

Seagull – Open Source tool for IMS testing



Introduction.....	2
Seagull.....	2
What is Seagull.....	2
Seagull principles and features.....	2
Dictionary.....	2
Scenario.....	3
Feature list.....	3
Why Open Source.....	5
Sponsors.....	5
Seagull and IMS benchmarking.....	6
Conclusion.....	6
For more information.....	7

Introduction

Many of the promises of IMS [http://en.wikipedia.org/wiki/IP_Multimedia_Subsystem] rely on in-depth testing of what is delivered to the end-user. While equipment vendors and network operators are focusing on developing IMS platforms and services, testing of those solutions become a real challenge. The base protocols used for IMS are rapidly changing and are extremely opened by nature (messages and fields can be customized at will), making it really difficult to test on time and on budget. In addition, legacy (SS7) protocols must not be forgotten.

To address this need, test tools used for IMS testing must be powerful, accessible, affordable and extremely agile.

With ease of testing for the IMS chain (network equipment manufacturers, network providers and service providers) comes better services for the end users, quicker deployment with higher confidence and thus greater revenue streams.

Here comes Seagull.

Seagull

What is Seagull

Seagull is a free, Open Source (GPL v2 license) multi-protocol traffic generator test tool released by HP. Primarily aimed at IMS (3GPP, TISPA, CableLabs) protocols (and thus being the perfect complement to SIPP [<http://sipp.sourceforge.net/>], another tool provided by HP for SIP testing), Seagull is a powerful traffic generator for functional, load, endurance, stress and performance / benchmark tests for almost any kind of protocol.

Released in June 2006 on the Sourceforge [sourceforge.net] portal, the first feedbacks were very positive and have even triggered an entry in a blog entitled "Another good day for Open Source and IMS [http://ivelinivanov.blogspot.com/2006_06_04_archive.html]".

Seagull principles and features

Seagull relies on a few strong principles, amongst which the concept of "dictionary" is probably the strongest.

Dictionary

As indicated earlier, protocols are changing quickly and custom messages and parameters must be taken into account too so that any IMS application can be easily tested in a timely fashion.

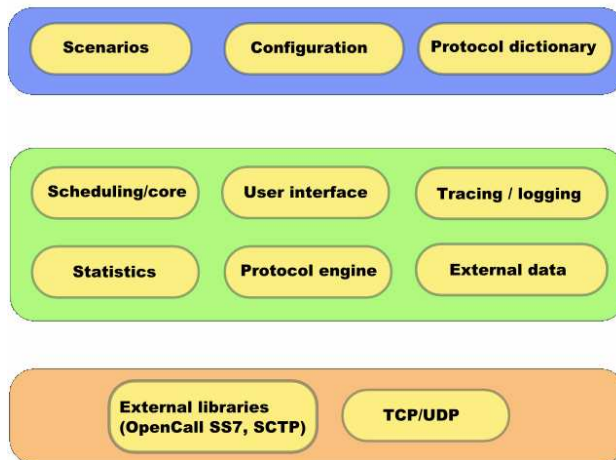
This requirement is fully addressed in Seagull whose openness allows not only to create custom messages and parameters in a snap, but also to add the **support of a brand new protocol in less than 2 hours** - with no programming knowledge. For that, Seagull comes with several protocol families embedded in the C++ source code:

- Binary/TLV (Diameter, Radius and many 3GPP and IETF protocols)
- External library (TCAP, SCTP, SSL)
- Text (XCAP, HTTP, SIP, H248 ASCII)

Protocols are then implemented on top of those protocol families using user editable XML

dictionaries. Those dictionaries describe how messages and parameters are encoded, allowing a great flexibility.

Figure 1. Seagull functional blocks



Scenario

A Seagull scenario - written in XML - describes the messages that are sent and received. It also indicates the behavior to adopt in case a message is unexpected or a check on a parameter fails.

Feature list

Without going into many details, here is a list of features that Seagull currently supports:

- Multi-protocol traffic generator
- Command line tool with text interface

Figure 2. Seagull interface

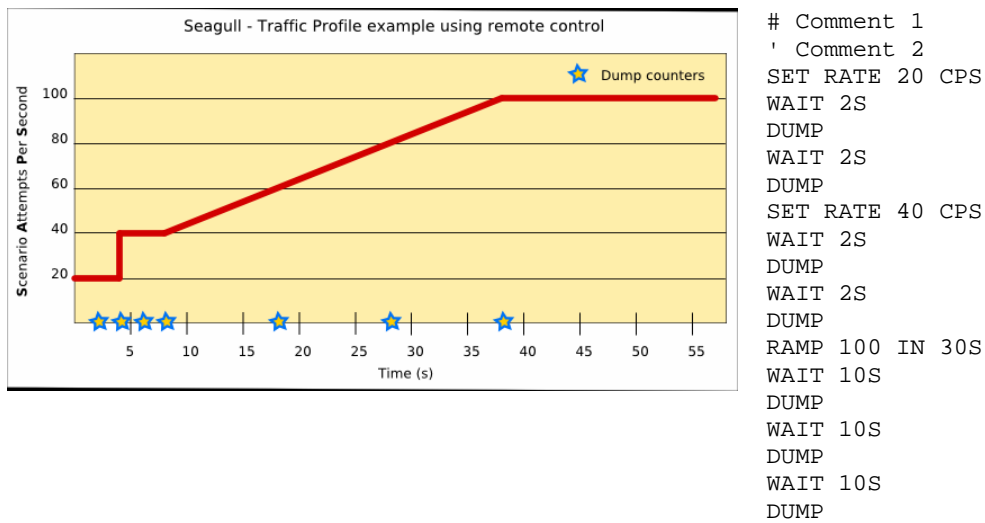
```

root@voip1:/home/ocadmin/diameter/run
-----
Start/Current Time | 2006-05-22 17:41:28 | 2006-05-22 17:41:46
-----
Counter Name | Periodic value | Cumulative value
-----
Elapsed Time | 00:00:01:009 | 00:00:17:162
Call rate (/s) | 1962,339 | 510,663
-----
Incoming calls | 0 | 0
Outgoing calls | 1980 | 8764
Msg Recv/s | 1962,339 | 510,663
Msg Sent/s | 1962,339 | 510,663
Unexpected msg | 0 | 0
Current calls | 0 | 0,000
-----
Successful calls | 1980 | 8764
Failed calls | 0 | 0
Refused calls | 0 | 0
Aborted calls | 0 | 0
Timeout calls | 0 | 0
-----
Last Info | Outgoing traffic
Last Error | No error
-----
Next screen : Press key 1 ----- [h]: Display help -----
  
