Log Analyzer for Dummies

Emilio Valente

LOG ANALYZER for Dummies

GCIH Gold Certification

Author: Emilio Valente, evalente@sdsc.edu

Advisor: James E. Purcell

Accepted: December 10, 2007

1. Introduction

ABSTRACT

Syslogging is an important aspect of troubleshooting. It helps keep an eye on what is happening on the network or reconstruct what happened (forensic analysis).

Many devices in the network (end-systems, network devices, appliances) usually create a large amount of information. It is difficult to monitor in real-time hundreds and hundreds of log messages per minute.

In my opinion there should be a simple type of automation in the form of a network log analyzer tool that through an easy-to-use friendly GUI and keywords it searches a database (queries) and allows the sysadmin to catch the right thing quickly.

There are expensive and sophisticated tools selling for thousands of dollars that assist the sysadmin in this matter but the discussion in this paper is something new: a network management logging tool for "dummies".

The components that make Syslogging are quite standard: sending device, centralized receiver, database and friendly user interface.

With a few simple existing tools I will explain how even an entry-level sys-administrator can easily build an effective and inexpensive network log analyzer. What I call "Log Analyzer for dummies"; is a versatile and stable tool, with a minimal cost, it can be easily installed in any environment, it can

support most devices, and almost any vendor, with large storage capability.

This Network Log Analyzer can be an invaluable tool for every sysadmin in the "Identification" phase of the Incident Handling process.

2. Milestone

a) Brief description of what a Syslogger does and what companies offer.

In general a centralizer Syslogger collects and stores Syslog messages sent by each configured device on the network (LAN and WAN): switches, routers, systems, appliances, or any devices that is able to create and send a simple log message.

There are many companies out there that have products off-of-the-shelf that are designed to collect and store the logs in a relational database, convert them to a desirable format and present it on a well enough friendly GUI to be used by sysadmin to troubleshoot issues.

Also, I should mention that some of the above tools have the so-called "intelligence" which, in addition to the previous cited features, they have the ability to correlate events and execute actions appropriately (ex.: shut down a switch port against a DoS attack).

Of course everything comes with a price. These companies sell a well-finished package for tens and even hundreds of thousand dollars. In particular I have tested three (3) companies' products and the prices ranged from $ 35,000 to $ 60,000.

At this point, when I was aware of the degree of the technical expertise necessary to build a reasonable tool, I realized that I had all that I need. Expertise in Syslogging, network devices, systems, databases, web server GUI; therefore I decide to take the adventure and build an inexpensive one by myself. The goal was to put together an architecture that allows, in whatsoever environment, a quick detection of an incident occurring (going on) or already happened shortly (few hours ago) during the "Identification" phase of the Incident Handling procedure.

Hereafter are the details and I hope this may help you to do the same.

b) Components of logging in details.

- **Relational Database**
- **Centralized Syslogger**
- **Web Interface**
- **Reports**

• Relational Database

It is your choice; you can install all 3 components on the same system. My recommendation is to install each component on a different system if you have available the necessary hardware. At least the database should reside on a different partition if you use the same system.

First you have to install the kind of database you wish to use (MySQL, Postgress, etc.). I have used Microsoft SQL because we already had a commercial license for it.

Next the Centralized Syslogger needs to be installed (on the same or on a remote system) and needs to be configured.

• Centralized Syslogger

I used Kiwi Syslogger (but you can use whichever you wish as long as it has the same functionality) that is generally free. Unfortunately, for this project we cannot use the free edition and this is actually the only expense that is necessary to build our Network Log Analyzer.

The commercial version (circa $159.00) gives us the possibility to Log to an ODBC database (Access/SQL/Oracle/MySQL/Informix etc) while the free edition doesn't have that needed feature.

Kiwi Syslogger Daemon runs on: Windows 98/ME, NT4/2000/2003, XP/Vista. I have installed it on a Win Server 2003; below are the complete guide and settings to make the Kiwi Syslogger sending log

messages to your database:

Centralized Syslogger configuration steps:

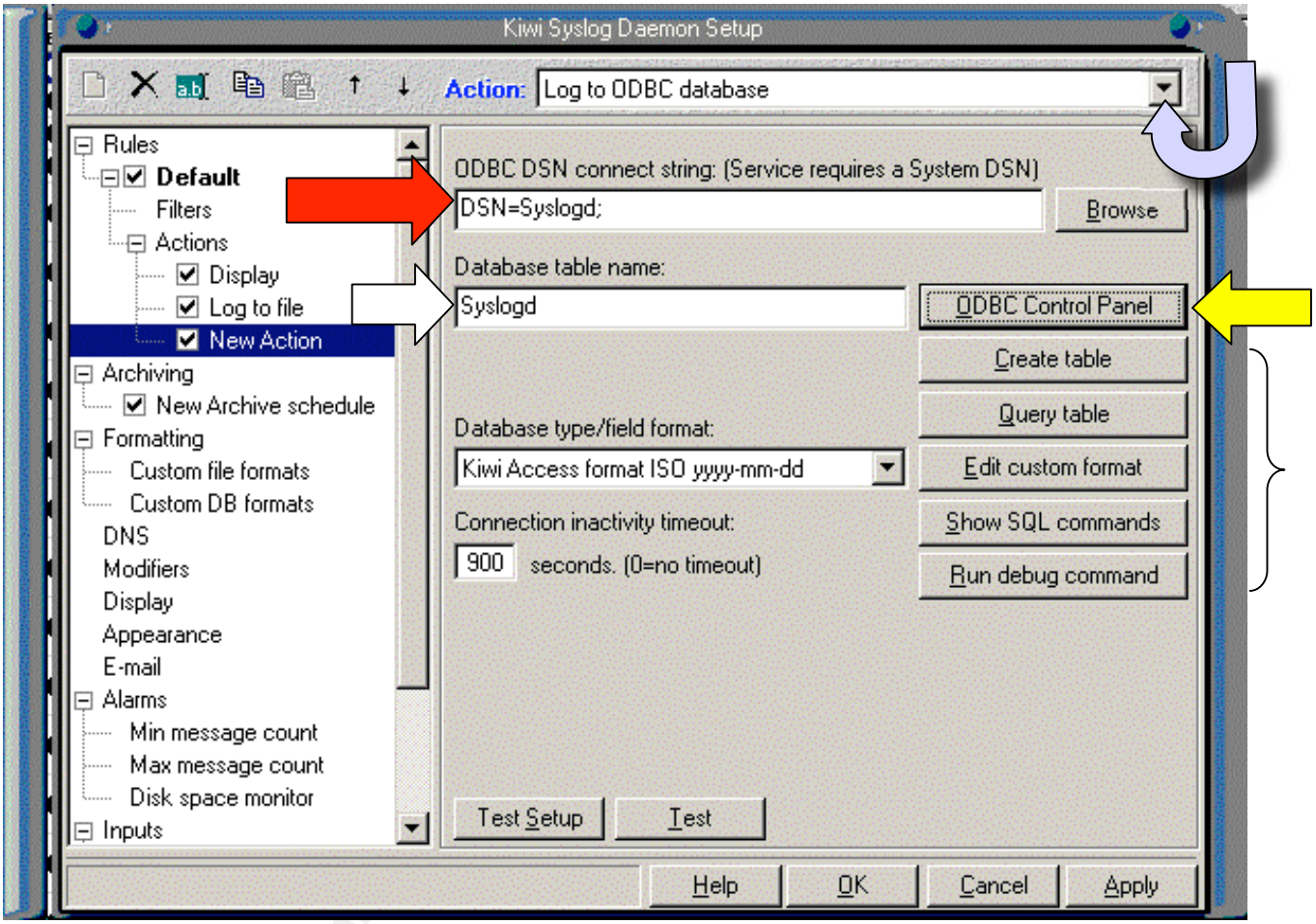


FIG. 1

First you need to open the application and click on "*File*" and then "*Properties*" and it will open the "*Kiwi Syslog Daemon Setup*" window **(FIG. 1)**. From the left panel under "*Action*" right click and select "*Add Action*", in the same left panel it will appear the field "*New Action*". Then at the right panel, as above indicated by the blue arrow, you have to drill down from the list called "*Action*" the setting "*Log to ODBC database*".

