

J-OWAMP Web Interface

Java Implementation of OWAMP

User's Manual for versions 1.2 and 2.1

Authors: Helder Veiga, José L. Oliveira, Rui Valadas, Paulo Salvador, António Nogueira

Affiliation: University of Aveiro / Institute of Telecommunications Aveiro
hveiga@av.it.pt; {jlo, rv}@det.ua.pt; {salvador, nogueira}@av.it.pt

Last updated: 12-12-2005

Contents

1	Introduction and requirements	3
2	System architecture	5
3	J-OWAMP Web Interface Installation Instructions	7
3.1	Linux installation	7
3.2	Windows installation	8
4	J-OWAMP Web Interface Description	10
4.1	Detailed description of the section Configurations	11
4.2	Detailed description of the section Session Results	13
5	Annexed figures	15

1 Introduction and requirements

In order to create an innovator platform for active measurements, that can also represent a basis for the development and test of new algorithms and models, we built a system designated by J-OWAMP (that stands for Java implementation of OWAMP) that corresponds to the analogous of the One-Way Active Measurement Protocol (OWAMP) model and a user friendly web interface to improve its use. This web interface can be used with J-OWAMP versions 1.2 and 2.1. The J-OWAMP version 1.2 implements the May 2004 draft proposal of OWAMP (<http://www.internet2.edu/~shalunov/ippm/draft-ietf-ippm-owdp-08.txt>) and the J-OWAMP version 2.1 implements the December 2004 draft proposal of OWAMP (<http://www.internet2.edu/~shalunov/ippm/draft-ietf-ippm-owdp-14.txt>).

The J-OWAMP system can be used both in Windows and Linux platforms, requiring only the installation of J2SE Java Runtime Environment (JRE), available in both Windows and Linux versions. The installation of this module is mandatory in order to allow the execution of Java applications. J2SE Java Runtime Environment (JRE) 5.0 is needed for supporting IPv6 on Microsoft Windows. With the J2SDK/JRE 1.4 release, there is IPv6 support on Linux.

Note: For IPv6 support, addresses with scope link-local (start with FE8) or site-local (start with FEC) shouldn't be used.

You must also save a MySQL Connector/J file (<http://dev.mysql.com/downloads/>) in the "`<jre_directory>\lib\ext`" directory to allow the interaction between J-OWAMP and a MySQL database. MySQL Connector/J is used for connecting to MySQL from Java.

OWAMP requires a synchronized clock in order to provide meaningful measurements. But, more importantly, the clock needs to be stable. If the system clock is stepped during an OWAMP session, the results can be misleading. OWAMP requires the use of GPS or NTP to synchronize the system clock.

The time synchronization, accuracy and resolution are some of the most important requirements of OWAMP. Better accuracy and resolution can be obtained by using performance counters in conjunction with the system time, in order to calculate accurate and smaller time increments. To do so, a time reference should be defined. In any instant the current time can be determined as follows: $currentTime = timeReference + (currentCounterValue - referenceCounterValue)/counterFrequency$. The time reference can be defined by using the system time, in which case the system should be synchronized, or directly from a NTP server. J-OWAMP allows these two approaches. For the second approach, J-OWAMP includes a NTP client which can be used to get the time reference directly from a NTP server, without the requirement of a synchronized clock on the machine hosting J-OWAMP. This approach can lead to even better results than the first one. On both approaches, time reference should be frequently synchronized. Therefore, an update interval should be defined. The NTP server address, the update interval and other constants can be defined at the file `jowamp\JOWAMP_files\JOWAMP_Constants.php`.

To visualize the graphs, in the results section, the user must install a Flash Player (<http://www.macromedia.com/go/getflashplayer>).

The manual is structured in the following way: section 2 presents the system architecture; section 3 shows the J-OWAMP Web Interface installation instructions; section 4 describes in detail the J-OWAMP web interface and section 5 includes the annexed figures.

2 System architecture

The J-OWAMP system implements the OWAMP architecture shown in Figure 1:

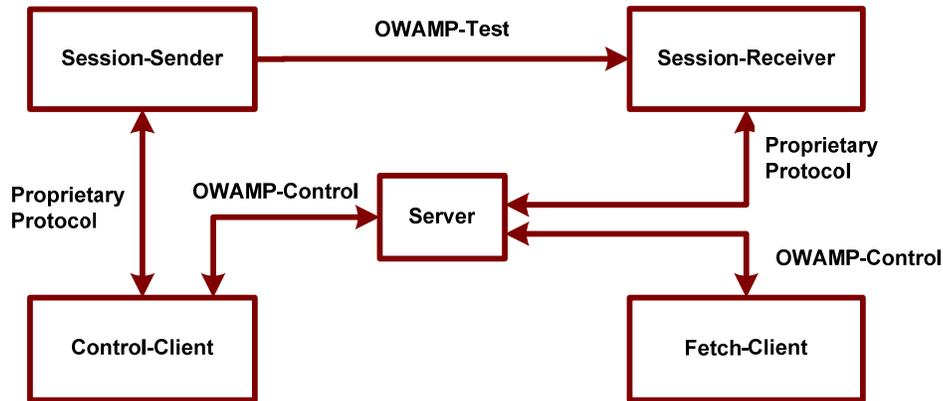


Figure 1. OWAMP architecture

The OWAMP architecture is based on two inter-dependent protocols, the OWAMP-Control and the OWAMP-Test, which can guarantee a complete isolation between client entities and server entities. The OWAMP-Control is used to begin and end test sessions as well as receive the results of those tests, whereas the OWAMP-Test protocol is used to allow the exchange of test packets between any two points that belong to the monitored network.

The proposed architecture includes the following elements:

Session-Sender: the sender of the test packets.

Session-Receiver: the receiver of the test packets.

Server: the entity that is responsible for the global management of the system. It can configure both network terminal elements that are being tested and receive the results of a test session.

Control-Client: a terminal system that programs requests for test sessions, triggers the beginning of a session set and can also finish one or all ongoing sessions.

Fetch-Client: a terminal system that triggers the requests for results of test sessions that have already ended or are still running.

A network element can carry out several logical functions at the same time (simplified scenario). For example, we can have only two network elements: one is carrying out the functions corresponding to a Control-Client, a Fetch-Client and a Session-Sender and the other one is carrying out the functions corresponding to a Server and a Session-Receiver (Figure 2).

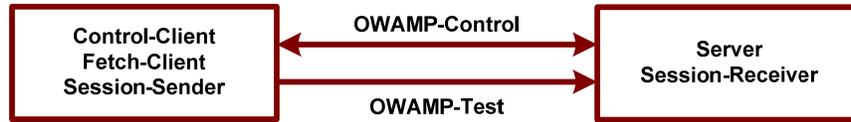


Figure 2. OWAMP simplified architecture

The architecture of Figure 1 allows the definition of only one client and one server in the network (possibly installed in machines with the highest processing capacity) and allows the installation of senders and receivers in any machine of the network, which leads to a lower processing impact. In this way, the network manager can perform tests all over the network from a single machine, which is not possible in the simplified scenario.

