device (i.e. Com port etc.)
Col- 3: Name: The name of the modem (as presented by driver)

BUTTON: 'Scan Modems': This will scan the system bus (i.e. Comm ports etc.) to look for available modem
BUTTON: 'Query Modems': This will query/interrogate the selected modem device
BUTTON: 'Add': Add a new modem to the system
BUTTON: 'Edit': Edit a currently active modem
BUTTON: 'Romove': Remove a modem from the system
BUTTON: 'Move Up': Move a selected modem up (in order of use priority)
BUTTON: 'Move Down': Move a selected modem up (in order of use priority)

Note: For more information on INIT strings and how to configure them, please see our online FAQ at http://www.mainstreetsoftworks.com/f/15.html.



Add Modem

Location

● Local Modem

Path:

○ Remote Modem

Host:

Port:

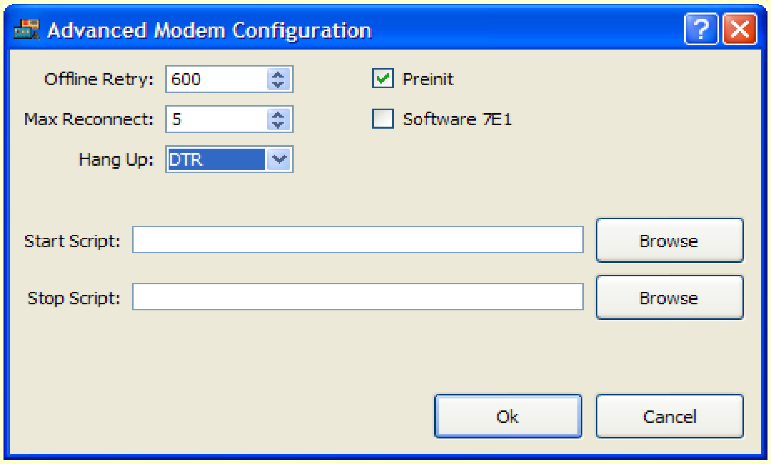
Details

Dial Prefix:

Init String:

Once the 'Add' button is clicked, you will see this add modem dialog. Use it to add a remote modem (on windows) and a local or remote modem on Unix/Unix like operating systems.

Details;
TXT BOX: 'Dial Prefix': Enter a prefix (if required) get a standard dial-tone (i.e. 9,)
TXT BOX: 'Init String': Initialization string to use for this unique modem make/model
BUTTON: 'Query': Use to query/interrogate the modem
BUTTON: 'Advanced': Used to set advanced settings (see table below)



Advanced Modem Configuration

Offline Retry: 600

Max Reconnect: 5

Hang Up: DTR

☑ Preinit

□ Software 7E1

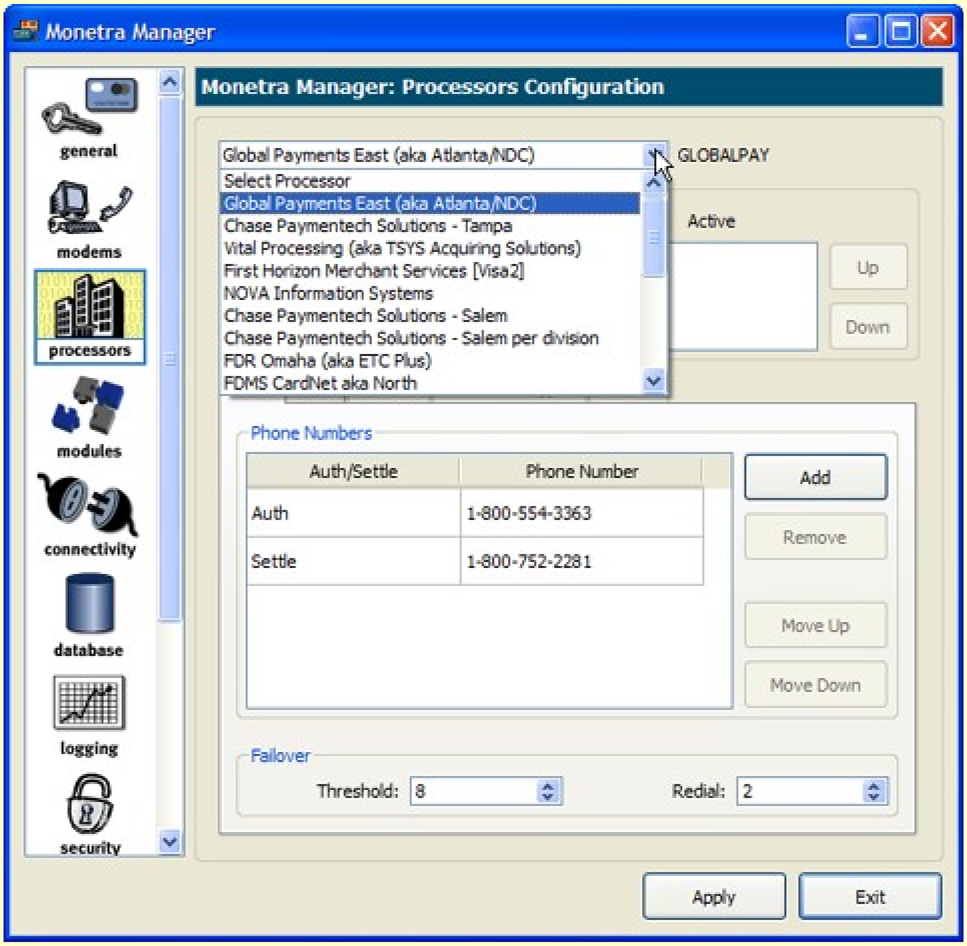
Start Script:

Stop Script:

For most installations the default settings (plus a good init string) should work. If you do need to perform advanced modem functions, here is where they are set.

'Offline Retry': Number of seconds to keep modem off-line before retry attempt
'Max reconnect':After the specified number of connection failures, the modem will be set offline, with a few exceptions. a) it must not be the ONLY modem and b) it must not be the last available modem
'Hang Up': DTR (to hang up by dropping DTR) or COMMAND (to hang up modem by sending +++ATH0)
'Preinit': Turn on/off sending ATZ pre-initialization string to modem
'Software 7E1': Only use if needed for cell-phone 7E1 conversions.
'Start Script':script to execute before accessing modem
'Stop Script':script to execute after done accessing modem

2.3 Manager- Processors



Monetra Manager
general
modems
processors
modules
connectivity
database
logging
security
Select Processor
Global Payments East (aka Atlanta/NDC)
Chase Paymentech Solutions - Tampa
Vital Processing (aka TSYS Acquiring Solutions)
First Horizon Merchant Services [Visa2]
NOVA Information Systems
Chase Paymentech Solutions - Salem
Chase Paymentech Solutions - Salem per division
FDR Omaha (aka ETC Plus)
FDMS CardNet aka North
Active
Up
Down
Add
Remove
Move Up
Move Down
Apply
Exit

Monetra Manager: Processors Configuration

Global Payments East (aka Atlanta/NDC) GLOBALPAY

Phone Numbers

Auth/Settle	Phone Number
Auth	1-800-554-3363
Settle	1-800-752-2281

Failover

Threshold: 8

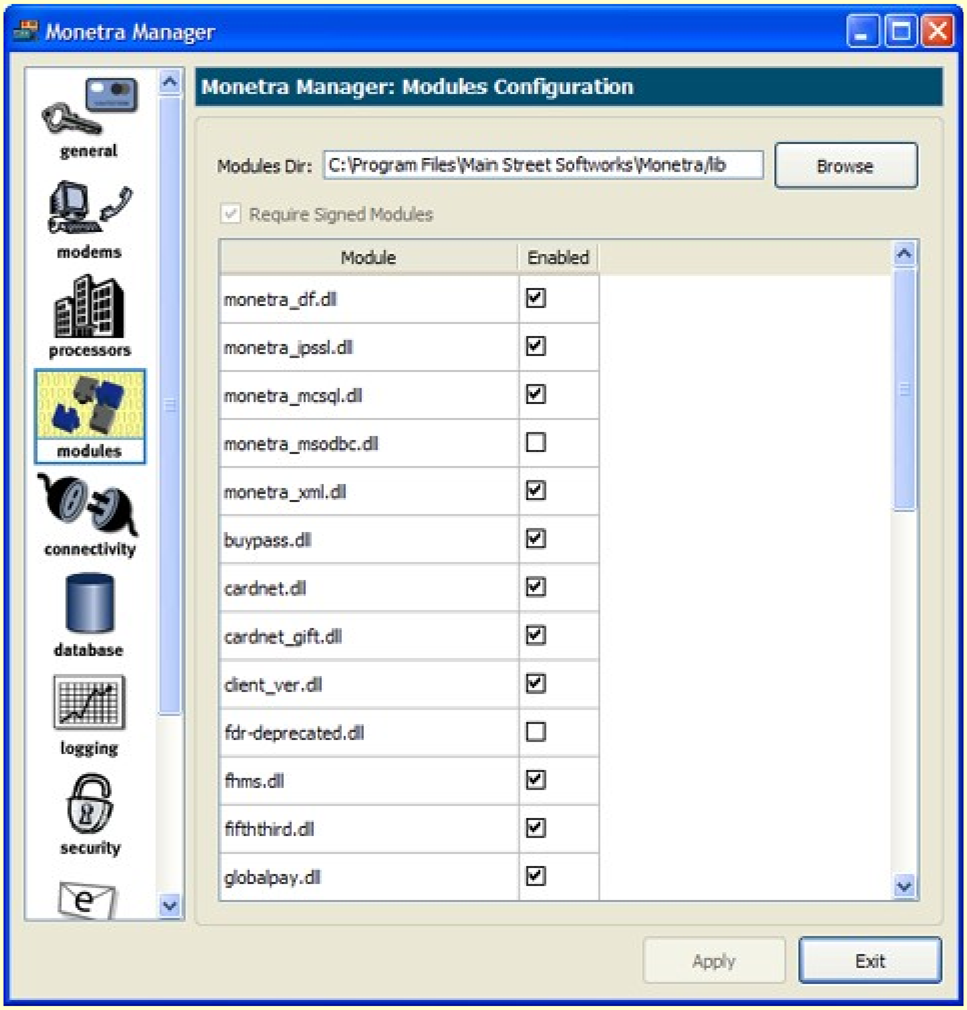
Redial: 2

The *Processors* configuration window is used to configure the connection parameters and processors for use with the Monetra® Server.

