Kfsensor MySQL Database Documentation

(2/15/07)

Summary: Kfsensor can log events to a SQL database, including MySQL. This document summarizes the setup steps, documents the Kfsensor tables, and includes an example SQL query.

Setup Summary

1. Create common user account on computer or domain where Kfsensor is installed, and also on computer where MySQL is installed (if on separate computers/domains)
2. Install MySQL (www.mysql.com) on computer to have MySQL database services.
3. Create database in MySQL for Kfensor to use
    a. CREATE DATABASE <database name>
4. Allow ODBC connector user to have access to new database
    a. GRANT ALL ON [database name].* to '[ODBC/MySQL user name]'@'[MySQL IP address or server name]' IDENTIFIED BY '[ODBC/MySQL user's password]'
5. Install MyODBC connector (http://www.mysql.com/products/connector/odbc/) on Kfsensor computer and configure a new odbc System DSN. (http://dev.mysql.com/doc/refman/5.0/en/myodbc-configuration-dsn-windows.html).
6. Enable SQL logging in Kfsensor
    a. MySQL logging is enabled by choosing the *Settings* menu option, then *Log Database*, and filling in the ODBC settings.
    b. Enable the option called *Enable database logging*.
    c. The *Monitor uses database* option tells Kfsensor to pull its event display information from the SQL database. This improves performance.
    d. *DB Latency* should be set to 0 if connecting to MySQL.
    e. *Store binary as text* should be checked if you want to search or query the data contents using SQL.
    f. Click on the *Configure* button when all settings are confirmed. This will create the necessary tables in the ODBC-referenced database.
    g. Once you click OK your will need to restart KFSensor for the changes to take effect.
7. Once the database logging is functional you can optionally load the events from your file logs into the database, using the File->Import Logs Into Database menu option. Also if you want the signature descriptions available in a custom report then these need to be loaded in using the Signatures->Import Signatures Into Database.

Kfsensor Table Summary

mysql>USE [database name];
mysql>SHOW TABLES;

Table Name	Description
kfaction	Lists possible actions that can be taken by a Kfsensor port, such as Close, Ignore, Read and Close, or Sim Std Server
kfclosedby	Who closed connection, client or server
kflog	Main table; list of sensor events; if store binary as text log option is not enabled
kflogt	Main table; list of sensor events; if store binary as text log option is enabled
kfprotocol	Stores allowed protocols (e.g. TCP, UDP, ICMP)
kfseverity	List of possible Kfsensor event criticalities (e.g. High, Medium, Low, None)
kfsigs	List of signatures in Kfsensor
kfsigsource	List of how signature rules were obtained for Kfsensor
kfstatus	List of sensor ports and scenarios and whether a particular sensor port is active in a scenario when a particular event is recorded
kftype	List of events recorded by Kfsensor sensor (ex. C for Connection)
kfver	List of Kfsensor program and database version numbers

Kfsensor Table Fields

kfaction
Description-Lists actions a Kfsensor port can take

Field Name	Type	Size	Description
action	Character	1	Primary key; letter representing action name
actionname	Var Character	30	Description of action

Default Values:

Action	Action name
A	Native
B	SimBanner
C	Close
E	SimExternal
I	SilentClose
M	SensorMonitorConnection
N	Sniff
R	ReadAndClose
S	SimServer
T	SimStdServer
X	NotSpecified

kfclosedby

Description-List of who closed the connection

Field Name	Type	Size	Description
severity	Character	1	Primary key; letter representing who closed the connection
severityname	Var Character	30	Description of host type who closed the connection

Default Values:

Severity	Severityname
C	Client
S	Server

kflog or **kflogt**

kflog description-List of sensor events; if store binary as text log option is not enabled
kflogt description-List of sensor events; if store binary as text log option is enabled

Field Name	Type	Size	Description
sensorid	Var Character	50	key; sensor's name, useful if you have multiple sensors
id	Interger	11	event id number, increments sequentially for each event over time
evtype	Character	1	Type of event recorded during connection [seems that it wants to be linked to kftype table, because they share same field name and many of the same values, but there are some unshared values-Roger]
action	Character	1	action taken by sensor port during event; possible values listed in kfaction table (ex. R for ReadandClose, N for Native)
name	Var Character	50	Event description (ex. TCP Closed Port, SSH, etc.)
description	Var Character	100	Extra event specific information; example values include NULL, Reset, Scan, Timed out, Port Scan warning, etc.
simname	Var Character	50	If a SimStd server was involved in the event, what was the SimStd server's name (ex. Telnet, FTP Guild, SMTP, etc.)
protocol	Character	1	key; indicates what protocol attack occurred on (e.g. TCP, UDP, ICMP). Values listed in