Note: All machines that are in charge of executing the different applications that compose this measurement system should be precisely synchronized in order to guarantee good measurements results.

3 J-OWAMP Web Interface Installation Instructions

3.1 Linux installation

In Linux, with Apache and PHP (with MySQL support)¹ installed (we used PHP Version 4.3.8):

- Unzip the J-OWAMP package to the `/var/www/html/` folder (Apache root directory);
- Install MySQL (<http://dev.mysql.com/downloads/>) and create the user “jowamp” with password “jowamp”, with all privileges;
- Install the J2SE Java Runtime Environment (JRE) (<http://java.sun.com/j2se/1.4.2/download.html>) for linux (use the **self-extracting file** ([j2re-1_4_2_08-linux-i586.bin, 13.72 MB](#))), in the `/var/www/html/` folder, and rename the folder `j2re<x.x.x_xx>` to `j2re`. Or change the path to the JRE bin folder in the `jowamp/JOWAMP_files/execInBackground.php` file, on line `'exec("/var/www/html/j2re/bin/" . $exe . " " . $args)';`
- Copy the file `mysql-connector-java-3.1.6-bin` to the `/var/www/html/j2re/lib/ext/` directory. This file is located in the `/var/www/html/jowamp/JOWAMP_files/auxiliaryFiles/` folder;
- Run `OWAMP_Server`, `OWAMP_SessionSender` and `OWAMP_SessionReceiver` using the following lines, in different terminal windows. For `OWAMP_Server`:
 - `cd /var/www/html/jowamp/JOWAMP_Files/owamp_protocol`
 - `/var/www/html/j2re/bin/java OWAMP_Server`

Use the same steps for `OWAMP_SessionSender` and `OWAMP_SessionReceiver`. For further information on running J-OWAMP elements, read the J-OWAMP user’s manual;

- In a web browser open the page <http://localhost/jowamp/login/register.php> to register new users. Only registered users can configure test sessions;
- Open the page <http://localhost/jowamp/jowamp.php> and use the Add terminal section to add your Senders and Receivers addresses.

¹ <http://www.php.net/manual/en/install.unix.php>.

3.2 Windows installation

In Windows, with Apache or IIS and PHP (with MySQL support)² installed (we used PHP Version 5.0.4):

- Unzip the J-OWAMP package to the default Apache or IIS web folder;
- Install MySQL (<http://dev.mysql.com/downloads/>) and create the user “jowamp” with password “jowamp”, with all privileges;
- Install the J2SE Java Runtime Environment (JRE) (<http://java.sun.com/j2se/1.4.2/download.html>) for Windows.
- Copy the file `mysql-connector-java-3.1.6-bin` to the “<jre_directory>\lib\ext” directory. This file is located in the `jowamp\JOWAMP_files\auxiliaryFiles\` folder;
- Run `OWAMP_Server`, `OWAMP_SessionSender` and `OWAMP_SessionReceiver` using the executables placed in the `jowamp\JOWAMP_files\owamp_protocol` folder. For further information on running J-OWAMP elements, read the J-OWAMP user’s manual;
- In a web browser, open the page <http://localhost/jowamp/login/register.php> to register new users. Only registered users can configure test sessions;
- Open the page <http://localhost/jowamp/jowamp.php> and use the Add terminal section to add your Senders and Receivers addresses.

Additional remarks:

Every terminal should be synchronized with each other. NTP can be an option, but GPS is the better solution.

The `OWAMP_Server` should be located in the same machine where the J-OWAMP Web server is installed.

When registering new terminal, the sender IP address should be in the following format:

```
IPv4_address:TCPportNumber // for IPv4 address  
[IPv6_address]:TCPportNumber // for IPv6 address
```

The `TCPportNumber` is the TCP port to be used in OWAMP-Control communication between the Control-Client and the respective OWAMP-SessionSender. The sender’s TCP port number corresponds to the port where the sender is waiting for connection requests. If the user wants to use the default `TCPportNumber` (4181), the sender IP address can be in the following format:

```
IPv4_address // for IPv4 address  
[IPv6_address] // for IPv6 address
```

² <http://www.php.net/manual/en/install.windows.php>.

The receiver address should be in the following format:

```
IPv4_address // for IPv4 address  
[IPv6_address] // for IPv6 address
```

The TCP port to use in the OWAMP-Control communication between the Server and receiver should be defined when running `OWAMP_Server`; otherwise the default TCP port will be used.

Note: Sender and Receiver address should have the same IP version (both IPv4 or IPv6). If one of these addresses, sender or receiver, is the loopback address the other should be loopback address too.

To run the `OWAMP_SessionSender` and `OWAMP_SessionReceiver` in other machines you can copy the folder `owamp_protocol` (in `jowamp/JOWAMP_Files/`) to those machines and then run these elements from there. Those machines must have JRE installed. To run these elements, use the same steps described above.

4 J-OWAMP Web Interface Description

The web interface includes a set of web pages that can be accessed from a single menu (Figure 3 and Figure 4). This menu has two levels. The first contains the following sections: *About J-OWAMP*, *Configurations*, *Session Results*, *Add Terminal* and *Download*. The second level can only be accessed from the first two sections of the first level and includes the following sub-sections: *About J-OWAMP* - *About* and *Details*, *Configurations* - *Single Session* and *Multiple Sessions*.

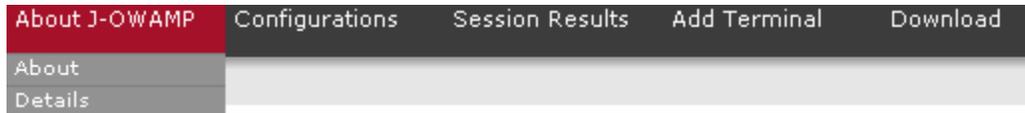


Figure 3. About J-OWAMP Menu

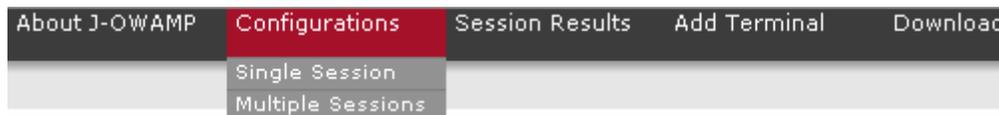


Figure 4. Configurations Menu

This menu allows a user to perform the following operations:

