The Lire Roadmap

July 2002

Joost van Baal

Francis J. Lacoste

Introduction

This document gives a roadmap for the development of the Lire software. It serves as a reference point when working on the software and states the current ideas and plans of the LogReport developers.

The roadmap contains three sections, in the first one we list the items that are related to the tool suite. The second part contains items that are related to the Lire framework and which relates to more infrastructural aspect. Finally, the last section contains long-term project that will implies deep infrastructural changes.

For each items, we try to list the person name and email responsible for the feature. This will usually be a person who can be joined on the <development@logreport.org> mailing list. You should also find the current status of the item and the items that need to be implemented before.

If you want to work on the item described in this roadmap, send an email to the <development@logreport.org> mailing list.

> **Important:** One should also consult the BUGS file in CVS which contains many small bugs and wishlist items.

End-User's Items

FreeBSD Binary Package

Status: Started
Responsible: Cédric Gross, Edwin Groothuis
Prerequisites: none

The FreeBSD package should get distributed in the offical FreeBSD ports collection. The package built by Cédric Gross is in FreeBSD's port queue since Tue Mar 05 12:20:01 PST 2002 (see http://www.freebsd.org/cgi/query-pr.cgi?pr=ports/35566). Edwin Groothuis has build another package, based on Lire 1.0. See http://www.FreeBSD.org/cgi/query-pr.cgi?pr=ports/39871. Since Sun Jul 7

20:55:25 PDT 2002 adrian at FreeBSD.org is dealing with this. (Mailing to freebsd-ports@freebsd.org and politely requesting some attention *might* speed up things.)

Improved Merging Interface

Status: Not started
Responsible: None
Prerequisites: the Section called *Configuration API*, the Section called *Storage API*

Altough the current distribution supports merging of reports, it should be better integrated in the **lr_cron** and **lr_config** interfaces.

Improved Reports

Status: Not started
Responsible: None
Prerequisites: the Section called *Enriched XML report format*

User configuration hooks should be added to tweak various display settings, the overall sexyness of the output should be improved. Especially the HTML layout should get improved.

Furthermore, the formatting backend should support all the XML reports we can generate. (E.g. a multiple column layout.) We currently don't yet use this functionality in any of our reports.

Improved Charts Generation Tool

Status: Not started
Responsible: None
Prerequisites: None

The current GD::Graph based implementation of the charts generation is really suboptimal. This should be better implemented by using a real charting framework like gnuplot or plotutils.

Messagestore Superservice

Status: Conceptual
Responsible: Arnaud Taddei, Cédric Gross
Prerequisites: None

The CVS now contains a DLF schema for a messagestore superservice which was contributed by Cédric Gross. This superservice should report on POP and IMAP servers as well as message-store multiplexers. Arnaud Taddei also express interests in working on that superservice.

Directory Superservice

Status: Not started
Responsible: Arnaud Taddei
Prerequisites: None

Arnaud Taddei also express interests in developping a directory superservice for LDAP servers.

More firewall services

Status: Not started
Responsible: None
Prerequisites: None

Users have requested ‘Firewall-1’, ‘snort’ and ‘Watchguard Soho’ firewall DLF converters.

Overhaul of the Email Superservice

Status: Conceptual
Responsible: None
Prerequisites: None

The current DLF schema of the email superservice is starting to show its limits. A lot of information is lost in the email log files and several important report (like refused connections, spam control, etc.) cannot be generated. The schema should be rewritten to closer resemble the log that one actually see with different fields for anti-spam activity, message collection, routing, etc. This will make writing email DLF converter a lot easier. The current DLF schema could become a derived schema and the stateful logic that is replicated across all email service could be moved to a generic analyser.

Featureful Online Responder

Status: Conceptual
Responsible: None
Prerequisites: the Section called *Configuration API*, the Section called *Storage API*

The online responder we offer on our website should be able to support all the features the command line Lire supports. The HTTP upload interface should be completed. Installation of a responder should be better documented and easier. There was an initial proposal sent to the <development@logreport.org> mailing list in the message PROPOSAL: New Online Responder Architecture (mid:<20020421210816.GY12459@Contre.COM>)..

Configuration GUI

Status: Not started

Responsible: None
Prerequisites: the Section called *Configuration API*

There should be a better configuration interface than the lr_config script we offer now. The CGI interface should get completed. A GUI interface should get added.

Some research on GUI libraries has been done by Plamen Bozukov in September 2001 (Message-ID: <Pine.LNX.4.10.10109271704180.25208-200000@pozvanete.bg>). He came to the following conclusion:

Table 1. Comparison of GUI libraries

score 1-5	GPL	portability	requirements	easy	features	binding
Qt	4	5	5	5	5	3
V	5	5	5	4	4	2
FLTK	5	5	5	4	3	5
GTK	5	4	5	5	4	4
WxWindows	5	5	4	5	5	5
Tk	5	5	5	3	4	5

QT

License: QPL/GPL. For windows version, license is possibly problematic. Portability: Microsoft Windows 95/98/2000, Microsoft Windows NT, MacOS X, Linux, Solaris, HP-UX, Tru64 (Digital UNIX), Irix, FreeBSD, BSD/OS, SCO and AIX. Requirements: C++ Compiler X libraries for Unix. Binding: Perl-binding is very old - Last updated November 17th, 1997. Python and Ruby.