Procedure:

1) Choose your processor from the drop-down
2) Check the 'Active' checkbox
3) Move available desired connection methods to the 'Active' box
4) Configure any methods via the TAB for each (example modem: phone numbers)

2.4 Manager- Modules



Monetra Manager: Modules Configuration

Modules Dir: C:\Program Files\Main Street Softworks\Monetra/lib

☑ Require Signed Modules

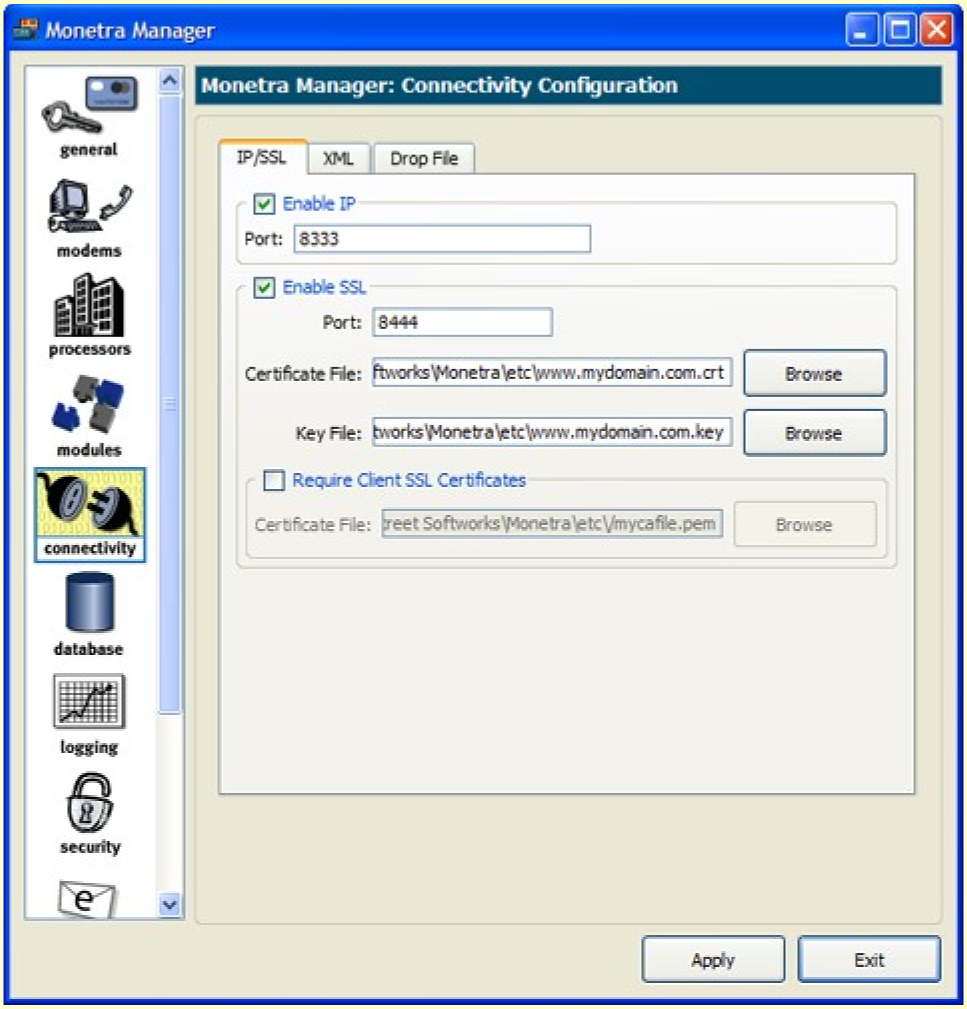
Module	Enabled
monetra_df.dll	☑
monetra_ipssl.dll	☑
monetra_mcsql.dll	☑
monetra_msodbc.dll	□
monetra_xml.dll	☑
buypass.dll	☑
cardnet.dll	☑
cardnet_gift.dll	☑
client_ver.dll	☑
fdr-deprecated.dll	□
fhms.dll	☑
fifththird.dll	☑
globalpay.dll	☑

The *Modules* configuration window is used to configure or identify modules for use with the Monetra® engine.

1 CHECK-BOX: 'Require Signed Modules'- depending on your license key, the engine might require signed modules to be on
2 DIRECTORY-SELECT: 'Modules Dir'- should point to the location of all loadable Monetra modules
3 TABLE: 'Active Modules'- Select/Enable all required modules for your environment

Note: To set parameters for custom loaded modules, see the 'Custom Fields' feature in the ADVANCED section of this manager reference.

2.5 Manager- Connectivity



Monetra Manager: Connectivity Configuration

☑ Enable IP

Port: 8333

☑ Enable SSL

Port: 8444

Certificate File: ftworks\Monetra\etc\www.mydomain.com.crt

Key File: tworks\Monetra\etc\www.mydomain.com.key

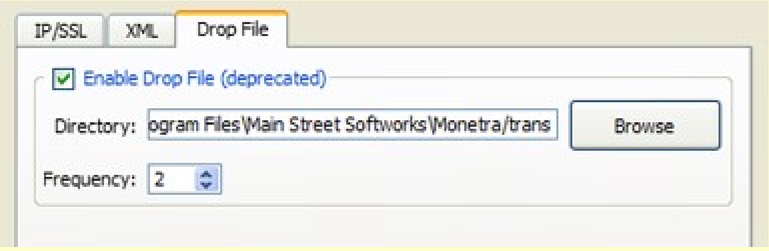
□ Require Client SSL Certificates

Certificate File: reet Softworks\Monetra\etc\/mycafile.pem

The *Connectivity* configuration window is used to set one or more application to Monetra communication methods.
TAB-IP/SSL;
CHECK-BOX: 'Enable IP'- enables TCP/IP communication
TEXT-BOX: 'PORT'- denotes on which IP port Monetra should listen
CHECK-BOX: 'Enable SSL'- enables Secure Socket Layer communication
TEXT-BOX: 'Port Number'- denotes SSL port Monetra should listen on.
TEXT-BOX: 'Certificate File'- SSL Certificate file location
TEXT-BOX: 'Certificate Key File'- SSL Certificate Keyfile location

CHECK-BOX: 'Require Client SSL Certificates'- requires client to have valid SSL cert to connect
TEXT-BOX: 'Certificate File'- SSL Certificate file location

Note: Since Monetra is a true Client<-->Server application, you must configure at least ONE communication method.

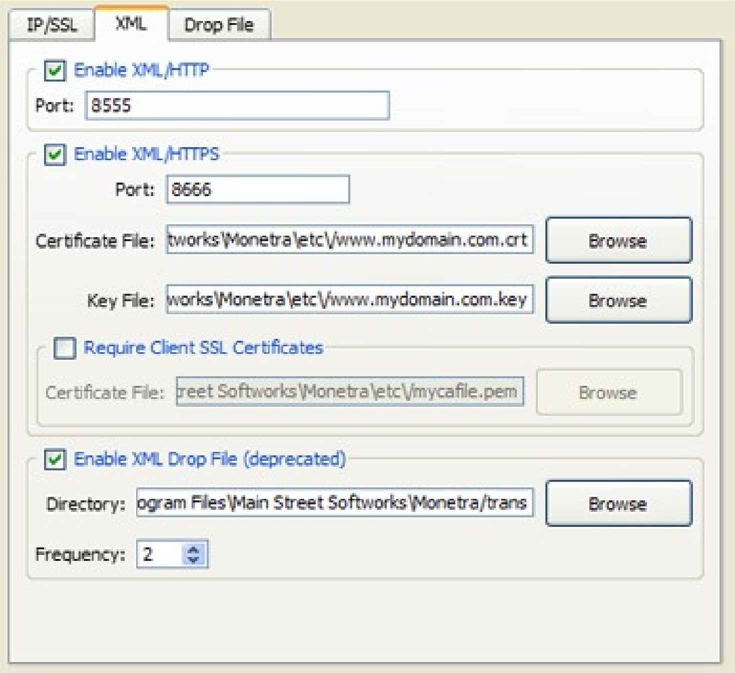


☑ Enable Drop File (deprecated)

Directory: ogram Files\Main Street Softworks\Monetra/trans

Frequency: 2

CHECK-BOX: 'Enable Drop File'- enables text- file- exchange communication
DIRECTORY-SELECT: 'Directory'- should point to the location of directory for which to engage text-based protocol file
SELECT-BOX: 'Frequency'- defines how often (seconds) Monetra should scan directory for new files



☑ Enable XML/HTTP

Port: 8555

☑ Enable XML/HTTPS

Port: 8666

Certificate File: tworks\Monetra\etc\/www.mydomain.com.crt

Key File: works\Monetra\etc\/www.mydomain.com.key

☐ Require Client SSL Certificates

Certificate File: reet Softworks\Monetra\etc\/mycafile.pem

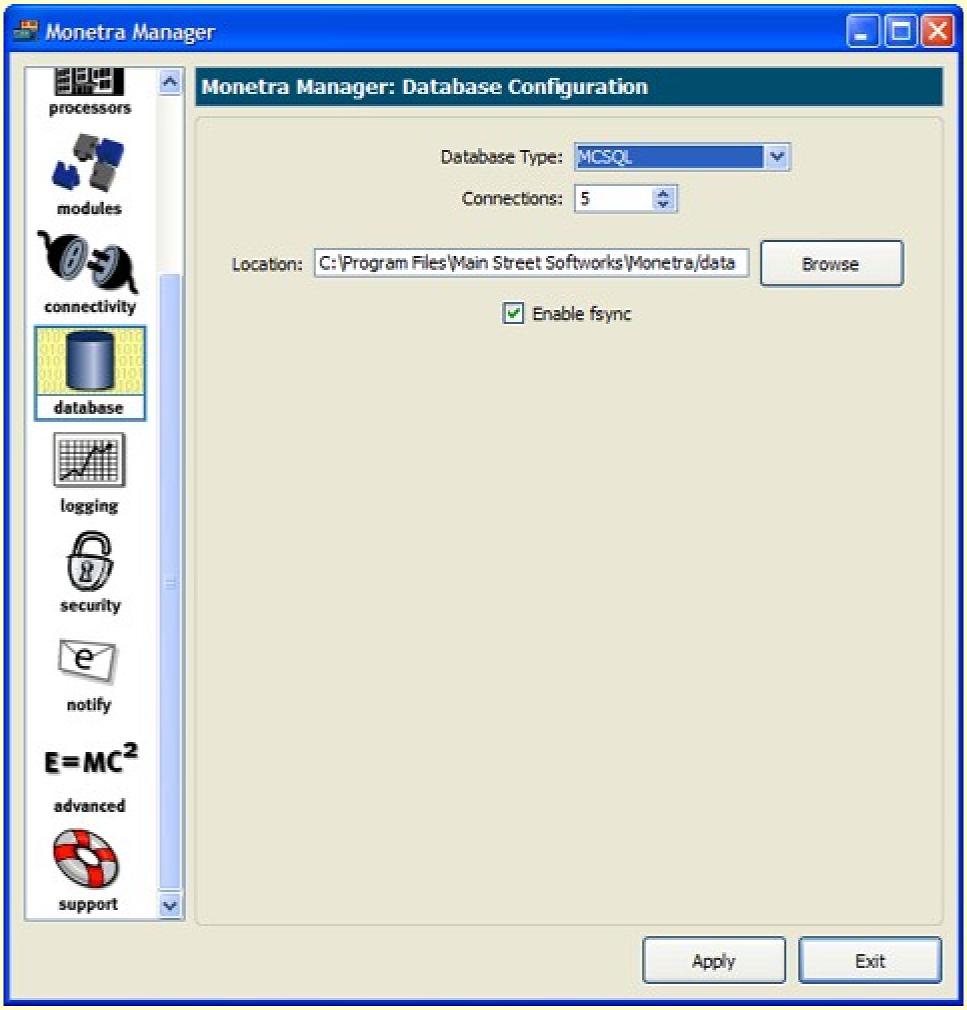
☑ Enable XML Drop File (deprecated)

Directory: ogram Files\Main Street Softworks\Monetra/trans

Frequency: 2

CHECK-BOX: 'Enable XML/HTTP'- enables XML over HTTP (POST) communication
TEXT-BOX: 'Port'- denotes the port on which Monetra should listen for XML HTTP requests
CHECK-BOX: 'Enable XML/HTTPS'- enables XML over HTTPS (POST) communication
TEXT-BOX: 'Port'- denotes the port on which Monetra should listen for HTTPS requests
TEXT-BOX: 'Certificate File'- SSL Certificate file
TEXT-BOX: 'Certificate Key File'- SSL Certificate
CHECK-BOX: 'Require Client SSL Certificates'- requires client to have valid SSL cert to connect
TEXT-BOX: 'Certificate File'- SSL Certificate file
CHECK-BOX: 'Enable XML Drop file'- enables XML text file based communication
TEXT-BOX: 'Directory'- denotes the directory Monetra should scan for XML based text files to process
TEXT-BOX: 'Frequency'- the time, in seconds that Monetra will pause to scan the directory for new XML files waiting to be processed.