- *About J-OWAMP*: allows the visualization of J-OWAMP description information (*About* – Figure 5³) and more detailed information about the OWAMP protocol (*Details* - Figure 6);
- *Configurations*: allows the configuration of tests sessions - *Single Session* (Figure 8) and *Multiple Session* (Figure 9). For security reasons, these two sub-sections require user authentication (Figure 7). With J-OWAMP it is possible to accomplish two kinds of test sessions: single test sessions (*Single Session*) and confidence interval test sessions (*Multiple Session*). The single test session corresponds to the configuration of one single test between two machines. The user can define the number of packets, the packet interarrivals, the packet length and the start time of the session. The results obtained with this type of session are the estimation performance metrics such as mean delay, losses, duplication and throughput. The confidence interval test session allows splitting the complete session period in a number of smaller test intervals and to perform a set of tests in each interval to enable the construction of 90% confidence intervals. For example, if the session period is set at 24 hours, the user can define 24 intervals (of one hour duration), and define also that in each interval a total of 10 tests should be performed in order to calculate the 90% confidence interval. In this case, the interval between the start of consecutive session can also be defined (say 2 minutes);

³ To visualize any of the annexed figures just click in the respective name and then click in the corresponding figure to return to origin.

- *Session Results* (Figure 10): allows the visualization (in tables and graphs) of the results of the configured tests sessions and the statistics calculated using these results;
- *Add Terminal* (Figure 11): allows the registration of new OWAMP probes (senders and receivers) to be used in test sessions. These probes should be active;
- *Download* (Figure 12): this section allows downloading different information related to J-OWAMP (source code, executables, etc).

4.1 Detailed description of the section Configurations

In the *Configurations* section the user is supposed to introduce the test configurations. For both *Single Session* and *Multiple Session*, the user can define the following configurations (Figure 8 and Figure 9, respectively):

Introduce the number of test packets (Default 100): number of test packets to be sent

Choose the Sender's address: IP address of the machine where the OWAMP Session-Sender is installed. The IP addresses presented here are registered in the *Add Terminal* section.

Choose the Receiver's address: IP address of the machine where the OWAMP Session-Receiver is installed. The IP addresses presented here are registered in the *Add Terminal* section.

Note: These IPs are automatically tested after the submission of the test session configuration, in order to verify if there is connectivity to that machine and if the correspondent OWAMP element is actually installed in the machine with the given IP address. In case of error, the test session is not started and the correspondent error is presented on the screen (Figure 14). Sender and Receiver address should have the same IP version (both IPv4 or IPv6). If one of these addresses, sender or receiver, is the loopback address the other should be loopback address too. The loopback address can only be used if the Control-Client and the Server elements are running on the same host.

Introduce the padding length of the test packets (Default 0): additional size, in bytes, that can be introduced in the test packets (by default each test packet has at least 24 bytes) .

Introduce the Start Time of this session: Introduction of the start time of the session.

Year: Its value is the current year when configuring the session.

Month: Its value is the number representation of the current month when configuring the session (1 for January).

Date: Its value is the current day of the month when configuring the session.

Hour: Its value is the current hour when configuring the session.

Minute: Its value is the current minute when configuring the session plus one for Single Session and the current minute for Multiple Sessions.

Second: Its default value is zero.

Introduce the timeout in seconds for the test packets (Default 10): maximum time interval during which a packet should be received. Packets that arrive to Receiver after timeout seconds are considered as lost.

To choose the interarrival time between test packets the user has to define three parameters:

Choose the interarrival time distribution:

- Fixed
- Exponential

Introduce the number of packets Schedule slot descriptions (Default 1):

Introduce the interarrival time parameter, in seconds (Default 1):

These three parameters define a group of packets to be sent with the request session. Each packet represents a ‘slot’. So, we have a schedule with a given number of ‘slots’. Each slot has a type and a parameter. Two types are supported: exponentially distributed pseudo-random quantity (*Exponential*) and a fixed quantity (*Fixed*). The parameter is expressed as a timestamp and specifies a time interval. For a *Exponential* type this interval is the mean value (or $1/\lambda$, if the distribution density function is expressed as $\lambda e^{-\lambda x}$ for positive values of x). For a *Fixed* type, the parameter is the delay itself. The sender starts with the beginning of the schedule and executes the instructions in the slots: for a *Exponential* type, it waits for an exponentially distributed time interval, with mean equal to the specified parameter, and then it sends a test packet (and proceeds to the next slot); for a *Fixed* type, it waits for the specified time interval and sends a test packet (and proceeds to the next slot). The schedule is circular: when there are no more slots, the sender returns to the first slot.

For the *Multiple Session*, the user can also define the following configurations (Figure 9):

Introduce the Stop Time of this session: Introduction of the stop time of the session.

Year: Its value is the current year when configuring the session.

Month: Its value is the number representation of the current month when configuring the session (1 for January).

Date: Its value is the current day of the month when configuring the session.

Hour: Its value is the current hour when configuring the session.

Minute: Its value is the current minute when configuring the session.

Second: Its default value is zero.

Introduce the interarrival time between sets of tests, in minutes (Default 60): time interval between sets of measurements.

Introduce the number of tests in each set (Default 10): number of test sessions to accomplish in each interval.

Introduce the interarrival time between tests, in seconds (Default 120): temporary separation between the beginning instants of each individual test session, of a set of tests.

The number of confidence intervals is automatically calculated after the configuration of the *Multiple Sessions's* start time, stop time and interarrival time between sets of tests. For the example on Figure 9, all tests are performed in a 24 hours period. In each hour, sets of 10 tests are performed, making a total of 240 tests. In each group, the starting time instants of the tests are separated by 2 minutes. All tests lasted for 1 minute and consisted in sending 60 packets of 24 bytes each (minimum test packet size), at an average rate of 1 packet/second. In order to conveniently characterize the packet average delay and packet loss ratio, 90% confidence intervals are calculated based on the 10 average values obtained in each test belonging to a group of 10 tests.

4.2 Detailed description of the section Session Results

The *Session Results* section is the more complex section. In this section the user can choose which representation he wants to use to visualize results (graph - Figure 18 or table - Figure 15) and which type of session those results belong to: a *Single Session (SS)* and/or a Confidence Interval (CI) session. The statistics results are calculated at the first time they are accessed and are saved on the local database of the web server where the J-OWAMP web interface is located.

To visualize the results, a user can take one of the following actions (Figure 10):

- To see the configuration details of a session, just click its user ID;
- To see the results details of a session choose the result representation (Graph or Table) on the same row of the corresponding session;
- To see the results details of two or more sessions at the same time select the checkbox of the corresponding sessions and the See Details button located at the end of the page.

For the graph representation, if one of the test sessions selected by the user is of the *Multiple Session (CI)* type, the user has the option to choose which statistics he wants to visualize. There are many kinds of statistics to choose: average delay, packet loss, throughput, etc (Figure 17).

