Libprelude - Bug #4

Making the new IDMEF API more consistant: refcount

05/22/2004 09:48 PM - Yoann VANDOORSELAERE

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Yoann VANDOORSELAERE	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
Resolution:	fixed		

Description

Currently, it is said that an object inserted into the IDMEF tree should not be touched by the caller anymore. This lead to problem in very common and simple case where it's more natural to code thing that way :

```
<pseudo code>
idmef_time_t *time;

time = idmef_time_new_gettimeofday();
if ( ! time ) { whatever }

idmef_alert_set_create_time(alert, time);
idmef_alert_set_detect_time(alert, time);
</pseudo code>
```

This code look like valid for most developers, however, it will lead to a crash on idmef_message_destroy() because we lack handling refcount support on specific object operation.

The proposal is as follow:

- idmef_xxx_t *idmef_xxx_new():
On standalone object creation, the object refcount should
be set to 0.

- idmef_xxx_t *idmef_yyy_new_xxx():
On creation of a children object, linked to a parent
object, the refcount should be set to 1.

- idmef_yyy_set_xxx(idmef_yyy_t *yyy, idmef_xxx_t *xxx)
When linking a standalone object to it's parent, it's refcount should be incremented.
incremented.

History

#1 - 11/08/2004 12:15 PM - Yoann VANDOORSELAERE

- Status changed from New to Closed

- Resolution set to fixed

Now that every IDMEF object got a refcount, this problem has been solved. Agreement has been reached to keep the way IDMEF object memory is handled explicit to the user, using the ref/unref mechanism.

#2 - 04/29/2009 12:26 PM - Yoann VANDOORSELAERE

- Project changed from PRELUDE SIEM to Libprelude

- Category deleted (1)