			kfprotocol table
severity	Character	1	Letter representing Kfsensor event criticality (i.e. High, Medium, Low, None)
starttm	Datetime	n/a	Start Date and time of the event recorded
startms	Small Integer	6	Start time of event, the millisecond portion
endtm	Datetime	n/a	End Date and time of the event recorded
endms	Small Integer	6	Time event ended, the millisecond portion
clientdomain	Var Character	255	Source client's domain name if known
clientip	Var Character	20	Source client IP address recorded in event
clientport	Integer	11	Source client port number recorded in event
bindip	Var Character	255	If the Sensor was bound to a specific IP then this will be listed
hostip	Var Character	20	Destination IP address recorded in event, usually sensor's IP address or a broadcast address
hostport	Integer	11	Destination port number recorded in event
conclosedby	Character	1	Who closed the connection, source client or Kfsensor server
limitexceed	Character	1	whether (i.e. T) or not (i.e. F) the amount of data received by the sensor exceeded the limit. If true then only some of the data received will be stored.
recbytes	Longblob	n/a	Number of bytes received from source client connection; stored in binary (in kflog) or text form (in kflogt)
sentbytes	Integer	11	Number of bytes sent back from sensor to source client connection in response to connection
sent	Longblob	n/a	Data sent from Kfsensor to source client; stored in binary (in kflog) or text form (in kflogt)
sigid	Var Character	12	Signature ID of event if any was attached
sigaction	Character	1	Signature rule action, if any is set, otherwise NULL

kfprotocol
Description- Stores allowed protocols (e.g. TCP, UDP, ICMP)

Field Name	Type	Size	Description
protocol	Character	1	Primary key; single letter representing protocol name
protocolname	Var Character	30	Protocol name description

Default Values:

protocol	protocolname
I	ICMP
T	TCP
U	UDP
X	NotSpecified

kfseverity
Description-List of possible Kfsensor event criticalities

Field Name	Type	Size	Description
severity	Character	1	Primary key; letter representing event criticality levels
severityname	Var Character	30	Event criticality levels

Default Values:

severity	severityname
L	Low
M	Medium
H	High
N	None

kfsigs
Description-List of Kfsensor event detection signatures

Field Name	Type	Size	Description
sigid	Character	12	Primary key;
message	Var Character	255	Signature description
rsref	Var Character	255	External reference to source
timeCreated	Datetime	n/a	Date/time it was created
timeArchived	Datetime	n/a	Date/time it was archived
timeEdited	Datetime	n/a	Date/time it was last edited
rarchived	Character	1	T if marked as archive
ractive	Character	1	T if active
ruleSource	Character	1	How the signature was obtained

Default Values:
XXX XXXX

kfsigsource

Description-List of how signature rules were obtained for Kfsensor

Field Name	Type	Size	Description
ruleSource	Character	1	Primary key; letter representing rule source name
rsname	Var Character	30	rule source name, where rule was obtained from

Default Values:

ruleSource	rsname
E	External
I	Imported
K	Keyfocus
H	Handcoded

kfstatus

Description-List of sensor ports and scenarios and whether a particular sensor port is active in a scenario when a particular event is recorded

Field Name	Type	Size	Description
sensorid	Var Character	50	key; name of sensor
scenario	Var Character	50	Name of active scenario being used on sensor
name	Var Character	50	Name of possible ports used in a scenario (e.g. Telnet, SSH, etc.)
ipaddress	Var Character	255	IP address sensor is bound to, if bound to a particular sensor IP address; if not bound to particular port value is NULL
port	Integer	11	Port number used by sensor port
protocol	Character	1	Protocol used by sensor port; possible values stored in kfprotocol table
action	Character	1	Default action of sensor port when connected to; possible values stored in kfaction
actionname	Var Character	50	Name of sensor port if there is one; otherwise NULL
active	Character	1	Records whether port is active (i.e. T) or not active (F) in scenario
isrunning	Integer	11	Internal status flag

kftype

Description-List of events recorded by Kfsensor sensor

Field Name	Type	Size	Description
evtype	Character	1	Primary key; single character representing event type name

evtypename	Var Character	30	Event type name

Default Values:

evtype	evtypename
E	Error
C	Connection
M	Monitor Error
D	DOS Attack
U	DOS Lock Up
S	Mail Sent
P	ICMP Ping
T	ICMP Time Stamp
A	Port Scan
X	NotSpecified

kfver

Description-List of Kfsensor program and database version numbers

Field Name	Type	Size	Description
dbvermajor	Integer	11	Kfsensor database major version number
dbverminor	Integer	11	Kfsensor database minor version number
kfvermajor	Integer	11	Kfsensor program major version number
kfverminor	Integer	11	Kfsensor program minor version number
kfverrelease	Integer	11	Kfsensor program minor minor version number
upddatedate	datetime	n/a	When Kfsensor database was last updated

Useful Example Query

```
SELECT     kflogt.sensorid, kflogt.id, kftype.evtypename,
kfaction.actionname, kflogt.description, kflogt.simname,
kfprotocol.protocolname,
                      kfseverity.severityname, kflogt.starttm,
kflogt.endtm, kflogt.clientdomain, kflogt.clientip, kflogt.clientport,
kflogt.bindip, kflogt.hostip, kflogt.hostport,
                      kflogt.conclosedby, kflogt.limitexceed,
kflogt.recbytes, kflogt.received, kflogt.sentbytes, kflogt.sent,
kflogt.sigid, kflogt.sigaction, kfsigs.message
FROM         kfprotocol RIGHT OUTER JOIN
                      kflogt LEFT OUTER JOIN
                      kfseverity ON kflogt.severity =
kfseverity.severity LEFT OUTER JOIN
                      kfclosedby ON kflogt.severity =
kfclosedby.severity LEFT OUTER JOIN
                      kftype ON kflogt.evtype = kftype.evtype LEFT
OUTER JOIN
```

```
                      kfaction ON kflogt.action = kfaction.action ON
kfprotocol.protocol = kflogt.protocol LEFT OUTER JOIN
                      kfsigs ON kflogt.sigid = kfsigs.sigid
```