For *SS* sessions, if the user chooses the table representation (Figure 15) he can click the session user ID to visualize the session's configuration and individual packet results details (Figure 16). The table representation (Figure 15) has the following statistics:

- The number of packets actually sent. For J-OWAMP version 1.2, its value is equal to the number of packets configured to be sent. For J-OWAMP version 2.1, its value is equal or less than the number of packets configured to be sent because if the sending instant of a packet is in the past the packet is not sent;

- The number of packets received: its value represents the number of all received packets, including the number of duplicated packets and packets that were received before the time out configured for the corresponding test session;
- The absolute delay values in milliseconds (minimum, average and maximum) and the delay variance;
- The time out loss ratio: represents the number of packets that were received before the time out configured for the corresponding test session. Its value is equal to $\frac{\text{number of packets received before time out}}{\text{number of packets actually sent}}$;
- The not received loss ratio: represents the number of packets that were sent and not received. Its value is equal to $\frac{\text{number of packets lost on the network}}{\text{number of packets actually sent}}$;
- The duplication ratio: represents the number of packets that were duplicated by the network (their sequence numbers were received more than once). Its value is equal to $\frac{\text{number of packets duplicated on the network}}{\text{number of packets actually sent}}$;
- The throughput: represents the throughput generated by J-OWAMP. Its value is equal to $\frac{\text{number of packet actually sent} \times \text{test packet length}}{\text{last packet sent time} - \text{first packet sent time}}$.

For *CI* sessions, if the user chooses the table representation (Figure 15) he can take one of the following actions:

- To see the configuration and results details of a session (includes all individual packet delays of each single session belonging to each interval) just click its user ID;
- To see the results details of an interval, click the corresponding interval number (Figure 19 and Figure 20). The results details of an interval include statistical results of each single session of the corresponding interval (Figure 19);
- To see the detailed statistical results of a single parameter (delay, loss, duplication or throughput) in a table layout, using a format that is easier to copy, click the corresponding parameter (Figure 21).

5 Annexed figures

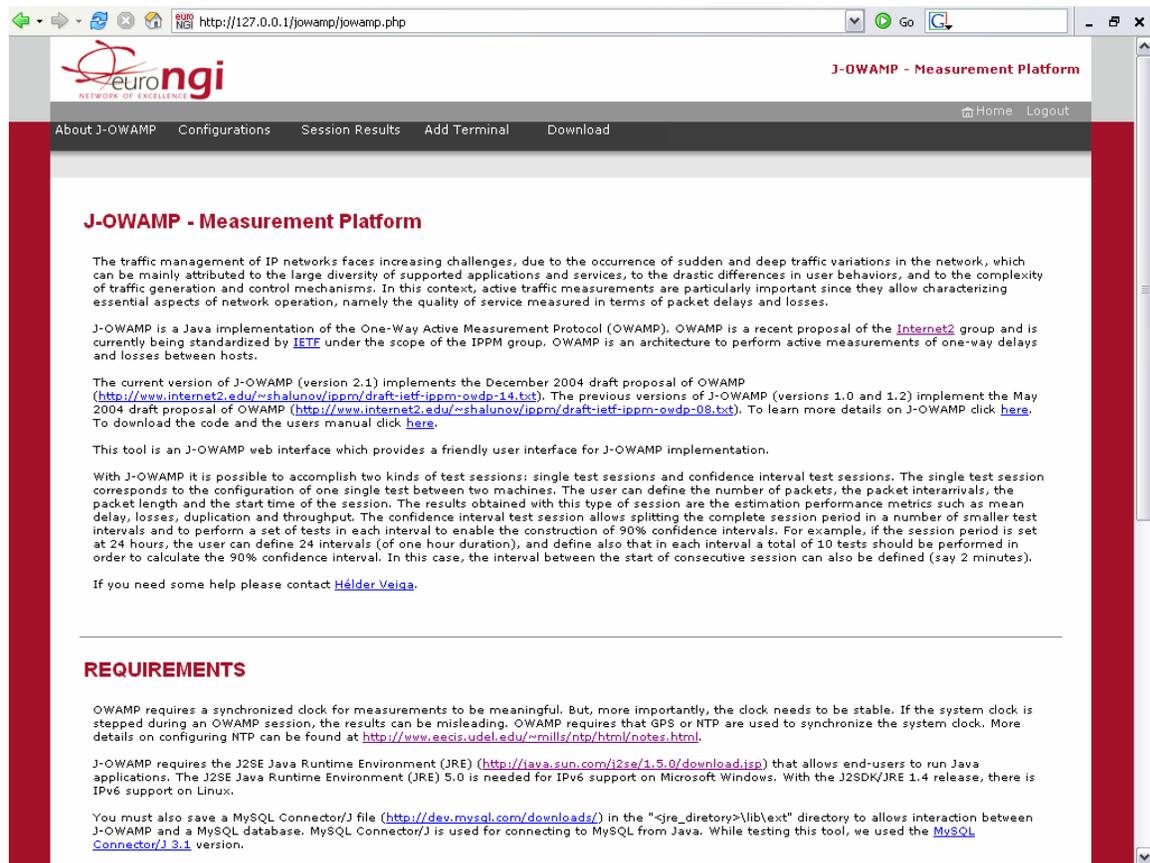


Figure 5. About J-OWAMP

ARCHITECTURE

J-OWAMP implements the OWAMP architecture shown below. The OWAMP architecture is based on two inter-dependent protocols, OWAMP-Control and OWAMP-Test, that can guarantee complete isolation between client entities and server entities. The OWAMP-Control is used to start and end OWAMP-Test sessions as well as receive the results of those tests, whereas the OWAMP-Test protocol is used to allow the exchange of test packets between any two points that belong to the monitored network.

The proposed architecture includes the following elements (Figure 1):

- **Session-Sender** - the sending endpoint of an OWAMP-Test session;
- **Session-Receiver** - the receiving endpoint of an OWAMP-Test session;
- **Server** - an end system that manages one or more OWAMP-Test sessions, is capable of configuring per-session state in session endpoints, and is capable of returning the results of a test session;
- **Control-Client** - an end system that initiates requests for OWAMP-Test sessions, triggers the start of a set of sessions, and may trigger their termination;
- **Fetch-Client** - an end system that initiates requests to fetch the results of completed OWAMP-Test sessions.

Fig. 1. OWAMP architecture.

A network element can carry out several logical functions at the same time. For example, we can have only two network elements, one implementing the Control-Client, the Fetch-Client and the Session-Sender and the other implementing the Server and the Session-Receiver (Figure 2).