At this point we have to set your DSN: Short for ***Data Source Name***. Data Source Name[1] provides connectivity to a database through an ODBC[2] driver. The DSN contains database name, directory, database driver, UserID, password, and other information. Once you have created and configured a DSN (showed by the red arrow above) for your specific database, the Syslogger will be connected to the database and messages are able to be archived in real-time. Here are the step-by-step to do so.

Click on the "ODBC Control Panel" (yellow arrow on **FIG. 1**) and select "System DSN" as shows in **FIG. 2**.

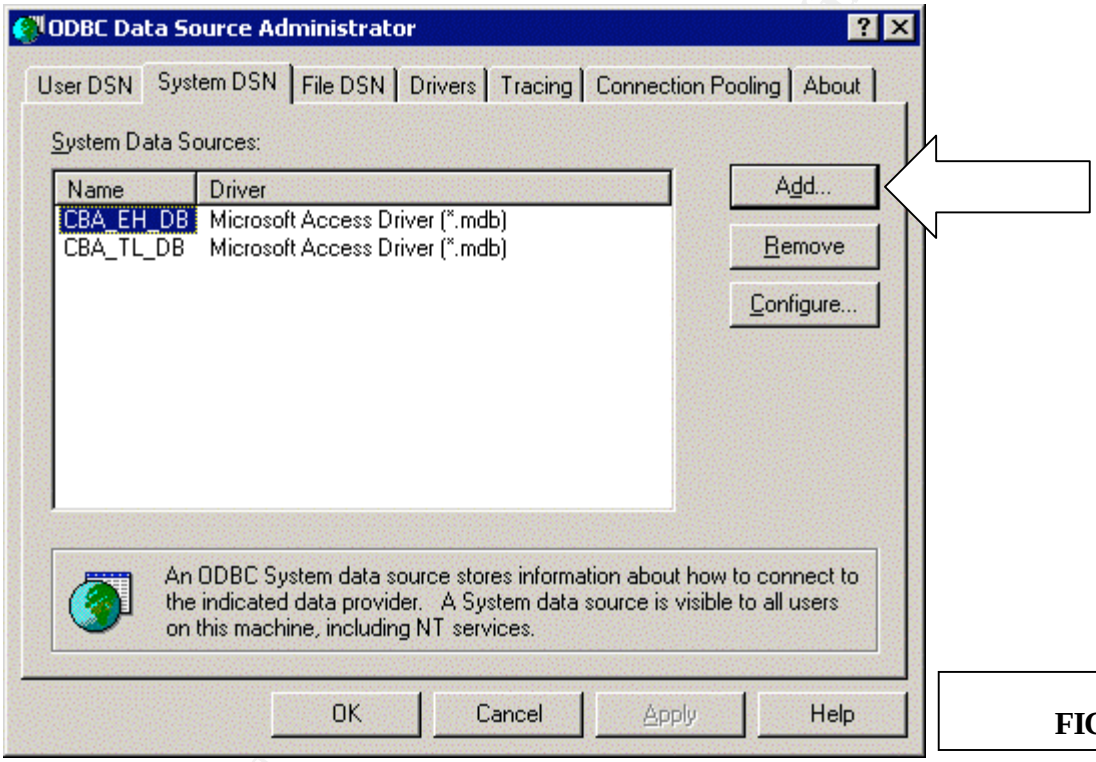


FIG. 2

Click "Add" for the driver corresponding to the type of database you have, as indicated below on **FIG. 3** and **FIG. 4**.