2.6 Manager- Database



Monetra Manager: Database Configuration

Database Type: MCSQL

Connections: 5

Location: C:\Program Files\Main Street Softworks\Monetra/data

☑ Enable fsync

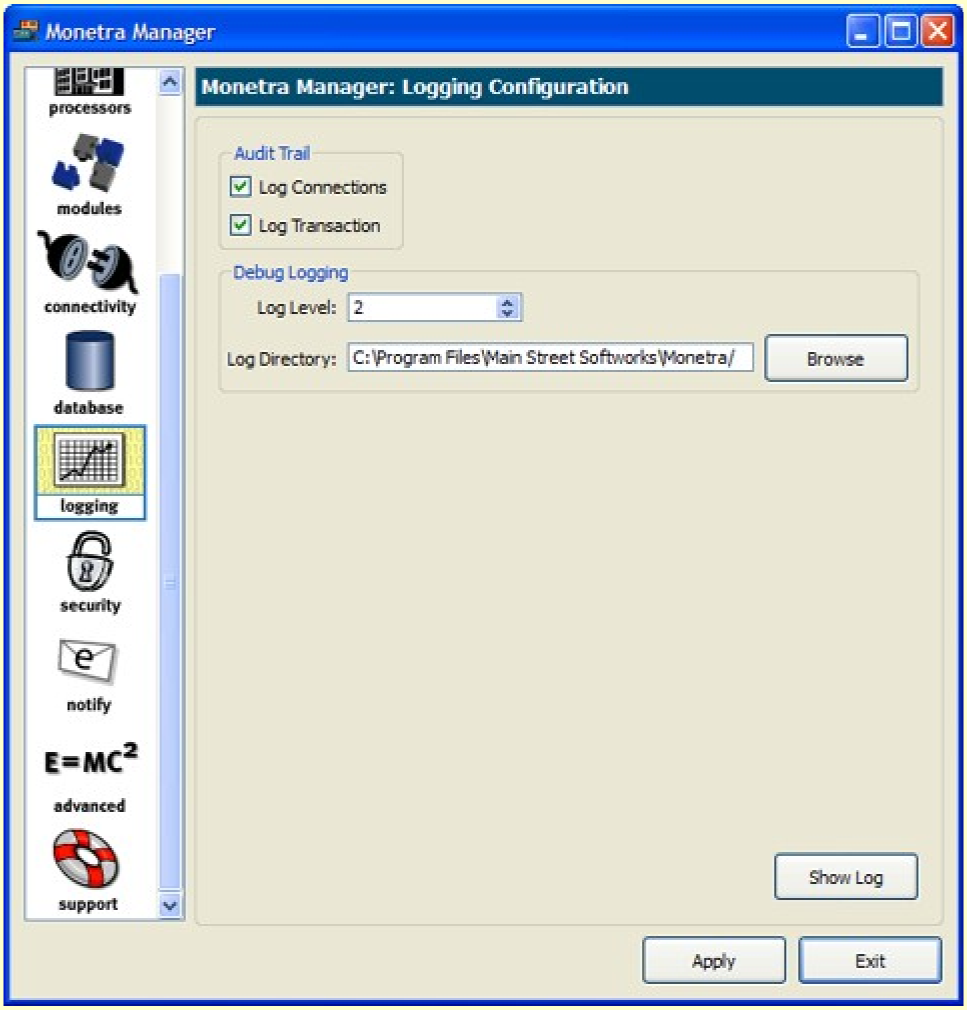
The *Database* configuration window is used to define and configure the default Monetra storage facilities. When using an alternate SQL database (such as mySQL) it is possible to run it remotely.

DROP-DOWN: 'Type'- choose the database TYPE
SELECT: 'Connections'- Number of connections to the DB
TEXT-BOX: 'Location'- Location of database (for monetra database)

Note 1: Depending on the license parameters, you might have more options to configure. For advanced SQL settings, see configuration by hand (section 3)

Note 2: ONLY ONE DATABASE TYPE CAN BE USED PER INSTALL.

2.7 Manger- Logging



Monetra Manager: Logging Configuration

Audit Trail

☑ Log Connections

☑ Log Transaction

Debug Logging

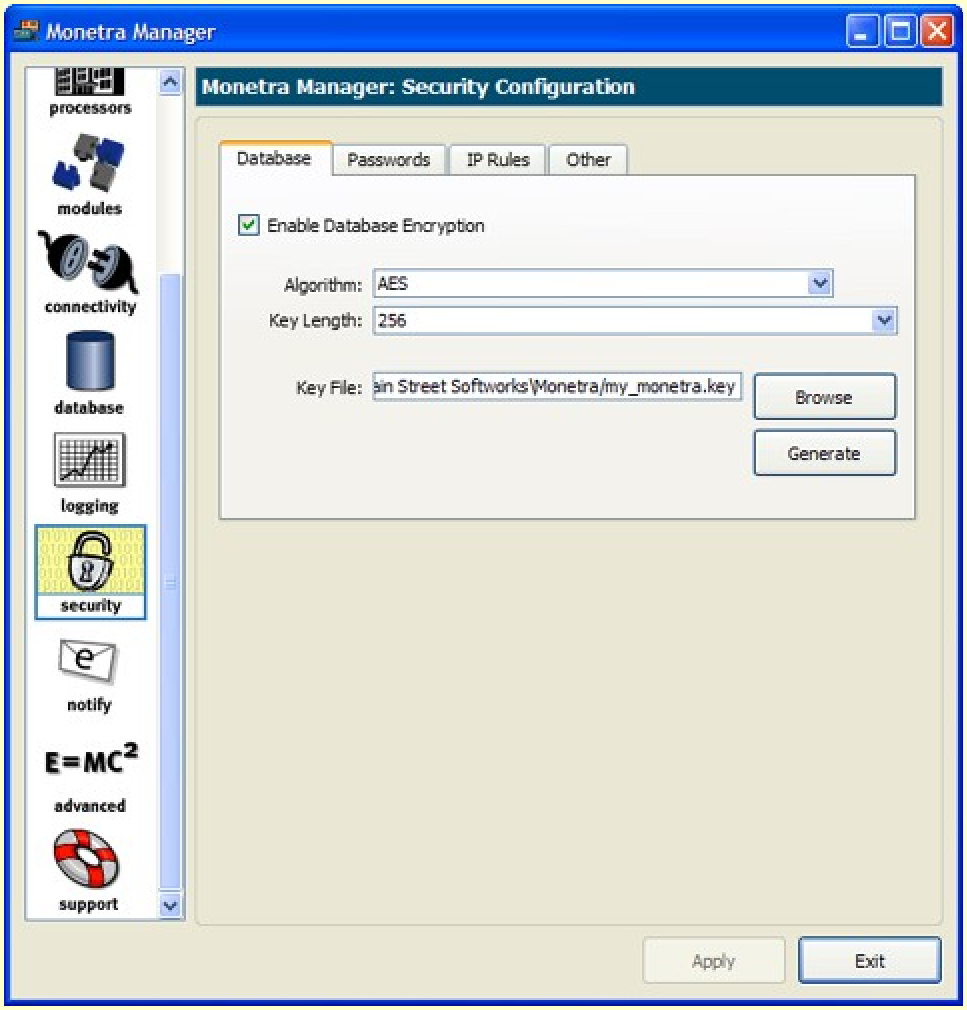
Log Level: 2

Log Directory: C:\Program Files\Main Street Softworks\Monetra/

Note: The Payment Server provides both internal and external data logging facilities. The external file that can be used to troubleshoot both inter-application and application-processor communications is called monetra.log. The amount of data output to this file is configurable.

'Log Connections'- Use internal connection level logging facilities
'Log Transactions'- Use transaction level logging facilities
'Log Level'- defines the system output at the core level
Note: To conform to PCI requirements, you must run (set) the 'Log Level' at a level 2 or lower in a standard production environment.
'Log File Directory'- Location of external system log files

2.8 Manager- Security



☑ Enable Database Encryption

Algorithm: AES

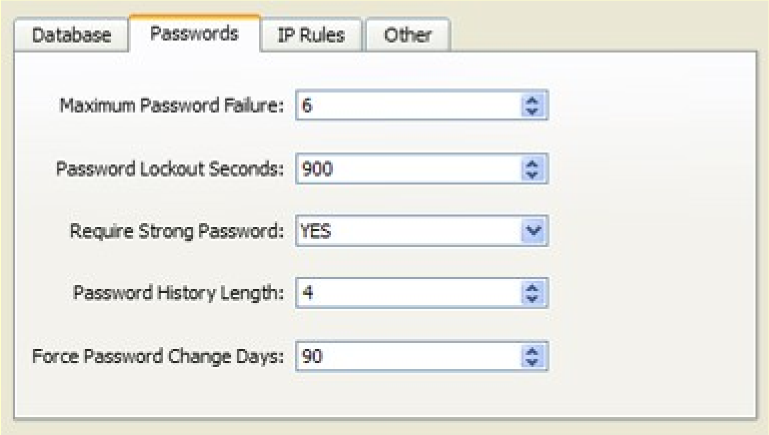
Key Length: 256

Key File: ain Street Softworks\Monetra/my_monetra.key

The security window allows you to configure database, password, application_firewall and set other parameters such as FIPS 140-2 (Other tab).

CHECK-BOX: 'Enable Database Encryption'- Enables database encryption.
DROP-DOWN: 'Algorithm' – Defines which encryption algorithm to apply.
DROP-DOWN: 'Key Length' – Defines the length/strength of KeyFile created/used.
DIRECTORY-SELECT: 'Encryption Key Length- Location to store KeyFile.
BUTTON: 'Generate' – Generate NEW encryption key.

NOTE: PCI SECURITY ITEM: These settings must be active and within PCI specification in a production environment.