Figure 6. Details

Single Session - Authentication

Login

Email:

Username:

Password:

Site version 2.1

Figure 7. Single Session - Authentication

J-OWAMP - Single Session

Introduce the number of test packets (Default 100):

Choose the Sender's address:

Choose the Receiver's address:

Introduce the padding length of the test packets (Default 0):

Introduce the Start Time for this session:

Year:

Month:

Date:

Hour:

Minute:

Second:

Introduce the timeout in seconds for the test packets (Default 10):

Choose the interarrival time distribution:

Fixed

Exponential

Introduce the number of packets Schedule slot descriptions (Default 1):

Introduce the interarrival time parameter, in seconds (Default 1):

Figure 8. Single Session

The screenshot shows a web browser window with the URL `http://127.0.0.1/jowamp/JOWAMP_files/JOWAMP_ShowPage.php?link=JOWAMP_MultipleSession.php`. The page header includes the "eurongi NETWORK OF EXCELLENCE" logo and the title "J-OWAMP - Measurement Platform". A navigation menu contains links for "About J-OWAMP", "Configurations", "Session Results", "Add Terminal", "Download", "Home", and "Logout".

The main content area is titled "J-OWAMP - Multiple Sessions (with confidence intervals)". It contains the following configuration fields:

- Introduce the number of test packets (Default 60):
- Choose the Sender's address:
- Choose the Receiver's address:
- Introduce the padding length of the test packets (Default 0):
- Introduce the Start Time for this session:
 - Year:
 - Month:
 - Date:
 - Hour:
 - Minute:
 - Second:
- Introduce the timeout in seconds for the test packets (Default 10):
- Choose the interarrival time distribution:
 - Fixed
 - Exponential
- Introduce the number of packets Schedule slot descriptions (Default 1):
- Introduce the interarrival time parameter, in seconds (Default 1):
- Introduce the Stop Time for this session:
 - Year:
 - Month:
 - Date:
 - Hour:
 - Minute:
 - Second:
- Introduce the interarrival time between sets of tests, in minutes (Default 60):
- Introduce the number of tests in each set (Default 10):
- Introduce the interarrival time between tests, in seconds (Default 120):

A "Submit" button is located at the bottom of the form. The footer of the page indicates "Site version 2.1".