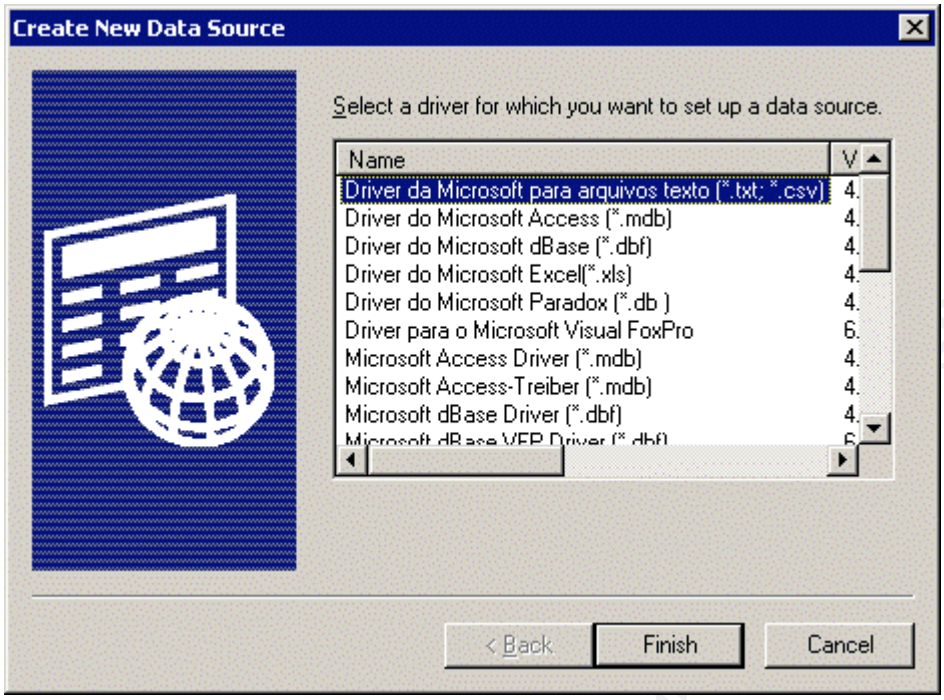


FIG. 3

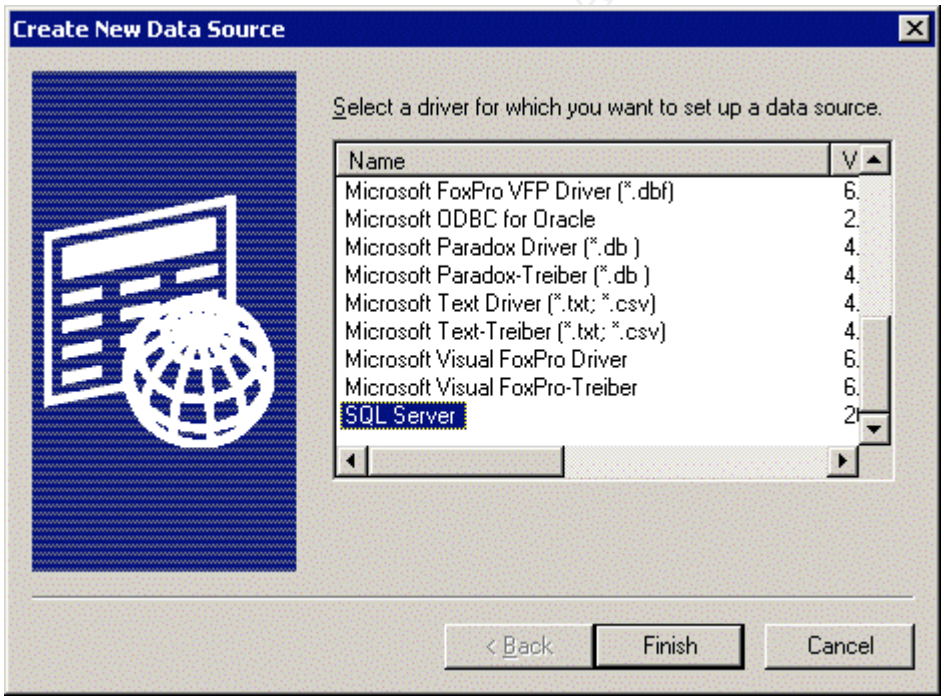


FIG. 4

Then click "*Finish*" and it will open the window (**FIG. 5** below) where you have to type the name

you assign to the data source (I suggest to use your database name for simplicity). Then the description (optional) and last, the server name to identify where the database is installed. For database installed on localhost you can drill down as indicated by the green arrow and you select "local" (if you previously have installed a database on the same system, it will find it automatically). Instead, if you wish to archive logs on a different system, you will type the ip address or a DNS name of the remote server.

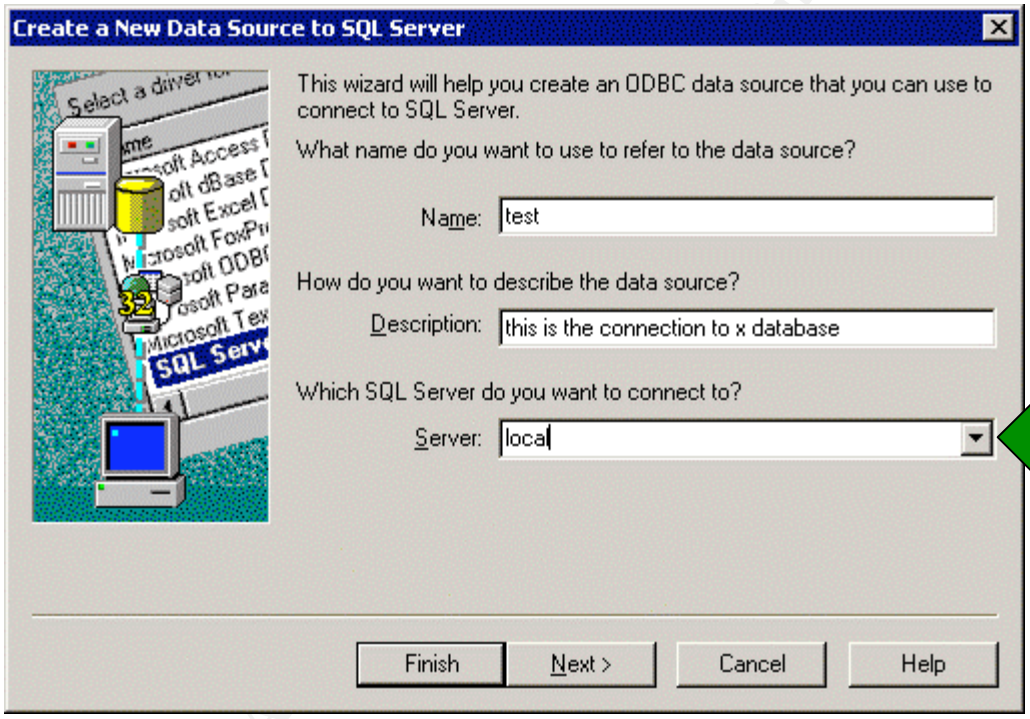


FIG. 5

Then click on "*Next*" and **FIG. 6** window will appear:

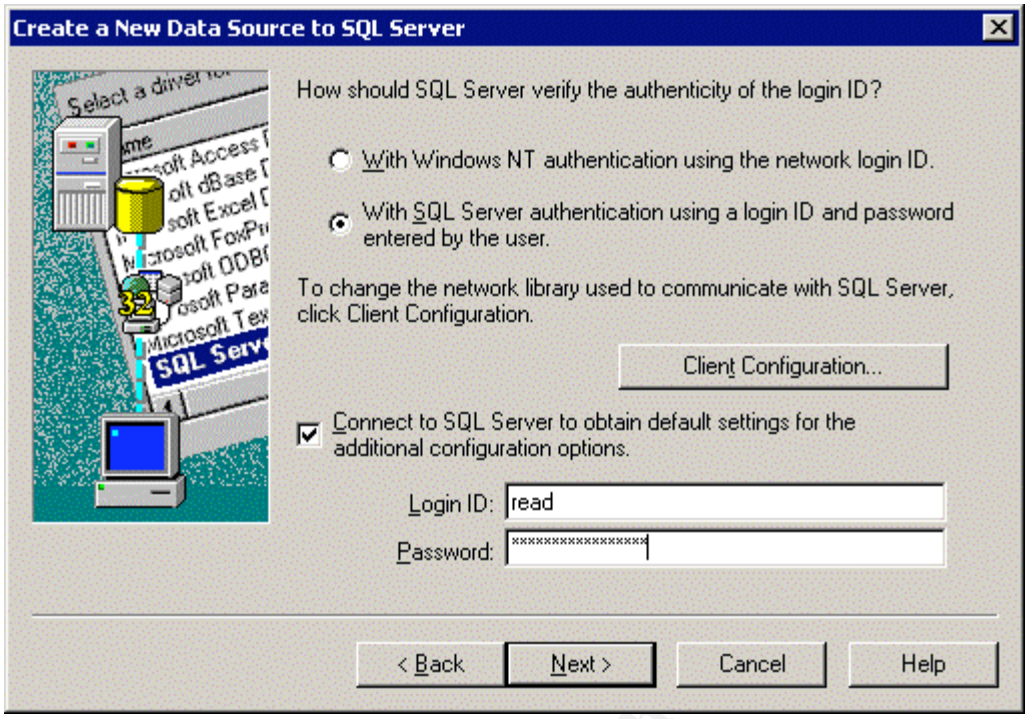


FIG. 6

At this point you can choose what type of authentication you would like to use for the connection. I used and suggest to set an ONLY-READ account on the database (see **FIG. 6**).

Then click on "*Next*" and the next window on **FIG. 7** will show you the default following options that I left unchanged. Notice that the last option "*Use the failover SQL server if the primary SQL server is not available*" is particularly helpful if you plan to have a redundant database.