Maximum Password Failure: 6

Password Lockout Seconds: 900

Require Strong Password: YES

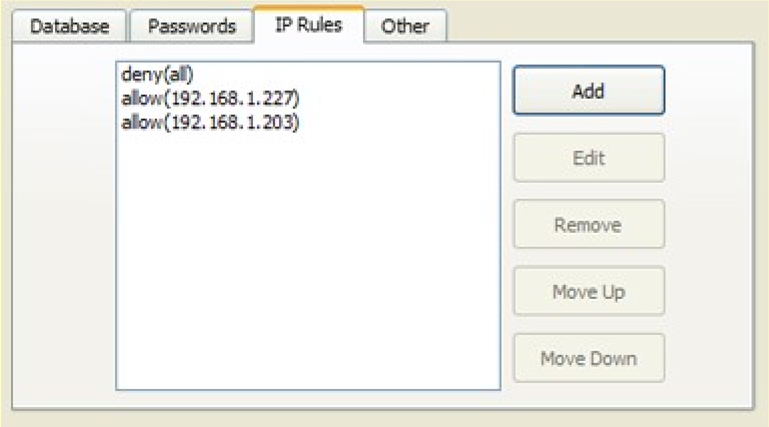
Password History Length: 4

Force Password Change Days: 90

Password management is an important part of strong security and required by the current PCI regulations. By configuring these settings Monetra can handle password management at the application level.

SELECT: 'Maximum Password Failures': Account will be locked after this many fails
SELECT: 'Password Lockout Seconds': Amount of time to lock account
SELECT: 'Require Strong Passwords': Require the use of STRONG passwords
SELECT: 'Password History Length': Number of times between password recycle
SELECT: 'Force Password Change Days': Number of days between force change

NOTE: PCI SECURITY ITEM: These settings must be active and within PCI specification. If these settings are suppressed then IT IS THE SYSTEM ADMINISTRATORS RESPONSIBILITY TO ENSURE COMPLAINCE VIA ALTERNATE PASSWORD MANAGEMENT FACILITIES (i.e. Managed LDAP etc.).

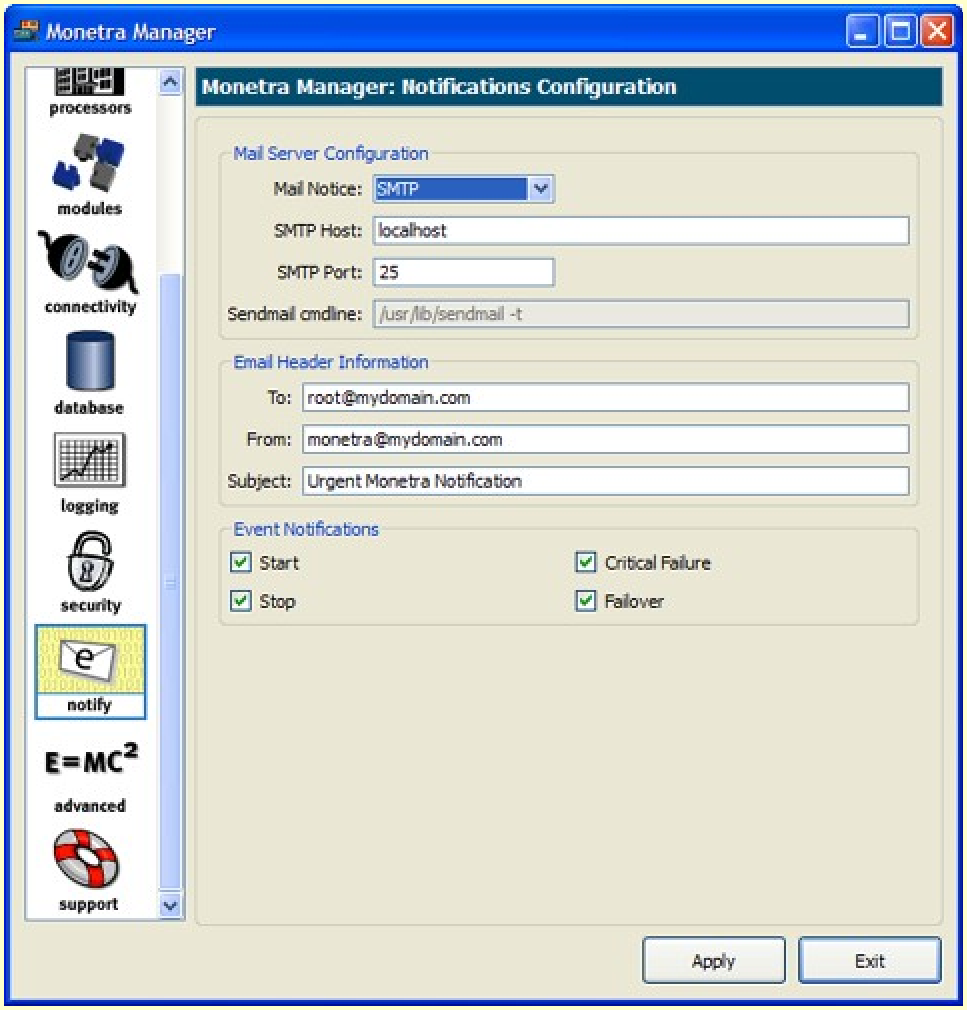


```
deny(all)
allow(192.168.1.227)
allow(192.168.1.203)
```

The IP Ruleset feature works much like a firewall ruleset. For example the rules above would first DENY ALL, then allow the machines at 192.168.1.127 and 192.168.1.203

You may Add, Edit and Remove rules as applicable.

2.9 Manager- Notification



Monetra Manager: Notifications Configuration

Mail Server Configuration

Mail Notice: SMTP

SMTP Host: localhost

SMTP Port: 25

Sendmail cmdline: /usr/lib/sendmail -t

Email Header Information

To: root@mydomain.com

From: monetra@mydomain.com

Subject: Urgent Monetra Notification

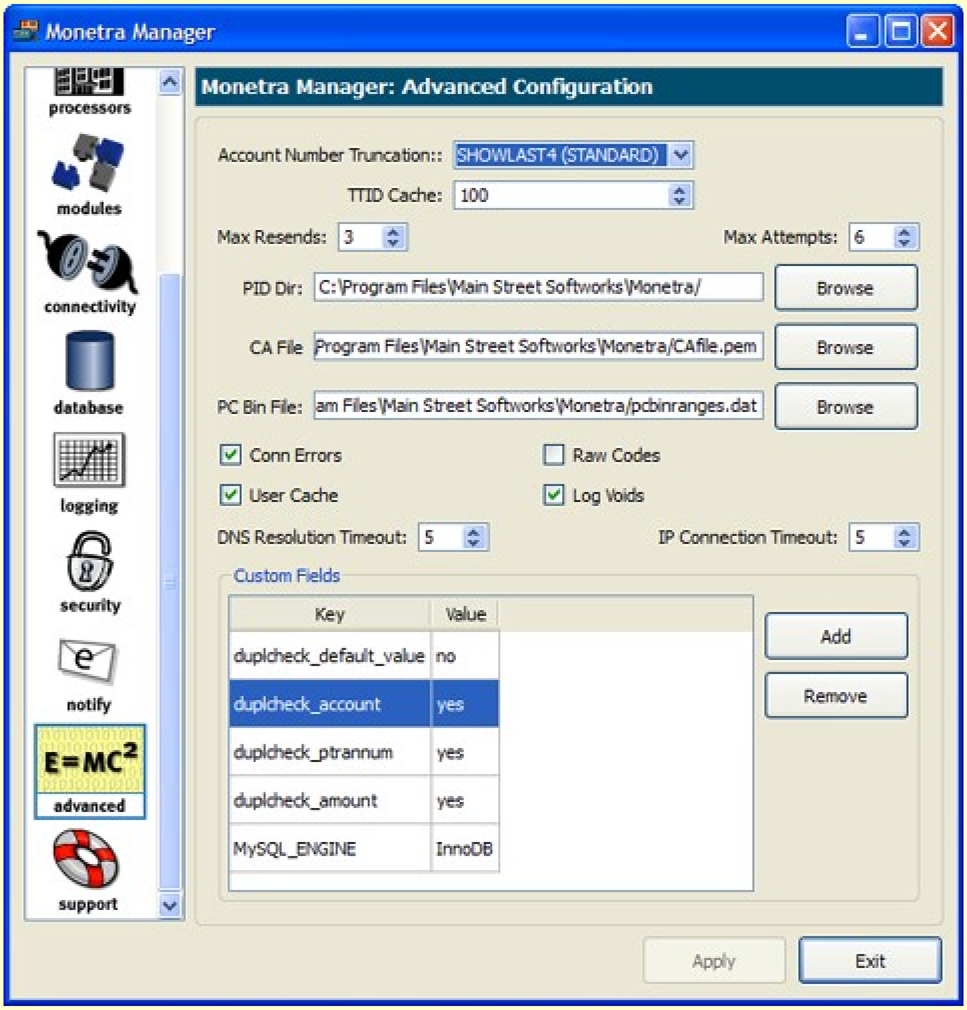
Event Notifications

☑ Start ☑ Critical Failure

☑ Stop ☑ Failover

DROP DOWN: 'Mail Notice'- sets mail delivery mechanism (SMTP, local etc.)
TEXT-BOX: 'SMTP Host'- SMTP mail server host address or DNS name
TEXT-BOX: 'SMTP Port'- SMTP mail server port number
TEXT-BOX: 'Sendmail cmdlinet- Sendmail configuration for Unix users
TEXT-BOX: 'To'- address of message recipients
TEXT-BOX: 'From'- notifications "FROM" address
TEXT-BOX: 'Subject'- notifications SUBJECT line
CHECK-BOX: 'Start'- send Notify on system start
'Stop'- send Notify on system shutdown
'Critical Failure'- send Notify on Abnormal/Critical failure events
'Failover'- send Notify on processor failover events

2.10 Manager- Advanced



Monetra Manager: Advanced Configuration

Account Number Truncation:: SHOWLAST4 (STANDARD)

TTID Cache: 100

Max Resends: 3 Max Attempts: 6

PID Dir: C:\Program Files\Main Street Softworks\Monetra/

CA File Program Files\Main Street Softworks\Monetra/CAfile.pem

PC Bin File: am Files\Main Street Softworks\Monetra/pcbinranges.dat

☑ Conn Errors ☐ Raw Codes

☑ User Cache ☑ Log Voids

DNS Resolution Timeout: 5 IP Connection Timeout: 5

Custom Fields

Key	Value
duplcheck_default_value	no
duplcheck_account	yes
duplcheck_ptrannum	yes
duplcheck_amount	yes
MySQL_ENGINE	InnoDB

Account Number Truncation: defines how Monetra truncates (archives) cardholder data
TTID cache: defines how Monetra should cache transaction ID's
Max Resends: defines how many times Monetra will try a connection method before failing over to alternate
Max Attempts: defines amount of Monetra's connection attempts
Process ID File: location of process ID file on Posix compliant platforms (i.e. Linux, Unix-like and Mac)
CA File: location of engine's Certificate Authority file
PC Bin File: location of Purchase Card BIN file
CHECK-BOX: '**Conn. Errors**'- log connection errors
'**User Cache**'- cache system user info.
'**Raw Codes**'- log raw transaction codes (AVS, CV, etc.)
'**Log Voids**'- log voided transactions
DNS Resolution Timeout: Timer for DNS resolution issues
IP Connection Timeout: Timer for TCP/IP connection issues
TABLE: '**Custom Fields**'- define any custom key=value pairs here (for custom module parameters etc.)

3 Configuration by Hand

Note: Ensure that you make all configuration files located in the /etc/monetra directory "Monetra-readable" only!