Figure 9. Multiple Session

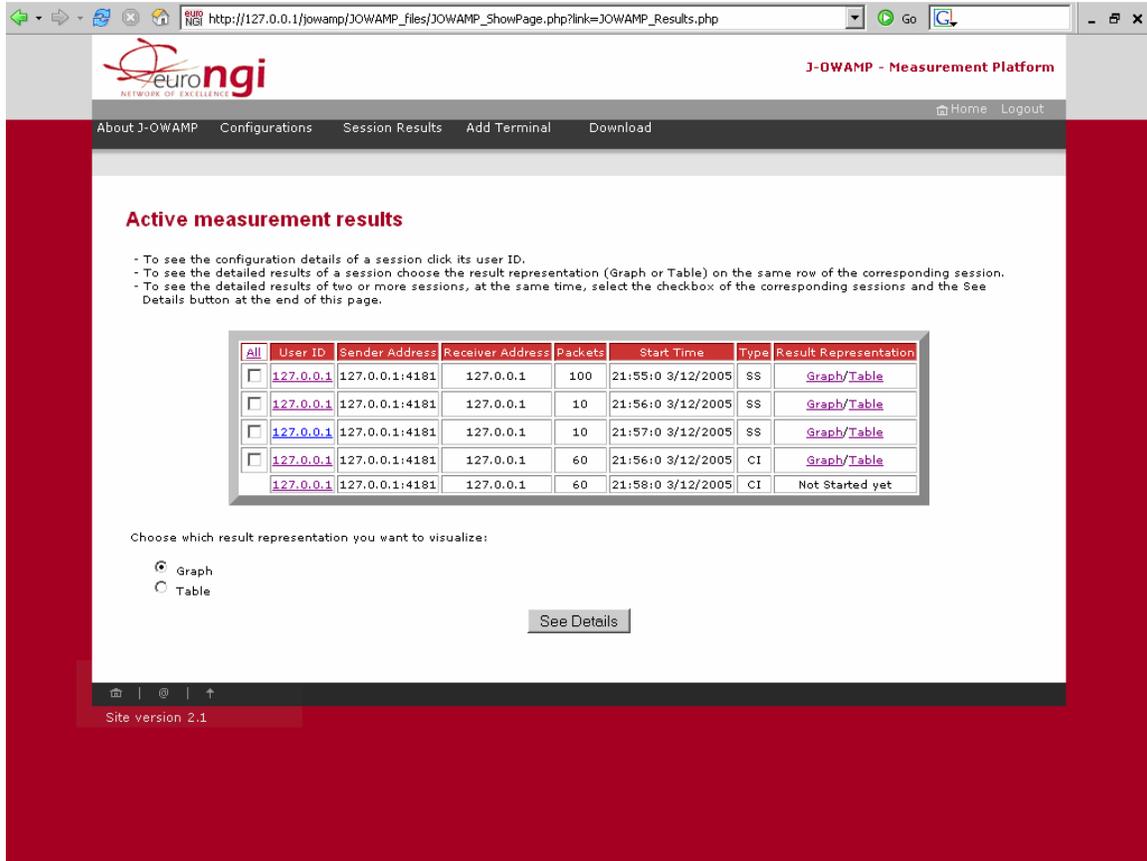


Figure 10. Session Results

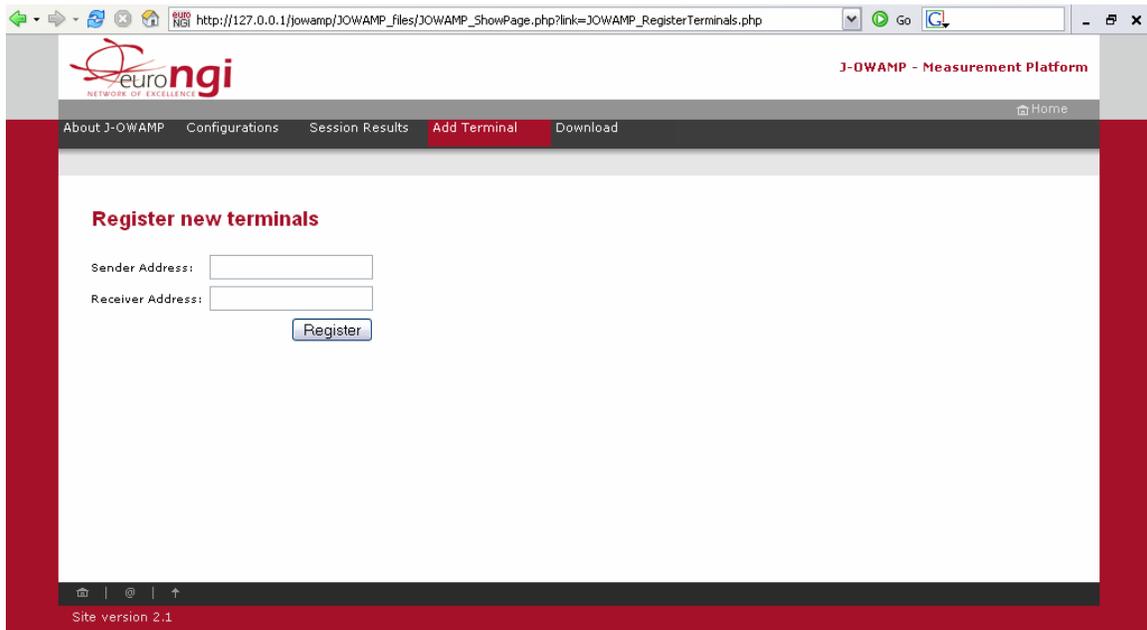


Figure 11. Add a new Terminal

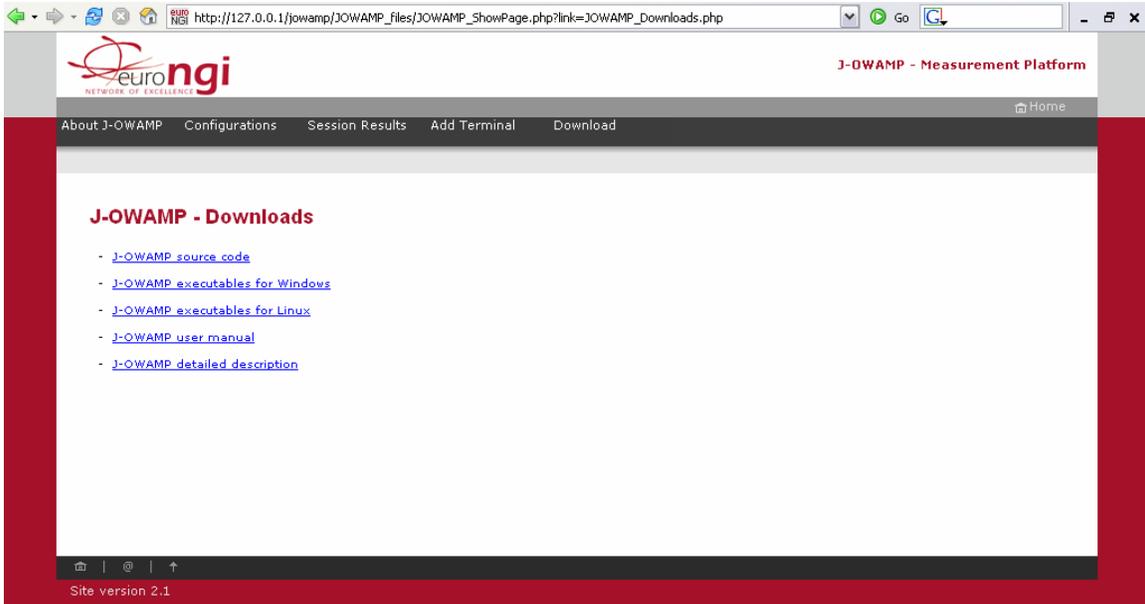


Figure 12. Download