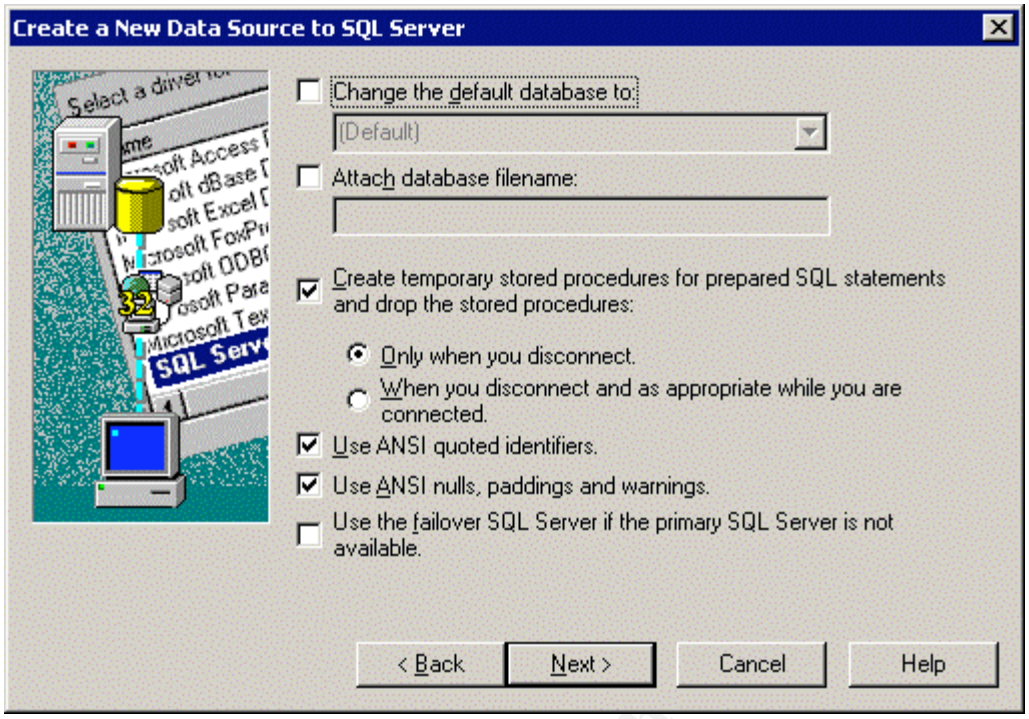


FIG. 7

Now you will click on "*Next*", but before describing the new coming window on **FIG.8**, I have to make an introduction.

In parallel to configure the Syslogger, we need to configure devices we desire to monitor. Each device will need in their logging configuration files our Network Log Analyzer (the system with Kiwi installed) ip address or DNS name. This way each device will send log messages to our centralized Syslogger.

For Linux the configuration file is located in /etc, the file to edit is Syslog.conf, while for Windows I personally use the free [3]"Eventlog to Syslog Utility".

The different architectures and designs of the "devices" imply that each different system has its own way to "package" the log message.

More straightforward: Windows systems send a log messages in a different format than Unix systems, Cisco Routers, or Juniper Routers and so on.

The good thing is that Kiwi beautifully accepts and digests every format/layout of the message.

The tricky part is that you are able to modify in Kiwi the format of the received file before forwarding it to your database using the commands to the right panel as indicated by the right bracket above on **FIG.1**. The way the data will be archived (records) really depends on which brand of database you are using. If a different format (usual is the case) than the default one is required to archive the records, that is a mandatory rule you have to follow to avoid messages are recorded with errors into the database. It will be clearer with the following examples.

I have tested 2 Microsoft products: Access 2003 and SQL 2000 STD edition.

I had to modify the date, time, and part of the description field according to MS access or MS SQL database requirements. Specific features must be modified according to your database's brand. Going back to the Kiwi Syslogger configuration that we have left on **FIG.7** after our last "*Next*".

Below on **FIG.8**, check "*Perform Translation for Character Data*" like shown below (I have cleared out my info):

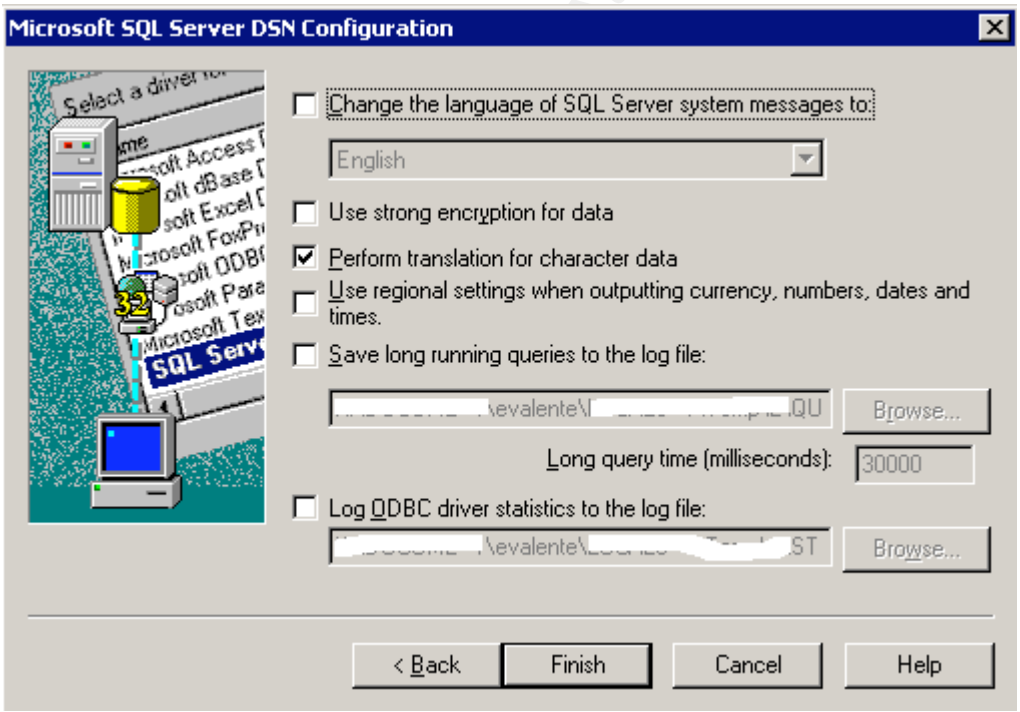


FIG. 8

Then push “Finish” and the following window (**FIG. 9**) will appear:

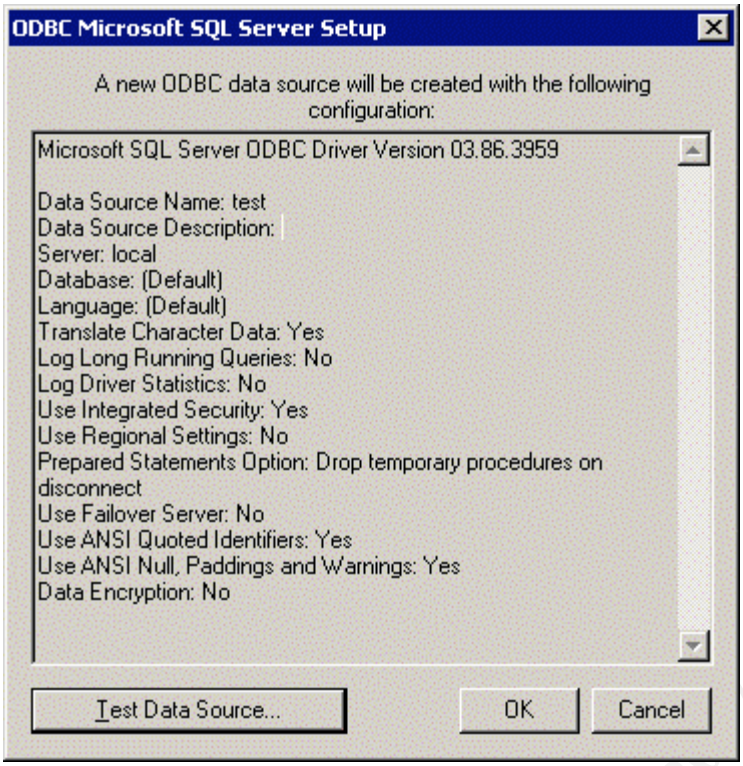


FIG. 9

FIG. 10

Next click on “Test Data Source” and if everything was configured properly you will see the window shown on **FIG.10** (otherwise the test will fail with a detailed error message).