Note: Configuration values may be either A=alpha, N=numeric, B=Boolean or a combination of any, and are annotated in the tables below.

Although the configuration files are commented throughout, we are providing a brief overview of the files and what functions of Monetra they configure here.

```
chown -R monetra:monetra @etc_path@/
chmod 770 @etc_path@/
chmod 660 @etc_path@/*
```

3.1 main.conf

3.1.1 Registration Information

This is the **main** MONETRA configuration file and must be edited prior to initial starting of the engine.

NAME:	Registration Key
	The registration license key is located in the client section of our website and can be downloaded alongside the binaries. You may also use the Monetra Installer utility to download and install this file. Note: You must install a proper .lic file for the software to function. Note: Default paths are as follows @etc_path@/monetra.lic example: /etc/monetra/monetra.lic

3.1.2 General Preferences

NAME:	Process ID Location
CONFIG KEY:	*pid=*
TYPE:	A
Default Value:	/usr/local/monetra
Description:	identifies the location of the Process ID file
Notes:	
PCI SECURITY ITEM	
NAME:	Logging Features
CONFIG KEY:	debug=

TYPE:	B/N
Default Value:	yes
DESCRIPTION:	this parameter sets the level of log output
Notes:	Possible values: no, yes, 2, 3, 4, 5 [3-5 dev only])
For PCI compliance, the debug level MUST NEVER BE SET HIGHER THAN 2 in a normal production environment	
NAME:	Logging Directory
CONFIG KEY:	debugdir=
TYPE:	A
Default Value:	@install_path@/
DESCRIPTION:	Example:`C:\Program Files\Main Street Softworks\Monetra\`
NAME:	PC BIN file location
CONFIG KEY:	PCBinFile=
TYPE:	A
Default Value:	/usr/local/monetra/pcbinranges.dat
DESCRIPTION:	File Monetra uses to identify purchase cards.
NAME:	Raw Transaction Codes
CONFIG KEY:	RawCodes=
TYPE:	B
Default Value:	no
DESCRIPTION	Returns the raw transaction codes with responses.
NAME:	Maximum Transaction Resends
CONFIG KEY:	MaxResends=
TYPE:	N
Default Value:	3
DESCRIPTION:	Sets the maximum number of attempts to send a transaction before removing it from queue.
NAME:	*Maximum Transaction Attempts*
CONFIG KEY	MaxAttempts
TYPE:	N
Default Value:	6

DESCRIPTION:	Maximum number of times to attempt a transaction # whether or not a connection was established # and whether or not the data was actually sent # to the processing institution. This avoids # endless loops where you must ForceQuit Monetra # when a transaction gets locked. This occurrence # most likely happens if MONETRA is mis configured.
PCI SECURITY ITEM	
NAME:	Log Connections
CONFIG KEY:	LogConnections=
TYPE:	A
Default Value:	yes
DESCRIPTION:	Logs connections into Monetra. Each connection will be assigned a connection ID, and log information such as start time and end time, as well as the type of connection made, and the reason for closing.
PCI: For PCI application auditing requirements, you must log connections.	
PCI SECURITY ITEM	
NAME:	Log Transactions
CONFIG KEY:	LogTransactions=
TYPE:	A
Default Value:	yes
DESCRIPTION:	Logs transaction history. Stores the time a transaction enters the queue, and the time it was complete, along with the transaction types, return codes and a TTID if applicable. All tied to the incoming connection ID.
PCI: For PCI application auditing requirements, you must log transactions.	

3.1.3 Engine Notifications

NAME:	Notices Configuration
CONFIG KEY:	MailNotices=
TYPE:	A
Default Value:	NONE
DESCRIPTION:	Sets if and how to receive notifications via e-mail

Notes:	NONE= no messages
	SMTP= connect via IP to SMTP server
	EXEC= execute a local copy of sendmail to send e-mail

NAME:	SMTP Host

CONFIG KEY:	SMPT host=
TYPE:	A/N
Default Value:	localhost
DESCRIPTION:	When connecting to an SMTP server, this identifies host
Notes:	
NAME:	SMTP Port
CONFIG KEY:	SMTP port=
TYPE:	N
Default Key:	25
DESCRIPTION:	When connecting to an SMTP server, this identifies port
Notes:	
NAME:	Local Mail Execution Config
CONFIG KEY:	sendmail=
TYPE:	A/N
Default Value:	/usr/lib/sendmail -t
DESCRIPTION:	EXEC settings for sending e-mail via sendmail program
Notes:	
NAME:	Mail To:
CONFIG KEY:	email_to=
TYPE:	A/N
Default Value:	root@localhost.com
DESCRIPTION:	Sets to whom to e-mail the messages
Notes:	
NAME:	Mail From:
CONFIG KEY:	email_from=
TYPE:	A/N
Default Value:	Monetra@localhost.com
DESCRIPTION:	Sets the FROM field in the message
Notes:	Helps prevent mail from being blocked in spam filters
NAME:	Mail Subject
CONFIG KEY:	email_subject=
TYPE:	A/N
Default Value:	URGENT MONETRA NOTIFICATION
DESCRIPTION:	Sets the SUBJECT field in the message
Notes:	

NAME:	Notification of Start
CONFIG KEY:	notify_start=
TYPE:	B
Default Value:	yes
DESCRIPTION:	Configures to send e-mail notification on MONETRA startup.
Notes:	

NAME:	Notification of Stop
CONFIG KEY:	notify_stop=
TYPE:	B
Default Value:	yes
DESCRIPTION:	Configures to send e-mail notification on MONETRA shutdown.
Notes:	

NAME:	Notification SIGSEGV
CONFIG KEY:	notify_sigsegv=
TYPE:	B
Default Value:	yes
DESCRIPTION:	Configures to send e-mail notification on improper MONETRA shutdown.
Notes:	another e-mail is sent when MONETRA is fully online again

3.1.4 Encryption Settings

PCI SECURITY ITEM	
NAME:	Encrypt Data
CONFIG KEY:	dbencrypt=
TYPE:	A
Default Value:	aes
DESCRIPTION:	This identifies the algorithm MONETRA should use to encrypt the data.
Notes:	none: don't encrypt anything local: local key file stdin: grab key from standard input
PCI: Currently, you must set to 'local' or 'stdin' to conform to PCI encryption requirements.	

PCI SECURITY ITEM	
NAME:	Encryption Algorithm
CONFIG KEY:	encalgorithm=
TYPE:	A
Default Value:	aes

DESCRIPTION:	This identifies the algorithm MONETRA should use to encrypt the data.
Notes:	Available algorithms # blowfish: (recommended) # aes: (aka Rjindael) # rc4: # rc5: # idea: # cast5:
PCI: You must use strong encryption such as AES or Blowfish to conform to PCI requirements.	
PCI SECURITY ITEM	
NAME:	Encryption Key Length
CONFIG KEY:	enckeylen=
TYPE:	N
Default Value:	256
DESCRIPTION:	This identifies the algorithmic key length MONETRA should use to encrypt the data.
Notes:	Note1: Variable, dependent on algorithm, please see above. Note2: Keylengths must be evenly divisible by 8
PCI: You must use strong encryption with a key length of 128bits or greater to conform to PCI requirements.	
PCI SECURITY ITEM	
NAME:	Encryption Keyfile Location
CONFIG KEY:	enckeyfile=
TYPE:	A
DEFAULT VALUE:	/usr/local/monetra/my_monetra.key
DESCRIPTION:	This identifies location of the Monetra key
Notes:	To generate a key, use the following shell command: # Monetra_keygen keylen outfile [egd pool] # EX: Monetra_keygen 128 /usr/local/Monetra/my_Monetra.key
PCI: To conform within PCI requirements, you must use a policy and procedure to protect against disclosure or misuse of encryption keys.	

3.1.5 Modem Configuration

NAME:	Modem Identifier
CONFIG KEY:	Modem%d=
TYPE:	A/N
Default Key:	/dev/ttyS0
DESCRIPTION:	Specifies which modem(s) are to be used by MONETRA, in a synchronous fashion.
Notes:	(ex. modem1= modem2= modem3=)
NAME:	Modem Initialization
CONFIG KEY:	Modem%d_init=
TYPE:	A/N
Default Key:	ATS11=35&M0
DESCRIPTION:	This identifies the initialization string on a per-modem basis.
Notes:	Initialization string "noinit" means to send no initialization: This assumes the start and stop scripts properly manipulate the initializations)
NAME:	Modem Pre Initialization
CONFIG KEY:	Modem%d_DoPreInit=
TYPE:	A/N
Default Key:	yes
DESCRIPTION:	Setting to enable/disable pre initialization parameters.
Notes:	By default, MONETRA sends a pre-init string of ATZ\r to the modem before sending the "real" init string. Set this to "no" to prevent MONETRA from sending the pre-init.
NAME:	Modem StartScript
CONFIG KEY:	Modem%d_start=
TYPE:	A/N
Default Value:	
DESCRIPTION:	Script to run before accessing modem.
Notes:	
NAME:	Modem StopScript
CONFIG KEY:	Modem%d_stop=
TYPE:	A/N
Default Value:	
DESCRIPTION:	Script to run after accessing modem.
Notes:	

NAME:	Modem Hangup
CONFIG KEY:	Modem%d_hangup=
TYPE:	A/N
Default Value:	
DESCRIPTION:	DTR or Command
Notes:	DTR (to hang up by dropping DTR) or COMMAND (to hang up modem by sending +++ATH0 to the modem) COMMAND is necessary if using ip:// style arguments.

NAME:	Dial Prefix
CONFIG KEY:	modem%d_dialprefix=
TYPE:	A/N
Default Value:	600
DESCRIPTION:	Number of seconds to keep modem 'offline' before reattempting to use it.
Notes:	Typically 9,,

NAME:	Offline Retry
CONFIG KEY:	modem%d_offlineretry=
TYPE:	A/N
Default Value:	600
DESCRIPTION:	Number of seconds to keep modem 'offline' before reattempting to use it.

NAME:	Max Reconnects
CONFIG KEY:	modem%d_maxreconn=
TYPE:	A/N
Default Value:	5
DESCRIPTION:	After the specified number of connection failures, the modem will be set offline with a few exceptions. a) it must not be the ONLY modem b) it must not be the last available modem

3.1.6 Additional SSL Settings

NAME:	FIPS 140-2
CONFIG KEY:	use_fips_mode=
TYPE:	A
Default Value:	no
DESCRIPTION:	Forces Monetra to use FIPS 140-2 complaint OpenSSL cryptography
Notes:	

NAME:	Certifying Authority file
CONFIG KEY:	CAfile=
TYPE:	A/N
Default Value:	/usr/local/Monetra/CAfile.pem
DESCRIPTION:	Certifying Authority certificates verification file.
Notes:	This is included with the MONETRA distro, and should never be replaced, but may be moved.
PCI SECURITY ITEM	
NAME:	Entropy Gathering
CONFIG KEY:	EGD=
TYPE:	A/N
Default Value:	/var/run/egd-pool
DESCRIPTION:	This is the location of the random data pool as generated by the EGD program.
Notes:	If your system does not have /dev/random or /dev/urandom, you must install an Entropy Gathering Daemon such as EGD or PRNGd.
PCI: Monetra must have some form of entropy to provide required security features. To conform to PCI requirements, data encryption must be activated and you will be required to provide an EGD source for Monetra to enable strong cryptographic support.	