```

- Protocols of the same family are described in an XML, user editable, dictionary (messages, parameters)

- Existing protocol families: Binary/TLV (Type, Length, Value), Raw binary, Text, external API (first implementation: HP OpenCall SS7)
- Support of IP (UDP/TCP), SCTP, SSL/TLS and SS7/TCAP transports
- Portable programming (tested and supported on Linux x86, ia64, HPUX, SunOS and Windows)
- Scenarios are described using XML files
- Multi-threaded for performances and reliability
- Dynamically adjustable scenario rate
- Uniform, Poisson or Best-effort scenario arrival distribution
- Remote-control (scenario-rate set, counter dump) through standard HTTP interface
- Pause and restart of traffic
- Support of automated traffic profile (varying scenario rate)

Figure 2. Example of Seagull remote control for traffic profile



- Smooth (no new scenarios then wait for ongoing scenarios to end) or brutal end
- Scenario display with message counters
- Scenarios have init (executed once), main (repeated for traffic) sections
- Scenarios have default sections for defense in case of unexpected messages
- A scenario can be mono (most cases) or multi-protocol
- Message and parameters checking possible (disabled by default)
- Support of parameter injection following a CSV like database

Figure 3. Scenario example with data injection from CSV database



- Multiple Seagull instances can be synchronized in the middle of scenario
- Intra scenario synchronization using a synchronization protocol (example application provided in Java language)
- Statistics: timer between two messages, scenario length, scenario rate, successful scenarios, failed scenarios (with reason)
- Protocol decoding and hexadecimal dump
- Trace files with or without timestamps (for performances and automation)

Why Open Source

HP is a strong supporter of Open Source [<http://opensource.hp.com/>] and, as such, embraces the Open Source model.

More specifically for Seagull, releasing in Open Source opens great opportunities:

- **For network equipment manufacturers:** with Seagull's flexibility and amazing price tag (!), testing IMS enabled equipments has never been easier. In addition, the Open Source model reduces the dependence on third party vendors' roadmaps. A feature is missing? Everything needed is available to do it yourself or ask a third party to do it for you.
- **For network providers:** in addition to the same benefits as for network equipment providers, network providers gain the truth on testing. Every test result can be reproduced and proven, simply because Seagull is so easily available. In addition, setting standard test tools allows comparing implementations and performance results.
- **For service providers:** with an Open Source test tool, even the smallest service provider, with the smallest budget, can test its service at a professional level. Yet another way to foster huge IMS service offering.
- **For end-users:** each and every end-user benefits from the entire testing chain that is enabled by Seagull. More interoperability, reliability and performances.

Sponsors

Seagull is sponsored by HP OpenCall Software [<http://www.hp.com/go/opencall/>]. In addition:

- Atos Origin [<http://www.marben-products.com/Diameter/overview.html>] provides support on Seagull and expert consultancy upon request.
- The COMET consortium [<https://www.comet-consortium.org/>] has participated for the support of XCAP and SIP protocols.
- Several users and companies have contributed features and bug fixes for Seagull

Seagull and IMS benchmarking

ETSI [<http://www.etsi.org/>], the European Telecommunications Standards Institute, is Europe's officially-recognized standardization body for Information and Communication Technologies (ICT), including telecommunications and broadcasting.

As previously announced through a press release

[http://www.etsi.org/pressroom/Previous/2006/2006_10_tispan.htm] ETSI/TISPAN is about to release a standard document for IMS benchmarking, ETSI TS 186 008. This document provides standard ways of setting up, measuring and reporting the performances of IMS based platforms and services. It applies not only for IMS/TISPAN, but also IMS/3GPP.

As a matter of fact, ETSI's official contact for this work item, Olivier Jacques

[<mailto:olivier.jacques@hp.com>], is also one of the main contacts for Seagull test tool. All the needs expressed by the ETSI standard document are currently being translated into Seagull so that benchmarks can be conducted by Seagull.

Conclusion

While not dedicated to IMS protocols, Seagull is a great tool for IMS testing. The beauty of Open Source model is that everyone can check by himself, modify or enhance at will, while not forgetting professional support if needed.

You even have the choice not to use Seagull. But who will?

For more information

www.hp.com/go/ims

IMS at HP

<http://gull.sourceforge.net/>

Seagull web site

© 2007 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Itanium is a trademark or registered trademark of Intel Corporation or its subsidiaries in the United States and other countries.

4AA0-XXXXENW, May 2006