Then click the “OK” button 3 times and you will be back to the Setup page **FIG.1** window. Then you create the table into your database pushing the button indicated by the brown arrow below, **FIG 11.**

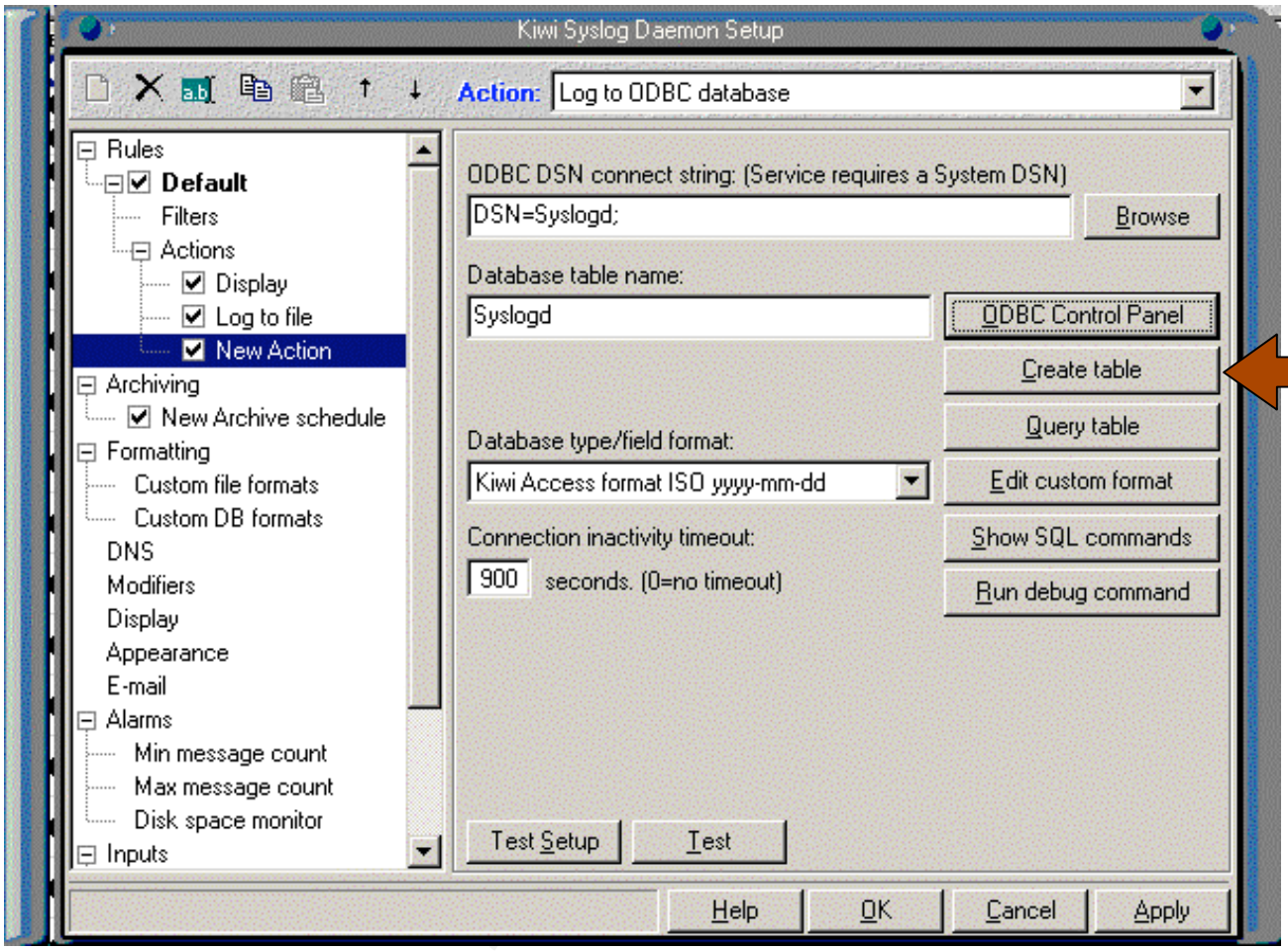


FIG. 11

At this point you are almost done with the tricky part.

Press "Test", in the bottom part, to check your entire configuration (you will get a green check sign ✓ or a red cross ✘).

If you get a green check you are OK and messages are going to be into the database in real-time (**FIG.12**).

If you get a red cross you will be prompted with a detailed error message as shown in **FIG 13**.