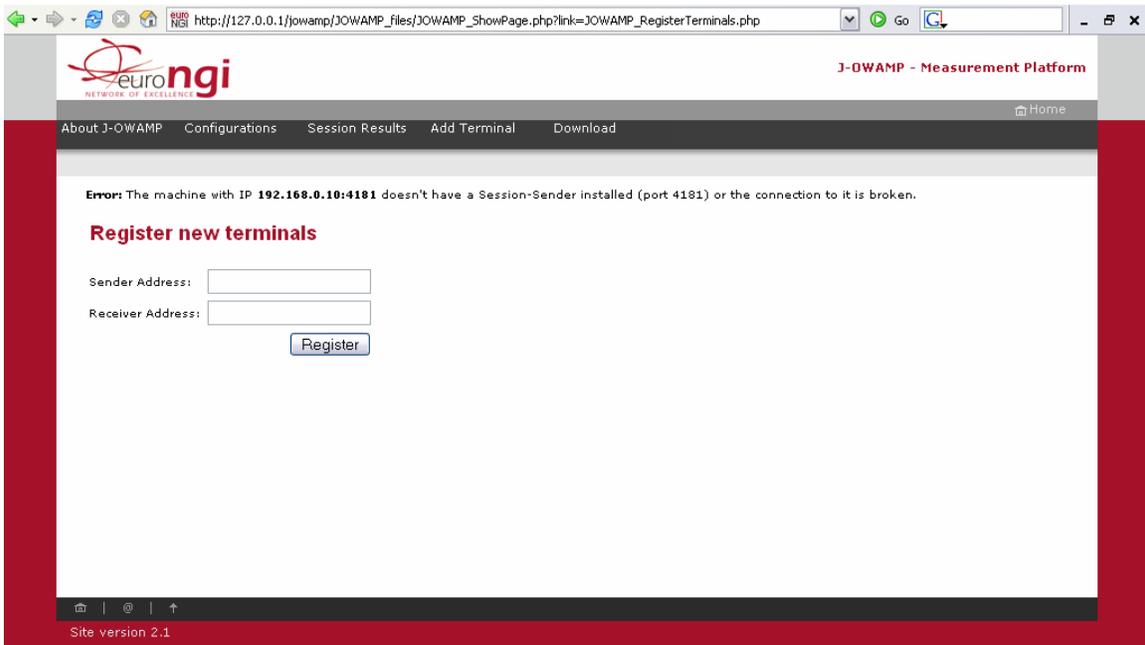


Figure 13. Register Terminals

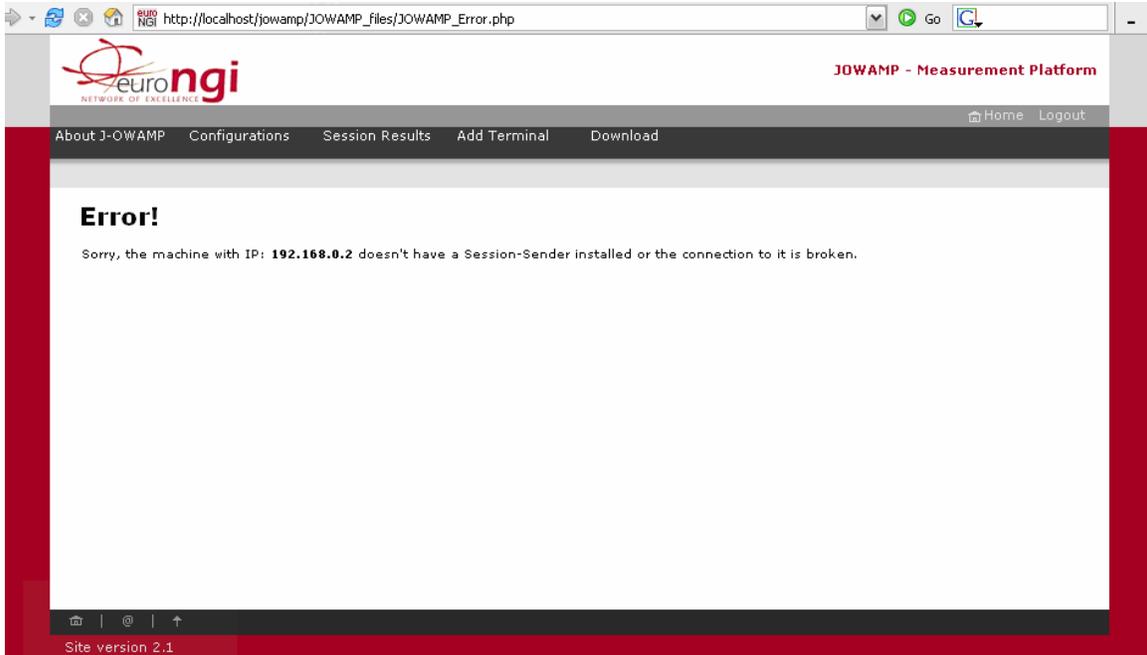


Figure 14. Connectivity verification

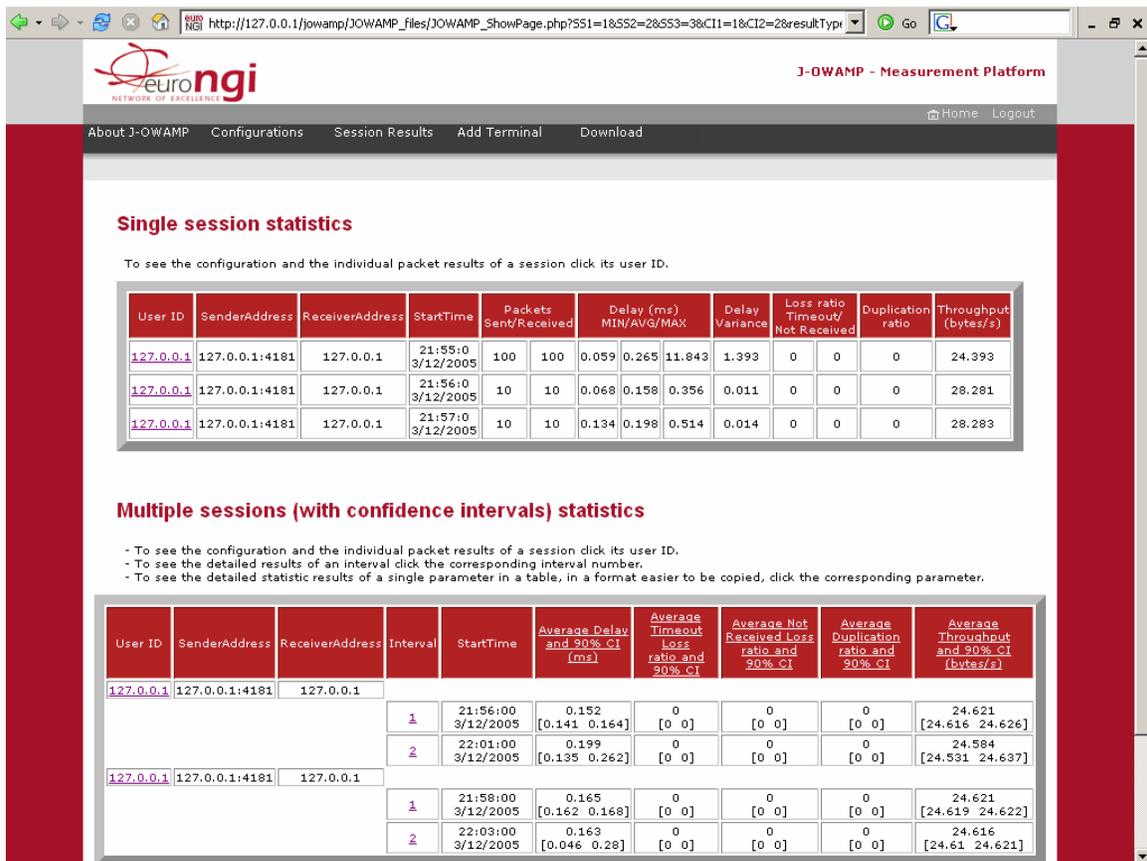


Figure 15. Results representation using Tables.