3.1.7 Performance Settings

NAME:	TTID Caching
CONFIG KEY:	TTIDcache=
TYPE:	N
Default Value:	100
DESCRIPTION:	This value will determine how many TTID's to cache.
Notes:	(1) Instead of having to lock, read, and update the database for each transaction insertion, MONETRA can cache available TTIDs in memory. (2) Since TTIDs are on a per-user account basis, a setup with a few hundred merchant accounts, but very few transactions per merchant account would waste memory, and not offer speed improvements. (3) 0 means caching off. (4) There is no maximum range for this number, but in the instance of an improper shutdown, having this number too high could result in many unused TTIDs.
NAME:	User Caching
CONFIG KEY:	EnableUserCache=
TYPE:	B
Default Value:	yes
DESCRIPTION:	With multiple users on the system, this configures caching user parameters.
Notes:	

3.1.8 Password Settings

PCI SECURITY ITEM	
NAME:	Password Failures
CONFIG KEY:	max_password_failures=
TYPE:	N
Default Value:	6
DESCRIPTION:	This value will determine how many failed login attempts before the account is automatically locked.
Notes:	(1) The value 0 represents disabled. (2) If this switch is disabled, the application must provide for PCI requirements.
PCI: You must have complex password security features in place, as per PCI standard 8.5.8 through 8.5.15	

PCI SECURTIY ITEM	
NAME:	Password Lockout
CONFIG KEY:	password_lockout_seconds=
TYPE:	N
Default Value:	900
DESCRIPTION:	This value will determine how long an account is before it is locked out.
Notes:	(1) The value 0 represents disabled. (2) If this switch is disabled, the application must provide for PCI requirements.
PCI: You must have complex password security features in place, as per PCI standard 8.5.8 through 8.5.15	

PCI SECURITY ITEM	
NAME:	Strong Password
CONFIG KEY:	require_strong_passwords=
TYPE:	B
Default Value:	yes
DESCRIPTION:	This value will determine if you require the password to be STRONG. Strong passwords are defined as “at least 7 characters in length and use both letters and numbers.
PCI: You must have complex password security features in place, as per PCI standard 8.5.8 through 8.5.15	

PCI SECURITY ITEM	
NAME:	Password History
CONFIG KEY:	password_history_length=
TYPE:	N

Default Value:	4
DESCRIPTION:	When changing passwords, this is the history to look at to ensure a password is not repeated.
Notes:	(1) The value 0 represents disabled. (2) If this switch is disabled, the application must provide for PCI requirements.
PCI: You must have complex password security features in place, as per PCI standard 8.5.8 through 8.5.15	
PCI SECURITY ITEM	
NAME:	Password Change Timeframe
CONFIG KEY:	force_password_change_days=
TYPE:	N
Default Value:	90
DESCRIPTION:	This is the amount of time allowed prior to a forced password change requirement.
Notes:	(1) The value 0 represents disabled. (2) If this switch is disabled, the application must provide for PCI requirements.
PCI: You must have complex password security features in place, as per PCI standard 8.5.8 through 8.5.15	

3.1.9 Misc Settings

NAME:	DNS Resolution timeout
CONFIG KEY:	dns_resolution_timeout=
TYPE:	N
Default Value:	5
DESCRIPTION:	Number of seconds before Monetra stops attempting DNS resolution
Notes:	High values may adversely affect fail-over processing
NAME:	TCP/IP Connection timeout
CONFIG KEY:	ip_connection_timeout=
TYPE:	N
Default Value:	5
DESCRIPTION:	Number of seconds before Monetra stops attempting IP connection
Notes:	High values may adversely affect fail-over processing
NAME:	Log Voids
CONFIG KEY:	LogVoids=
TYPE:	B
Default Value:	yes
DESCRIPTION:	This setting will determine if Voided transactions get logged in the user history report (GetLog) .
Notes:	A marker ties these entries to the external monetra.log
NAME:	Log Errors
CONFIG KEY:	LogErrors=
TYPE:	B
Default Value:	yes
DESCRIPTION:	This setting will determine if connection errors get logged (errorLog) .
Notes:	A marker ties these entries to the external monetra.log
NAME:	Daemonization
CONFIG KEY:	Daemonize=
TYPE:	B
Default Value:	yes
DESCRIPTION:	Sets whether MONETRA should automatically daemonize itself.
Notes:	For those who are familiar with MONETRA, it does the same thing as specifying the -N switch on the command line.

3.2 prefs.conf

This is the MONETRA general preferences configuration file and may be edited prior to starting the engine.

3.2.1 Communication Methods

NAME:	Enable Internet Protocol (IP)
CONFIG KEY:	enableIP=
TYPE:	B
Default Value:	yes
Description:	Enable IP communication for MONETRA.
Notes:	
NAME:	Port Number
CONFIG KEY:	IPportnum=
TYPE:	N
Default Value:	8333
DESCRPTION:	IP port number to monitor.
Notes:	
PCI SECURITY ITEM	
NAME:	Enable Encrypted (SSL) IP Communications
CONFIG KEY:	enableSSL=
TYPE:	B
Default Value:	yes
DESCRIPTION:	Enable SSL communications within MONETRA.
PCI: For remote administration and secure communications across a public network, secure messaging services must be provided. You must have a secure communications channel such as SSL in use to conform to PCI regulations.	
PCI SECURITY ITEM	
NAME:	SSL Port Number
CONFIG KEY:	SSLportnum=
TYPE:	N
Default Value:	8444
DESCRIPTION:	SSL port number to monitor.
PCI: For remote administration and secure communications across a public network, secure messaging services must be provided. You must have a secure communications channel such as SSL in use to conform to PCI regulations.	

PCI SECURITY ITEM	
NAME:	Use SSL Client Certificates
CONFIG KEY:	SSLCertRequired=
TYPE:	A
Default Value:	no
DESCRIPTION:	Require Client to have valid SSL Certificate file to connect.
PCI: For remote administration and secure communications across a public network, secure messaging services must be provided. You must have a secure communications channel such as SSL in use to conform to PCI regulations.	
NAME:	SSL CA File- (Client)
CONFIG KEY:	SSLCAFile=
TYPE:	A
Default Value:	@etc_path@/mycafile.pem
DESCRIPTION:	SSL CA file.
Notes:	
NAME:	SSL Certificate File
CONFIG KEY:	SSLCertificateFile=
TYPE:	A
Default Value:	/etc/monetra/www.mydomain.com.crt
DESCRIPTION:	SSL Certificate file.
Notes:	
NAME:	SSL Keyle
CONFIG KEY:	SSLCertificateKeyFile=
TYPE:	A
Default Value:	etc/Monetra/www.mydomain.com.key
Notes:	SSL Certificate Key file.
PCI SECURITY ITEM	
NAME:	Enable Drop File
CONFIG KEY:	EnableDF=
TYPE:	B
Default Value:	no
DESCRIPTION:	Enable shared directory support.
PCI: Due to the fact this method writes data directly to the filesystem in plain text, drop files should not be considered a secure method of integration, without extensive alternate system security in place (i.e. File permissions, DMZ etc.). It is the responsibility of the integrator to ensure proper use of Dropfiles.	

PCI SECURITY ITEM	
NAME:	Drop File Directory
CONFIG KEY:	dfdir=
TYPE:	A
Default Value:	/usr/local/Monetra/trans
DESCRIPTION:	Directory or location to monitor.
PCI: Due to the fact that this method writes data directly to the filesystem in plane text, drop files should not be considered a secure method of integration without extensive alternate system security in place (i.e. File permissions, DMZ etc.). It is the responsibility of the integrator to ensure proper use of Dropfiles.	

PCI SECURITY ITEM	
NAME:	Scan Frequency - Directory
CONFIG KEY:	DFscanfrequency=
TYPE:	N
Default Value:	2
DESCRIPTION:	Time in seconds
PCI: Due to the fact this method writes data directly to the filesystem in plane text, drop files should not be considered a secure method of integration, without extensive alternate system security in place (i.e. File permissions, DMZ etc.). It is the responsibility of the integrator to ensure proper use of Dropfiles.	

3.2.2 IP Security

PCI SECURITY ITEM	
NAME:	IP Rulesets
CONFIG KEY:	Iprule%d=
TYPE:	A
Default Value:	Iprule1=allow(all)
DESCRIPTION:	(1) Rulesets that determine valid IP ranges for connections (2) Specify in synchronous fashion (1-50) iprule%d=[allow/deny]([xxx[-xxx].xxx[-xxx].xxx[-xxx].xxx[-xxx]/all]) # Example1: iprule1=deny(all) # Example2: iprule2=allow(192.168.0-255.0-255) # Example3: iprule3=allow(204.152.189.116) (3)Note: the keyword 'all' is equal to 0-255.0-255.0-255.0-255
PCI: For enhanced security we have included an application level firewall. This firewall should be used in addition to hardware firewalls to increase the security of connecting applications.	

3.2.3 Database Configuration

NAME:	Database Type
CONFIIG KEY:	dbtype=
TYPE:	A
Default Value:	MCSQL
DESCRIPTION:	(1)Sets database type to be used, as loaded in the modules.conf Possible values: # MCSQL [MONETRA PROPRIETARY DATABASE WITH SQL FRONTEND] # MYSQL # POSTGRES # ORACLE # DB2 # IODBC #UnixODBC
Notes:	
PCI SECURITY ITEM	
NAME:	SQL User Name
CONFIG KEY:	SQLusername=
TYPE:	A
Default Value:	Monetra
DESCRIPTION:	Sets the username of the database
Notes	If database type is not MCSQL, set the username and password fields.
PCI: For PCI compliance, do not use default admin passwords such as ROOT or SA for database access. PCI standard 8.1 and 8.2	

NAME:	SQL Password
CONFIG KEY:	SQLpassword=
TYPE:	A
Default Value:	secret
DESCRIPTION:	Sets the database password to be used
Notes:	If database type is not MCSQL, set the username and password fields.
NAME:	Database Location
CONFIG KEY:	SQLdbname=
TYPE:	A/N
Default Value:	Monetra
DESCRIPTION:	Name of database for external SQL.
Notes:	
NAME:	Total Database Connections
CONFIG KEY:	SQLconnections=
TYPE:	N
Default Value:	5
DESCRIPTION:	Number of simultaneous database connections maintained.
Notes:	

NAME:	Monetra database Sync
CONFIG KEY:	MCSQL_fsync=
TYPE:	B
Default Value:	Yes
DESCRIPTION:	Sync data after disk writes.
Notes:	Will help to protect data integrity in the event of power loss, crash etc.

3.3Processors.conf

This is the configuration file that must be edited to define transaction communications support from MONETRA to the various processing facilities.