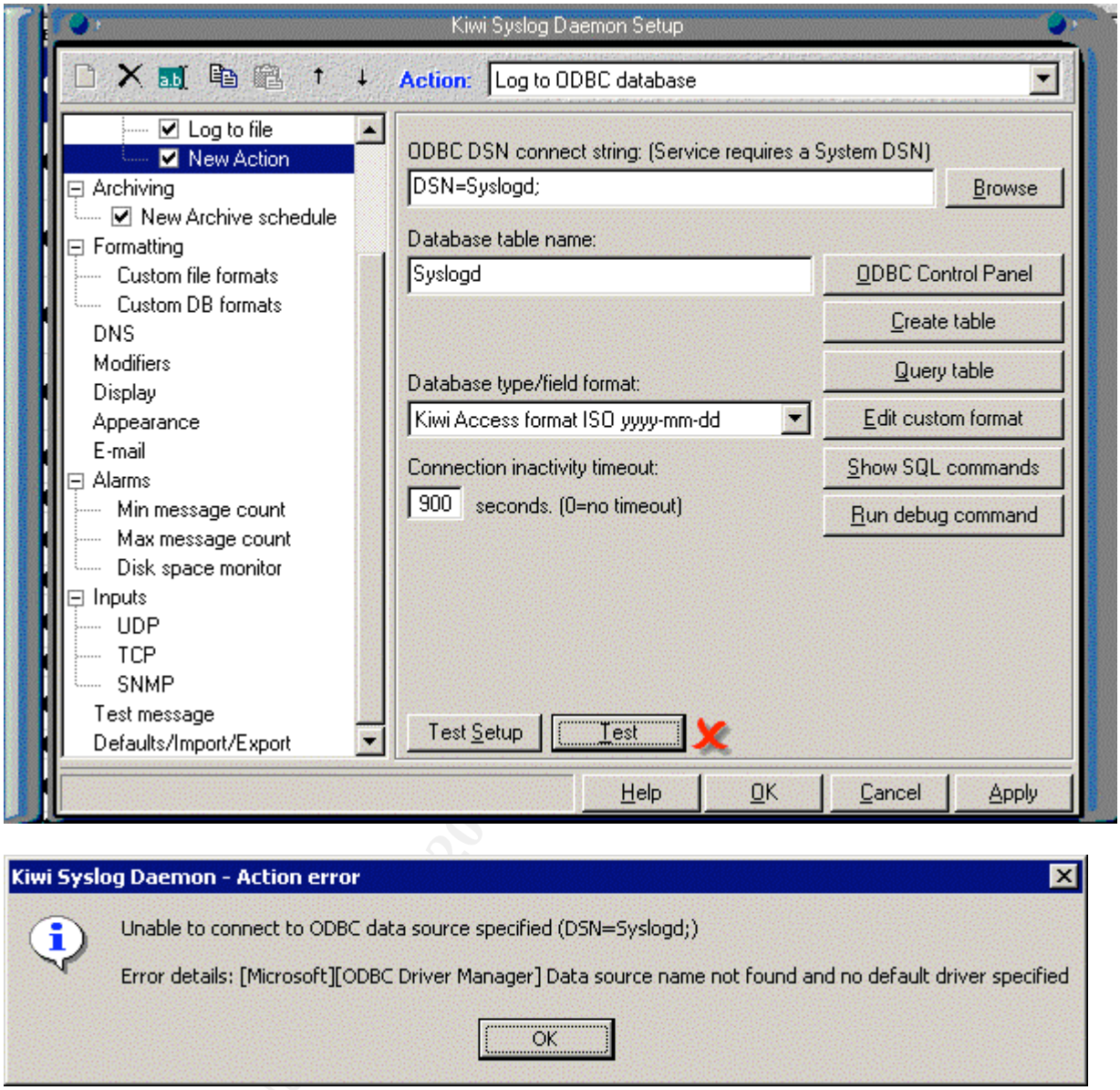


FIG. 13

To debug your error there are also several features on the Setup page that can be explored: "Query Table", "Edit Custom Format", "Show SQL Commands" and "Run Debug Command".

For a complete list of guidelines and instructions of how to configure Kiwi Syslogger I have provided the link to the different versions of their user manuals in the "Reference" page at the end of this paper[4].

After we have successfully sent a message from the device (router, switch or system) and looked at it into our database (depending upon which type you have you should use the specific utility to do so) we can proclaim that our Centralized Syslogger successfully stores data in real-time into the database and that they are available for us to be analyzed by our Network Log Analyzer (which we have not built yet!).

Few advices about the database, I would like to focus on: security, maintenance, and updates.

Database Security:

As I have stated above, it is recommended that the Kiwi Syslogger use ONLY-READ account when logging messages into the database. Please disable the default "public" account. Keep the restrictions on privileges for new database accounts on this database since there are "sensitive" data in it (remember usually hackers delete tracks and logs when leaving the compromised system). For this reason I strongly recommend encrypting the logs on the network using one of the many utilities offered by the vendors. See [5]*Kiwi Secure Tunnel* that does exactly that and it is free.

Database Maintenance:

Besides the usual recommendations about backing-up your database, the fundamental thing to keep in mind about a Syslogging Database is that, no matter how many devices your organization has, to store the enormous amount of data is always a big issue. Logs accumulates in your database faster that you realize and if you don't have a plan in your mind at the beginning of your project, you are jammed.

I personally keep only last 3 months of logs running on the database and I use the neat built-in feature maintenance of the MS SQL, appropriately configured. Every night it reconciles and assesses for consistency the entire database and cuts log messages that are 3 months old. Of course I did backup and store logs older that 3 months.

Database Updates:

We are trying to keep an eye on strange logs and quickly track down compromised hosts, or stolen

data that may damage our business; therefore weekly updating of our database according to the vendor's periodical release, is of paramount importance and necessity.

- **Web Interface:**

Here is where our Network Log Analyzer takes form:

The Web interface (I called "Syslog Manager") is the one I recommend for our Network Log Analyzer because of its flexibility and dynamic outputs.

In fact, I would say that this is the most important part of our architecture since without it our logs analysis would take too long or be impossible. In fact if you have hundreds of devices sending logs, you can see from the console window of the Kiwi Syslogger how fast the messages are logged and you don't even have the time to read a part of them.

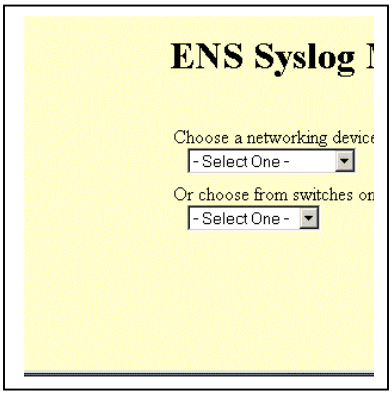
The idea is to build something that allows any user/sysadmin, with a friendly and fast interface, to find efficiently and quickly any small piece of information in the huge amount of logging data.

The webpage is the user interface and can be built using [6]Microsoft *active server pages* (asp), or [7]Hypertext Preprocessor (php) or whichever web language you prefer.

I used php that it seemed to me the easier one and it included some JavaScript for part of the page (the calendar).

You start downloading the latest version of PHP and create two files: upper part (searches criteria entries) of the interface and the bottom part (results of the searches).

The below **FIG. 14** shows in two different colors the 2 parts (files) of the entire page (ENS is the name of the networking group here at SDSC).



After creating the two files you have to start coding the different sections of the page according to the function you would like to be executed. For the date and time I have used a JavaScript calendar (one of the many free source codes) that can easily be found in Internet (little icon to the side of the date). Then you have to program the connection to your database and that depends on the type of database you are using; the syntax varies by vendor (this also can be easily found online).

From the left upper panel (**FIG. 14**) you have to select one networking device (in the future I would like to make it a comparison between 2 or more device message) from the drill down menu (it can be a switch or a router or a server). You also have the choice of selecting only from the switches list of devices or (this is the most useful choice) you can search "All devices" and the search by time and/or keywords will be executed across all the machines. Then the starting date automatically goes to the current date and time while the ending date automatically goes back to the last two hours of activity (very useful feature to quickly check the latest activity of each device).

Then as I mentioned, I created four (4) different keywords that can be used simultaneously for specific searches. The keyword search can be combined with the date search, of course.

The bottom part of the page (**FIG.14**) shows several useful info and results of the searches:

- **Last ID:** generated into the database to track specific logs by their ID, this number is always unique and increases constantly in spite of the database periodical resize

- **Total Number of Records:** the amount of messages stored as records to measure how much data periodically fills the database (you can easily calculate the differences over time and get useful info)

- **"Print Results"** button: The results listed and ordered by date and time (latest on the top). Notice that you can create (programming code) your own fields in the results top row as column record names of the table. I have decided to strip out the unnecessary entries in the raw message field and easily find readable to separate date, time, priority and hostname (opportunistically

resolved with DNS names), but this choice is really up to your preference.

And now below are practical examples of real life utilization, in the "identification" phase of Incident Handling, of the described Network Log Analyzer.