V

License: GNU LGPL. Portability: Windows,OS/2,Unix.

FLTK

License: LGPL. Portability: Unix,Windows,OS/2. Requirements: C++ Compiler X libraries for Unix. Bindings: Perl: 2 different solutions; Python

GTK

License: GNU LGPL. Portability: Unix, Windows. Requirements: C Compiler X libraries for Unix. Bindings: all possible languages. There is the glade interface builder which makes it easy to design GTK interface.

WxWindows

License: GNU Library General Public. Portability: Unix,Windows,Mac. Requirements: C++ Compiler GTK libraries for Unix. Bindings: Perl, good binding for Python.

Tk

Portability: Windows, Unix, Mac. Bindings: all possible scripting languages.

More information on various GUI toolkits is on The GUI Toolkit, Framework Page (http://www.free-soft.org/guitool/).

Support reports in multiple languages.

Status: Not started
Responsible: None
Prerequisites: the Section called *Internationalisation Framework*

Lire should support other languages in its error messages and other output, as well as in the report specifications. This is a long-term task.

SQL Based Backend

Status: Not started
Responsible: None
Prerequisites: the Section called *Separation of the Analysis Process*

An important target for Lire is to developped a SQL based reporting engine which should offer more scalable reporting and probably make it worthwhile to use the framework in an interactive development.

Framework Items

Configuration API

Status: Design
Responsible: <flacoste@logreport.org>
Prerequisites: None

Lire's framework should contain a configuration API that should be used by all of its components. See the message PROPOSAL: Lire Configuration Framework (mid:<20020430172451.GJ18114@Contre.COM>) to the <development@logreport.org> mailing list for more details.

Storage API

Status: Not started
Responsible: None
Prerequisites: the Section called *Configuration API*

Lire should offer an API to a persistent store which could be used by all components to store and retrieve parts of data.

The current archive implementation is used for two purposes. First, it is used for inter-components communication during one jobs life-time and it's also used for long-term storage. The first part should be moved to a control file mechanism like the one described in the message PROPOSAL: New Online Responder Architecture (mid:<20020421210816.GY12459@Contre.COM>) to the <development@logreport.org> mailing list. The second functionality is the proper domain of the persistent store API.

The persistent store should also be arranged around the user/server hierarchy described in the configuration framework proposal.

Enriched XML report format

Status: Not started
Responsible: None
Prerequisites: None

In order to be able to produce reports which are more meaningful for the reader, the information contained in the XML reports produced by Lire should be modified to contain other information like labels, ratio information, overall statistics, etc.

Separation of the Analysis Process

Status: Not started
Responsible: None
Prerequisites: None

Currently, the analysers that produce the derived schema and extended schema DLF data work in the report generation process. Those should be moved into a separate process between the log normalisation process and the report generation process. This would permit some optimisations in the data format and is necessary to support other report generation backends.

Internationalisation Framework

Status: Not started
Responsible: None
Prerequisites: None

Standard internationalisation components like xml-i18n-tools or gettext should be integrated into the framework. The XML components should also be modified to support other charsets than basic ASCII.

Dynamic Registration of Components

Status: Not started
Responsible: None

Prerequisites: None

Altough the Lire framework is meant to be extended and there is already several APIs provided to do so, most components are still statically registered. For example, each new superservice and service must be registered in several static lists. Same things for the output formats. It would be better if those lists could be built dynamically and if we would provide simple tools to register new components (something like a **lire-install** command).

C Based Implementation of the APIs

Status: Not started
Responsible: None
Prerequisites: None

To make the Lire framework usable in more context, it might be worthwile to reimplement the APIs in C so that mapping for other languages than perl could be made available. This would also make it possible to support lex/yacc based DLF converters. It could also help performance.

Long-Term Items

Schemas with Multiple Event Types Support

Status: Not started
Responsible: None
Prerequisites: Unknown

Currently, when a DLF schema represents different kind of events, it is done by using a subset of all the fields present in the schema. An example of this can be found in the firewall superservice which provides support for IDS, packet accounting and packet filtering kind of events.

A limitations of the current approach is that the fields required for a particular events can't be specified. Another limitation is that similar events across superservice cannot be identified.

A proposed solution can be found in the document (mid:<3C9E3B2D.B09B4EF9@sun.com>) sent by Arnaud Taddei to the development mailing list.

Cross-superservice reporting

Status: Not started
Responsible: None
Prerequisites: Unknown, probably the Section called *Schemas with Multiple Event Types Support*

One things often requested by users is the ability to generate reports across superservices.

One possible solution which would use normalized events schema across superservice is explained in the document (mid:<3C9E3B2D.B09B4EF9@sun.com>) sent by Arnaud Taddei to the development mailing list.