3.3.1 Processor Specific Setup

```
A header is surrounded by [ ]
The internals of the [ header ] are determined by the 'processor' modules
loaded.

Under each header, multiple fields are allowed. Each processor has a specific
subset of the fields which it will accept.  Additional fields will have no
effect.

Please see the appropriate headings for each processor to see the available
field subsets.

NOTES:
  A '%d' denotes an incremental number, starting at 1 and progressing to 50
  unless otherwise specified

- Different types include
* Boolean - [Y]es, [N]o, [T]rue, [F]alse, 1, 0
* Comma Delim. - Comma delimited list of Keys
* Numeric - Numbers only
* Text - Text description

All available fields are as follows:
======== COMMON FIELDS ===================

 -- active --
Type : Boolean
Desc : Set the processor to active or inactive
Ex   :    active=yes

-- conn_priority --
Type : Comma Delim.
Desc : Order to try connectivity methods to processing institution. Available
KEYs are: HTTPS, IP, SSL, OTHER, and DIAL
Ex   : conn_priority="IP,HTTPS,DIAL"
======== DIALUP FIELDS ===================

-- dialup --
Type : Boolean
Desc : Enable/Disable dialup ability for processor
Ex   : dialup=yes

-- threshold --
Type : Numeric
Desc : Number of pending transaction before attempting to grab another modem
Ex   : threshold=8

-- redial --
Type : Numeric
Desc : Number of items to redial before considering Connection a failure
Ex   : redial=5

-- phone%d --
```

```
Type : Text
Desc : List of up to 50 phone numbers to dial for authorizations
Ex   : phone1=1-800-555-4444
       : phone2=1-800-666-5555

 -- settle_phone%d --
Type : Text
Desc : List of up to 50 phone numbers to dial for settlements
Ex   : settle_phone1=1-800-555-4444
       : settle_phone2=1-800-666-5555
```

======= **IP/FRAME FIELDS** ===================

```
-- ip --
Type : Boolean
Desc : Enable/Disable ip/frame ability for processor
Ex   : ip=yes

-- ip_reconn --
Type : Numeric
Desc : Number of times to try to reconnect before trying another connectivity
method
Ex   : ip_reconn=5

 -- ip_offline_retry --
Type : Numeric
Desc : Number of seconds to wait after 'failover' to try this connectivity
method again
Ex   : ip_offline_retry=600

 -- ip_loadbalance --
Type : Boolean
Desc : If multiple ips provided for auth/settlement, should Monetra attempt to
load balance between them? Helpful in cases where multiple leased-lines exist.
Ex   : ip_loadbalance=yes

-- ip_host%d --
Type : Text
Desc : List of up to 50 ip addresses/host names to use for authorizations
Ex   : ip_host1=209.251.159.130
         ip_host2=209.251.159.131

 -- ip_port%d --
Type : Numeric
Desc : List of up to 50 ip port numbers to use for authorizations. This number
_must_ match the number of ip_hosts
Ex   : ip_port1=15000
       : ip_port2=15000

-- ip_settle_host%d --
Type : Text
Desc : List of up to 50 ip addresses/host names to use for settlements
Ex   : ip_settle_host1=209.251.159.130
         ip_settle_host2=209.251.159.131

 -- ip_settle_port%d --
Type : Numeric
Desc : List of up to 50 ip port numbers to use for settlements. This number
_must_ match the number of ip_settle_hosts.



Ex   : ip_settle_port1=15000
       : ip_settle_port2=15000
```

```
Ex   : ip_settle_port1=15000
       : ip_settle_port2=15000

 -- ip_connections --
Type : Numeric
Desc : Number of simultaneous connections allowed. Each connection       will
spawn it's own thread.
Ex   : ip_connections=5

# ================ SSL FIELDS ===================
 -- ssl --
Type : Boolean
Desc : Enable/Disable ssl gateway ability for processor
Ex   : ssl=yes

 -- ssl_reconn --
Type : Numeric
Desc : Number of times to try to reconnect before trying another connectivity
method
Ex   : ssl_reconn=5

-- ssl_offline_retry --
Type : Numeric
Desc : Number of seconds to wait after 'failover' to try this connectivity
method again
Ex   : ssl_offline_retry=600

-- ssl_host%d --
Type : Text
Desc : List of up to 50 ip addresses/host names to use for      authorizations
Ex   : ssl_host1=209.251.159.130
         ssl_host2=209.251.159.131

-- ssl_port%d --
Type : Numeric
Desc : List of up to 50 ip port numbers to use for authorizations.      This
number _must_ match the number of ssl_hosts
Ex   : ssl_port1=15000
       : ssl_port2=15000

 -- ssl_settle_host%d --
 Type : Text
Desc : List of up to 50 ip addresses/host names to use for settlements
Ex   : ssl_settle_host1=209.251.159.130
         ssl_settle_host2=209.251.159.131
-- ssl_settle_port%d --
Type : Numeric
Desc : List of up to 50 ip port numbers to use for settlements. This number
_must_ match the number of ssl_settle_hosts
Ex   : ssl_settle_port1=15000
       : ssl_settle_port2=15000

-- ssl_connections --
Type : Numeric
Desc : Number of simultaneous connections allowed. Each connection will spawn
it's own thread.
Ex   : ssl_connections=5

======== HTTPS FIELDS ===================
-- https --
Type : Boolean
Desc : Enable/Disable https gateway ability for processor
Ex   : https=yes
```

```
 -- https_reconn --
Type : Numeric
 Desc : Number of times to try to reconnect before trying        another
connectivity method
Ex   : https_reconn=5

-- https_offline_retry --
Type : Numeric
Desc : Number of seconds to wait after 'failover' to try this connectivity
method again
Ex   : https_offline_retry=600

-- https_loadbalance --
Type : Boolean
Desc : If multiple ips provided for auth/settlement, should        Monetra
attempt to load balance between them? Helpful in cases where multiple leased-
lines exist
Ex   : https_loadbalance=yes

 -- https_host%d --
Type : Text
Desc : List of up to 50 ip addresses/host names to use for
authorizations
Ex   : https_host1=209.251.159.130
         https_host2=209.251.159.131

 -- https_post%d --
Type : Text
Desc : List of up to 50 http POST addresses. This number        _must_ match
the number of https_hosts
Ex   : https_host1=/cgi-bin/post1
         https_host2=/cgi-bin/post2

-- https_port%d --
Type : Numeric
Desc : List of up to 50 ip port numbers to use for authorizations.
This number _must_ match the number of https_hosts
Ex   : https_port1=15000
      : https_port2=15000

 -- https_settle_host%d --
Type : Text
Desc : List of up to 50 ip addresses/host names to use for settlements
Ex   : https_settle_host1=209.251.159.130
         https_settle_host2=209.251.159.131

 -- https_settle_post%d --
Type : Text
Desc : List of up to 50 http settlement POST addresses. This number _must_
match the number of https_hosts
Ex   : https_settle_host1=/cgi-bin/post1
         https_settle_host2=/cgi-bin/post2

 -- https_settle_port%d --
Type : Numeric
Desc : List of up to 50 ip port numbers to use for settlements. This number
_must_ match the number of https_settle_hosts
Ex   : https_settle_port1=15000
       : https_settle_port2=15000

 -- https_connections --
Type : Numeric
Desc : Number of simultaneous connections allowed. Each connection will spawn
```

```
it's own thread.
Ex   : https_connections=5

FIRST DATA-DATAWIRE CONFIGURATION NOTES:

If  you are given a “primary URL” from your ISO, it must be mapped to our
parameters of HOST, PORT and POST.

For example, if you received a Primary URL like this
https://staging1.datawire.net/sd/ , it would be entered into monetra as
follows:

https_host1=https://staging1.datawire.net
https_port1=443 (note default ssl port is 443)
https_post1=/sd/
```

3.3.2 Global Payments [GlobalPay]

Support for Global Payments, aka NDC includes Dial-up, Leased Line and direct SSL.

Please review the settings for DIAL-UP, IP and SSL as listed above.

An example GlobalPay configuration might look like this.

```
[GLOBALPAY]
active=yes
conn_priority=ssl,dial
dialup=yes
threshold=8
redial=2
phone1=1-800-554-3363
settle_phone1=1-800-554-3363
ip=no
ip_reconn=5
ip_offline_retry=600
ip_addr_loadbalance=no
ip_connections=2
ssl=yes
ssl_reconn=5
ssl_offline_retry=600
ssl_addr_loadbalance=no
ssl_host1=igusproda.globalpay.com
ssl_port1=443
ssl_host2=igusprodb.globalpay.com
ssl_port2=443
ssl_settle_host1=igusproda.globalpay.com
ssl_settle_port1=443
ssl_settle_host2=igusprodb.globalpay.com
ssl_settle_port2=443
ssl_connections=2
```

3.3.3 Vital Processing [Vital]

Support for Vital Processing aka VisaNet includes Dial-up, Leased Line, HTTPS and direct SSL.

Please review the settings for DIAL-UP, IP, HTTPS and SSL as listed above.

An example Vital configuration might look like this.

```
[VITAL]
active=yes
conn_priority=ssl,dial
dialup=yes
threshold=8
redial=2
phone1=1-800-554-3363
settle_phone1=1-800-554-3363
ip=no
ip_reconn=5
ip_offline_retry=600
ip_addr_loadbalance=yes
ip_connections=2
ssl=yes
ssl_reconn=5
```

```
ssl_offline_retry=600
ssl_addr_loadbalance=no
ssl_host1=ssl2.vitalps.net
ssl_port1=5003
ssl_settle_host1=ssl2.vitalps.net
ssl_settle_port1=5003
ssl_connections=2
https=no
https_reconn=5
https_offline_retry=600
https_host1=ssl.pgs.wcom.net
https_port1=443
https_post1=/scripts/gateway.dll?Transact
https_settle_host1=ssl.pgs.wcom.net
https_settle_port1=443
https_settle_post1=/scripts/gateway.dll?Transact
https_connections=2
```

3.3.4 National Processing [NPC]

Note: NPC/BAMS is a settle only platform. Authorizations are typically handled via VITAL processing. As such, Monetra support for BAMS connectivity includes Dial-up and IP (dedicated) Settlement.

Please review the settings for DIAL-UP and IP as listed above.

An example NPC/BAMS configuration might look like this.

```
[NPC]
active=no
conn_priority=dial
dialup=yes
threshold=8
redial=2
settle_phone1=1-866-265-6605
settle_phone2=1-866-265-6606
ip=no
ip_reconn=5
ip_offline_retry=600
ip_connections=2
```

3.3.5 Paymentech [PAYMENTECH]

Support for Paymentech includes Dial-up IP and HTTPS.

Please review the associated communications settings as listed above.

An example Paymentech configuration is as follows.

```
[PAYMENTECH]
active=no
conn_priority=https,dial
dialup=yes
threshold=8
redial=2
ip=no
ip_reconn=5
ip_offline_retry=600
```

```
ip_connections=2
https=yes
https_reconn=5
https_offline_retry=600
https_host1=netconnect.paymentech.net
https_port1=443
https_post1=/NetConnect/controller
https_settle_host1=netconnect.paymentech.net
https_settle_port1=443
https_settle_post1=/NetConnect/controller
https_connections=2
```

3.3.6 First Data- Omaha- [OMAHA]

Support for First Data (Omaha) aka FDR includes Dial-up and HTTPS.

Please review the associated communications settings as listed above.