- **Reports:**

The day-to-day activity starts with an analysis of the midnight reports that Kiwi send via email and that can be summarized in two types:

1) Archived Status Report

2) Daily Syslog Statistics

The first notify that the file contained the entire day activity has been successfully archived and also shows other useful info as described below:

```
///                 Archive Status Report                ///
-------------------------------------------------------------
Date and Time:      Fri, 07 Dec 2007 00:00:00

Schedule name:      New Archive schedule
Source Folder:      C:\Program Files\Syslogd\Logs\
Destination Folder: D:\TheFile4Syslogs\

+------------------------------+----------------+------+------+
| File name:                   |  File size     | Move | Zip  |
+------------------------------+----------------+------+------+
| CatchEverything.txt           |  550,751.82 KB |  OK  | N/A  |
+------------------------------+------------+------+------+----

End of report.
```

For the second a detailed analysis is required to understand and investigate possible abnormal activities. In the "Identification" phase of the Incident Handling "Signs of an incident" is the starting point of the investigation. A

precise analysis of logs has to be done before declare that an incident occurred.
Here below are two examples of the info reporting a normal activity **(a)** and one where is reporting an abnormal number of messages for the devices called Brazil **(b):**

a)

```
///       Kiwi Syslog Daemon Statistics         ///
--------------------------------------------------
24 hour period ending on: Wed, 05 Dec 2007 00:00:00 -0800 Syslog Daemon started on:
Tue, 13 Nov 2007 23:35:19
Syslog Daemon uptime:     21 days, 0 hours, 24 minutes
--------------------------------------------------

+ Messages received - Total:          13576787
+ Messages received - Last 24 hours:  57696 Messages received - Since
+ Midnight: 61080
+ Messages received - Last hour:      10590
+ Messages received - This hour:      7531
+ Messages per hour - Average:        2545

+ Messages forwarded:                 0
+ Messages logged to disk:            61080

+ Errors - Logging to disk:           0
+ Errors - Invalid priority tag:      0
+ Errors - No priority tag:           0
+ Errors - Oversize message:          0

+ Disk space remaining on drive C:    9888 MB

--------------------------------------------------


  Breakdown of Syslog messages by sending host
+--------------------+------------+------------+
| Top 20 Hosts       |  Messages  | Percentage |
+--------------------+------------+------------+
| cisco4000          |        220 |      0.36% |
| Brazil             |        664 |      1.09% |
| pssp432            |       7982 |     13.07% |
| bwbnb1             |      15512 |     25.40% |
| buino21            |      18216 |     29.82% |
| bewww5             |      18486 |     30.26% |
+--------------------+------------+------------+


    Breakdown of Syslog messages by severity
+--------------------+------------+------------+
| Message Level      |  Messages  | Percentage |
+--------------------+------------+------------+
| 0 - Emerg          |          0 |      0.00% |
```

```
| 1 - Alert          |      26803 |     43.88% |
| 2 - Critical       |          0 |      0.00% |
| 3 - Error          |          0 |      0.00% |
| 4 - Warning        |          0 |      0.00% |
| 5 - Notice         |      33557 |     54.94% |
| 6 - Info           |        720 |      1.18% |
| 7 - Debug          |          0 |      0.00% |
+--------------------+------------+------------+


End of Report.
```

b)

```
///       Kiwi Syslog Daemon Statistics         ///
-----------------------------------------------------
24 hour period ending on: Fri, 07 Dec 2007 00:00:00 -0800 Syslog Daemon started on:
Tue, 13 Nov 2007 23:35:19
Syslog Daemon uptime:     23 days, 0 hours, 24 minutes
-----------------------------------------------------

+ Messages received - Total:          13705656
+ Messages received - Last 24 hours:  59691 Messages received - Since
+ Midnight: 64055
+ Messages received - Last hour:      10382
+ Messages received - This hour:      7533
+ Messages per hour - Average:        2668

+ Messages forwarded:                 0
+ Messages logged to disk:            64055

+ Errors - Logging to disk:           0
+ Errors - Invalid priority tag:      0
+ Errors - No priority tag:           0
+ Errors - Oversize message:          0

+ Disk space remaining on drive C:    9233 MB

-----------------------------------------------------

  Breakdown of Syslog messages by sending host
+--------------------+------------+------------+
| Top 20 Hosts       |  Messages  | Percentage |
+--------------------+------------+------------+
| cisco4000          |        232 |      0.36% |
| Brazil             |       2086 |      3.26% |
| pssp432            |       8095 |     12.64% |
| bwbnb1             |      16517 |     25.78% |
| buino21            |      18617 |     29.06% |
| bewww5             |      18508 |     28.90% |
+--------------------+------------+------------+
```

Breakdown of Syslog messages by severity

Message Level	Messages	Percentage
0 - Emerg	0	0.00%
1 - Alert	26804	41.84%
2 - Critical	0	0.00%
3 - Error	0	0.00%
4 - Warning	0	0.00%
5 - Notice	34404	53.71%
6 - Info	723	1.13%
7 - Debug	2124	3.32%

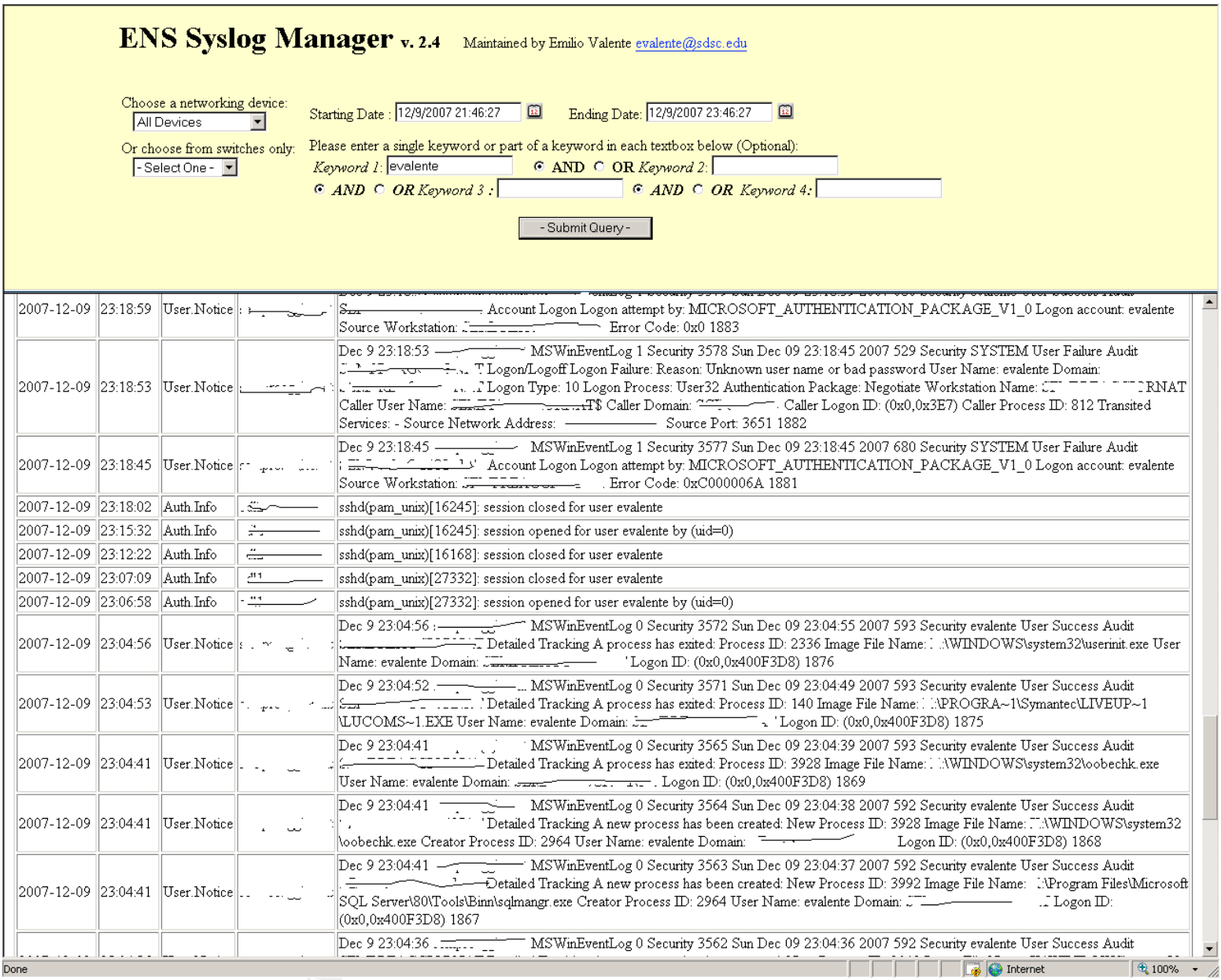
End of Report.

3. Case study

In the report **b)** we can notice that the hp server Brazil that is a Windows OS is reporting more that 2000 log messages while usually it only reports a few hundreds a day and also that overall all the devices have an abnormal raising in the total number of logs compared to the baseline.

From here the next step is to use the web interface to quickly find out what is the issue or if indeed an incident occurred. I added a feature in the newer revision 2.4 on the web interface that allows me to search activity occurred across all devices for the last 2 hours or more by keywords (see below fig.15 sensitive data have been cleared out) and correlate same events that occur in more than one device.

FIG. 15



You can see that I have been working and logged on several devices (Windows and Linux systems) in one click is possible to narrow down one common event that appears in several devices without go research through the mountain of Syslog messages of the individual device.

In particular for the case study, the hp device we have seen above, as soon as I searched for the last 6 hours of messages in that particular day into the database I was able to identify account name

"SColbert" logged in successfully through ssh on system "Jerome" at 4:07am.

Now since this account has been disabled because Colbert is an employee that is currently in a leave of absence, something was wrong and we are in front of an incident.

With few clicks and without log in each of the hundreds of devices I was able to start right the way the initial Incident Handing procedure at least not spending time to logon into a different system to identify the initial status. I was also able to look with the "keywords" search across all the devices logs and discover when and where account SColbert have been visited any other systems and "successfully" or "failing" attempts.

The result is where and when the intruder as been tried to logon and in which system he was actually able to get in and report the findings in the Incident Handling documentation.

Time is everything: the powerful correlation that with the use of this tool can be done is an immensely advantage in terms of time and precision that can be invaluable for any sysadmin at the identification phase of an incident handling procedure.

I will be happy to answer any questions and provide support for anybody that is willing to adapt a similar solution. My contact info is:

Emilio Valente
Phone: 858-822-0928
evalente@sdsc.edu

4.References

[1] http://www.webopedia.com/TERM/D/DSN.html (2007) Internet

[2] http://www.webopedia.com/TERM/D/ODBC.html (2007) Internet

[3] https://engineering.purdue.edu/ECN/Resources/Documents/UNIX/evtsys/

[4] http://www.kiwisyslog.com/support/ (2007) Internet

[5] http://www.kiwisyslog.com/kiwi-secure-tunnel-overview/ (2007) Internet

[6] http://en.wikipedia.org/wiki/Active_Server_Pages (2007) Internet

[7] http://en.wikipedia.org/wiki/PHP (2007) Internet

Last Updated: August 15th, 2020

Upcoming SANS Training

Click here to view a list of all SANS Courses

SANS FOR508 Sydney August 2020	Sydney, AU	Aug 17, 2020 - Aug 22, 2020	Live Event
SANS Virginia Beach 2020	Virginia Beach, VAUS	Aug 30, 2020 - Sep 04, 2020	Live Event
SANS London September 2020	London, GB	Sep 07, 2020 - Sep 12, 2020	Live Event
SANS Baltimore Fall 2020	Baltimore, MDUS	Sep 08, 2020 - Sep 13, 2020	Live Event
SANS Munich September 2020	Munich, DE	Sep 14, 2020 - Sep 19, 2020	Live Event
SANS Australia Spring 2020	, AU	Sep 21, 2020 - Oct 03, 2020	Live Event
SANS San Antonio Fall 2020	San Antonio, TXUS	Sep 28, 2020 - Oct 03, 2020	Live Event
SANS Northern VA - Reston Fall 2020	Reston, VAUS	Sep 28, 2020 - Oct 03, 2020	Live Event
SANS FOR500 Milan 2020 (In Italian)	Milan, IT	Oct 05, 2020 - Oct 10, 2020	Live Event
SANS Amsterdam October 2020	Amsterdam, NL	Oct 05, 2020 - Oct 10, 2020	Live Event
SANS Brussels October 2020	Brussels, BE	Oct 05, 2020 - Oct 10, 2020	Live Event
SANS Prague October 2020	Prague, CZ	Oct 12, 2020 - Oct 17, 2020	Live Event
SANS London October 2020	London, GB	Oct 12, 2020 - Oct 17, 2020	Live Event
SANS Orlando 2020	Orlando, FLUS	Oct 12, 2020 - Oct 17, 2020	Live Event
SANS October Singapore 2020	Singapore, SG	Oct 12, 2020 - Oct 24, 2020	Live Event
SANS Stockholm October 2020	Stockholm, SE	Oct 19, 2020 - Oct 24, 2020	Live Event
SANS Dallas Fall 2020	Dallas, TXUS	Oct 19, 2020 - Oct 24, 2020	Live Event
SANS Rome October 2020	Rome, IT	Oct 19, 2020 - Oct 24, 2020	Live Event
SANS SEC504 Rennes 2020 (In French)	Rennes, FR	Oct 19, 2020 - Oct 24, 2020	Live Event
SANS Cologne October 2020	Cologne, DE	Oct 26, 2020 - Oct 31, 2020	Live Event
SANS San Francisco Fall 2020	San Francisco, CAUS	Oct 26, 2020 - Oct 31, 2020	Live Event
SANS Geneva October 2020	Geneva, CH	Oct 26, 2020 - Oct 31, 2020	Live Event
SANS SEC560 Lille 2020 (In French)	Lille, FR	Oct 26, 2020 - Oct 31, 2020	Live Event
SANS Tel Aviv November 2020	Tel Aviv, IL	Nov 01, 2020 - Nov 05, 2020	Live Event
SANS London November 2020	London, GB	Nov 02, 2020 - Nov 07, 2020	Live Event
SANS Rocky Mountain Fall 2020	Denver, COUS	Nov 02, 2020 - Nov 07, 2020	Live Event
SANS DFIRCON 2020	Miami, FLUS	Nov 02, 2020 - Nov 07, 2020	Live Event
SANS Sydney 2020	Sydney, AU	Nov 02, 2020 - Nov 14, 2020	Live Event
SANS Krakow November 2020	Krakow, PL	Nov 02, 2020 - Nov 07, 2020	Live Event
SANS Paris November 2020	Paris, FR	Nov 02, 2020 - Nov 07, 2020	Live Event
APAC ICS Summit & Training 2020	Singapore, SG	Nov 13, 2020 - Nov 21, 2020	Live Event
SANS OnDemand	OnlineUS	Anytime	Self Paced
SANS SelfStudy	Books & MP3s OnlyUS	Anytime	Self Paced