An example Omaha configuration is as follows.

```
[OMAHA]
active=no
conn_priority=https,dial
dialup=yes
threshold=8
redial=2
phone1=1-800-228-9074
settle_phone1=1-800-228-9074
https=yes
https_reconn=60
https_offline_retry=15
https_host1=vxn1.datawire.net
https_port1=443
https_post1=/sd/
https_host2=vxn.datawire.net
https_port2=443
https_post2=/sd/
https_connections=2
```

3.3.7 Nova [NOVA]

Support for Nova processing includes Dial-up only. Note: Nova can be configured as a split route where authorizations can happen via Vital processing and Settlements can happen via Nova. Please review the associated communications settings as listed above

An example Nova configuration is as follows.

```
[NOVA]
active=no
conn_priority=dial
dialup=yes
threshold=8
redial=2
phone1=1-770-396-7701
settle_phone1=1-770-396-7701
```

3.3.8 First Horizon Merchant Services [FHMS]

Support for First Horizon includes Dial and HTTPS .

Please review the associated communications settings as listed above

An example First Horizon configuration is as follows.

```
[FHMS]
active=no
conn_priority=https,dial
dialup=yes
threshold=8
redial=2
https=yes
https_reconn=5
https_offline_retry=600
https_host1=gateway-bmd.nxt.com
https_port1=443
https_post1=/FTB/process_transaction.cgi
https_settle_host1=gateway-bmd.nxt.com
https_settle_port1=443
https_settle_post1=/FTB/process_transaction.cgi
https_connections=2
```

3.3.9 First Data CardNet [CardNet]

Support for First Data CardNet includes Dial-up and HTTPS.

Please review the associated communications settings as listed above.

An example First Data CardNet configuration is as follows.

```
[CARDNET]
active=no
conn_priority=https,dial
dialup=yes
threshold=8
redial=2
ip=no
ip_reconn=5
ip_offline_retry=600
ip_connections=2
https=yes
https_reconn=60
https_offline_retry=15
https_host1=vxn1.datawire.net
https_port1=443
https_post1=/sd/
https_host2=vxn.datawire.net
https_port2=443
https_post2=/sd/
https_connections=2
```

3.3.10 First Data CardNet GIFT [CardNetGift]

Support for First Data CardNet Gift includes Dial-up and HTTPS.

Please review the associated communications settings as listed above

An example CardNet Gift is as follows.

```
[CardNetGift]
active=no
conn_priority=https
dialup=no
threshold=8
redial=2
ip=no
ip_reconn=5
ip_offline_retry=600
ip_connections=2
https=yes
https_reconn=60
https_offline_retry=15
https_host1=vxn1.datawire.net
https_port1=443
https_post1=/sd/
https_host2=vxn.datawire.net
https_port2=443
https_post2=/sd/
https_connections=2
```

3.3.11 First Data Nabanco [NABANCO]

Support for First Data Nabanco includes Dial-up and HTTPS.

Please review the associated communications settings as listed above.

An example Nabanco configuration is as follows.

```
[Nabanco]
active=no
conn_priority=https
dialup=no
threshold=8
redial=2
ip=no
ip_reconn=5
ip_offline_retry=600
ip_connections=2
https=yes
https_reconn=60
https_offline_retry=15
https_host1=vxn1.datawire.net
https_port1=443
https_post1=/sd/
https_host2=vxn.datawire.net
https_port2=443
https_post2=/sd/
https_connections=2
```

3.3.12 Paymentech Salem [SALEM]

Support for Paymentech Salem includes Dedicated CKT (IP) for Authorizations and Settlement.

Please review the associated communications settings as listed above.

An example Paymentech Salem configuration is as follows.

```
[SALEM]
active=no
conn_priority=ip
dialup=no
ip=yes
ip_reconn=5
ip_offline_retry=600
ip_connections=5
ip_waitdisconnect=86400
```

Alternately there is another module called SALEMDIV. This module should be used when routing transactions on behalf of multiple merchants. And example of that configuration is as follows.

```
[SALEMDIV]
active=no
conn_priority=ip
dialup=no
ip=yes
ip_reconn=5
ip_offline_retry=600
ip_connections=5
```

```
ip_waitdisconnect=86400
```

3.3.13First Data Nashville [Nashville]

Communication support for First Data Nashville includes Dial-up and HTTPS.

Please review the associated communications settings as listed above.

An example Paymentech Salem configuration is as follows.

```
[NASHVILLE]
active=no
conn_priority=https,dial
dialup=yes
threshold=8
redial=2
https=yes
https_reconn=60
https_offline_retry=15
https_host1=vxn1.datawire.net
https_port1=443
https_post1=/sd/
https_host2=vxn.datawire.net
https_port2=443
https_post2=/sd/
https_connections=2
```

3.3.14 Heartland Payment Systems [HEARTLAND]

Communication support for Heartland Payment Systems includes Dial-up and SSL.

Please review the associated communications settings as listed above.

An example Heartland configuration is as follows.

```
[HEARTLAND]
active=no
conn_priority=ssl,dial
dialup=yes
threshold=8
redial=2
phone1=1-800-726-0369
phone2=1-800-253-6049
settle_phone1=1-800-726-0369
settle_phone2=1-800-253-6049
ssl=yes
ssl_reconn=5
ssl_offline_retry=600
ssl_addr_loadbalance=no
ssl_host1=sslprod.secureexchange.net
ssl_port1=22345
ssl_settle_host1=sslprod.secureexchange.net
ssl_settle_port1=22346
ssl_connections=2
```

3.3.15 RBS Lynk [RBSLynk]

Communication support for RBS Lynk (aka Lynk Systems) includes Dial-up, IP and SSL.

Please review the associated communications settings as listed above.

An example RBSLynk configuration is as follows.

```
[RBSLYNK]
conn_priority=ssl,dial
active=no
dialup=yes
threshold=8
redial=2
ip=no
ip_reconn=5
ip_offline_retry=600
ip_addr_loadbalance=yes
ip_connections=2
ssl=yes
ssl_reconn=5
ssl_offline_retry=600
ssl_addr_loadbalance=no
ssl_connections=2
```

3.3.16 Fifth Third Bank [FifthThird]

Communication support for Fifth Third includes Dial-up , IP and HTTPS.

Please review the associated communications settings as listed above.

An example Fifth Third configuration is as follows.

```
[FIFTHTHIRD]
conn_priority=https,dial
active=no
dialup=yes
threshold=8
redial=2
ip=no
ip_reconn=5
ip_offline_retry=600
ip_addr_loadbalance=yes
ip_connections=2
https=yes
https_reconn=5
https_offline_retry=600
```

3.3.17 FDMS BuyPass/Concord/Atlanta [BuyPass]

Communication support for Concord/BuyPass includes Dial-up, IP and HTTPS.

Please review the associated communications settings as listed above.

An example BuyPass configuration is as follows.

```
[BuyPass]
conn_priority=https
active=yes
dialup=no
threshold=8
```

```
redial=2

https=yes
https_reconn=5
https_offline_retry=600
```

3.3.18 First Data ValueLink [ValueLink]

Communication support for First Data ValueLink includes Dial-up and IP.

Please review the associated communications settings as listed above.

An example ValueLink configuration is as follows.

```
[ValueLink]
conn_priority=https,dial
active=no
dialup=yes
threshold=8
redial=2
ip=no
ip_reconn=5
ip_offline_retry=600
ip_addr_loadbalance=yes
ip_connections=2
https=yes
https_reconn=5
https_offline_retry=600
```

3.3.19 Stored Value Systems [SVS]

Support for SVS includes Dial-up and IP.

Please review the associated communications settings as listed above.

An example SVS configuration is as follows.

```
[ValueLink]
active=no
conn_priority=dial
dialup=yes
threshold=8
redial=2
ip=no
ip_reconn=5
ip_offline_retry=600
ip_connections=5
```

3.4 Modules.conf

This is an important file used to configure the modular profile of the MONETRA engine.

3.4.1 Global Modules Configurator

NAME:	Modules Base Path
CONFIG KEY:	module_path=
TYPE:	B
Default Value:	/usr/local/monetra/lib
DESCRIPTION:	Path to base module directory.
Notes:	Ensure the proper permissions are set for this directory
NAME:	Require Signed Modules
CONFIG KEY:	Require_signed_modules=
TYPE:	A
Default Value:	yes
DESCRIPTION:	Require modules to be cryptographically signed.

3.4.2 Communication Modules

These modules create the communication connection(s) from the MONETRA Server to the Client application. Please note that **multiple** communication modules may be loaded per MONETRA server/instance.

NAME:	Load Module
CONFIG KEY:	loadmodule=
TYPE:	B
Default Value:	mymodule.so or mymodule.dll
DESCRIPTION:	Module used to communicate from the client application, to the MONETRA engine, and back.
Notes:	(1)The following modules are distributed and supported Monetra_df.so [drop-file] Monetra_ipssl.so [TCP/IP-SSL] Monetra_xml.so [XML]

3.4.3 Database Modules

These modules configure the data/parameter storage subsystem for the MONETRA daemon. Please note that only **one** database module may be loaded per MONETRA server/instance.

NAME:	Load Module
CONFIG KEY:	loadmodule=
TYPE:	B
Default Value:	mymodule.so
DESCRIPTION:	Module used to provide data/parameter functionality within the target MONETRA system.
Notes:	(1)The following modules are available and supported. Monetra_postegres.so [www.postgresql.org] Monetra_mysql.so [www.mysql.com] Monetra_oracle.so [www.oracle.com] Monetra_db2.so [www.ibm.com/software/data/database/] Monetra_iodbc.so [www.iodbc.org] Monetra_unixodbc.so [www.unixodbc.org]

3.4.4 Processor Modules

These modules configure the outbound and certified processor communications link. These modules are distributed closed source due to legal restrictions and are the modules most closely related to system compliance (updates etc.). Please note that multiple processor modules may be loaded per MONETRA server/instance.

NAME:	Load Module
CONFIG KEY:	loadmodule=
TYPE:	B
Default Value:	mymodule.so
DESCRIPTION:	Module used to provide processor (financial EDI data clearing) functionality/parameters for the target MONETRA system.
Notes:	(1)The following modules are available and supported. globalpay.so [Global Payments] omaha.so [FDMS Omaha] cardnet.so [FDMS CardNet] nabanco.so [FDMS Nabanco] nashville.so [FDMS Nashville] paymentech.so [Paymentech Tampa] salem.so [Paymentech Salem] salemdiv.so [Paymentech Salem divnums] vital.so [Vital Processing] fhms.so [First Horizon Merchant Services] fifththird.so [Fifth Third Bank] rbslynk.so [rbsLynk] nova.so [Nova Information] heartland.so [Heartland Payment] npc.so [National Processing] valuelink.so [ValueLink] svs.so [Stored Value Systems] cardnet_gift.so [FDMS CardNet Gift] buypass.so [FDMS Concord/BuyPass] (2) Module source is closed due to legal restrictions. (3) Main Street may be engaged for custom development/implementation of processor related modules.

3.5 shm.conf

This is an important file used to configure the SHM (memory) version of the MONETRA engine.

3.5.1 SHM Memory Configuration

NAME:	SHM System Size
CONFIG KEY:	SHM_SYS_SIZE_KB=
TYPE:	N
Default Value:	4096
DESCRIPTION:	Max amount of shared memory to allocate.
NAME:	Number of SHM blocks
CONFIG KEY:	MAX_SHM_MEM_BLOCKS=
TYPE:	N
Default Value:	200000
DESCRIPTION:	