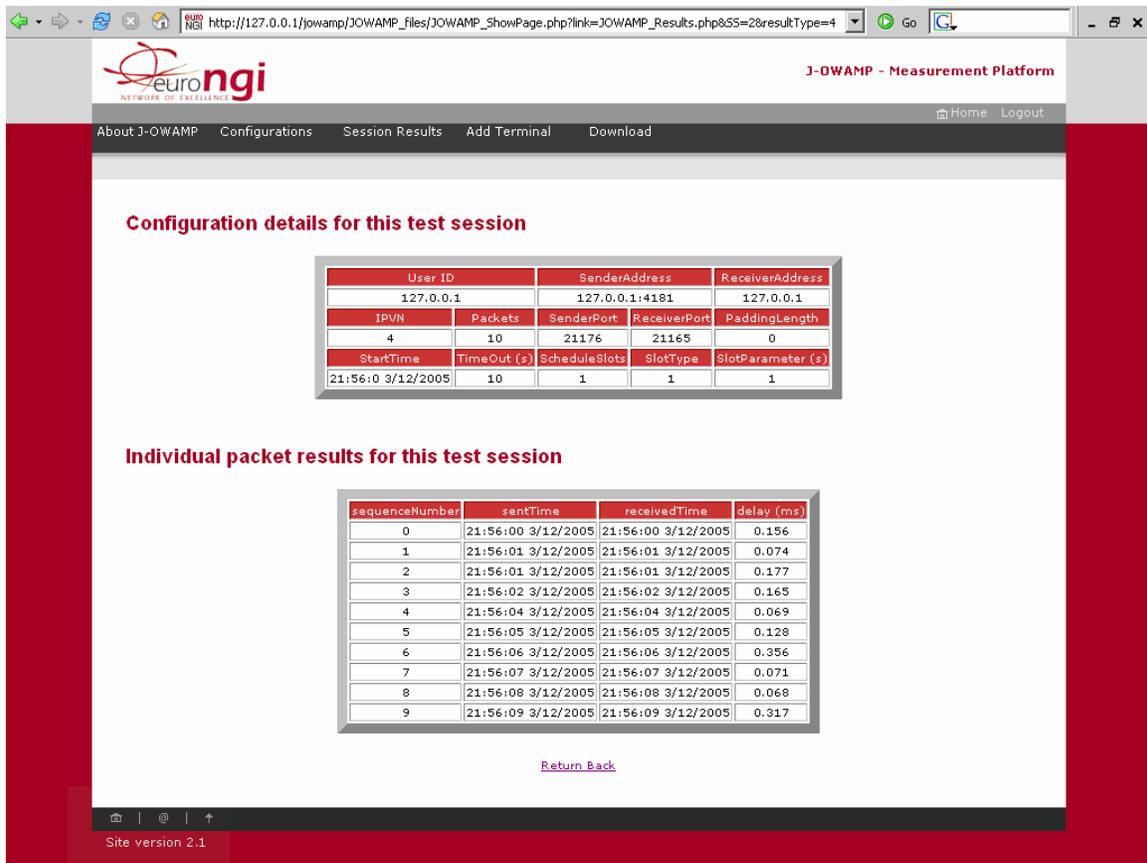


Figure 16. Configuration details and individual packet statistics.

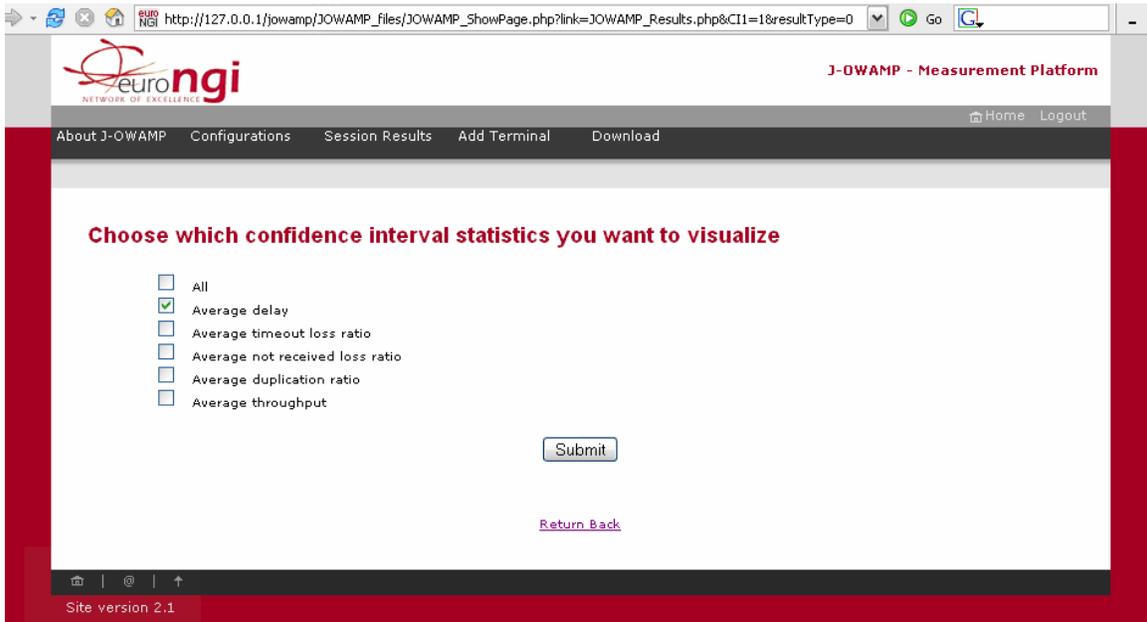


Figure 17. Multiple Session confidence interval.

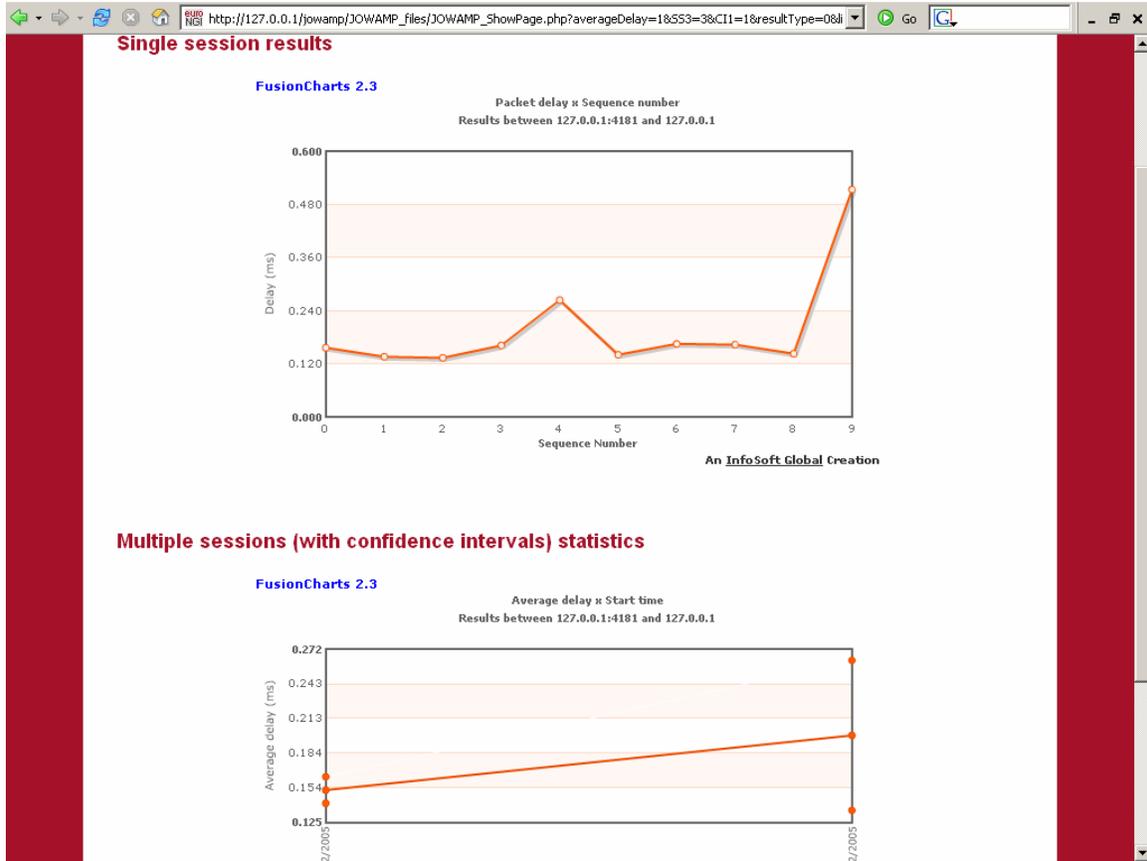


Figure 18. Results representation using graphs.

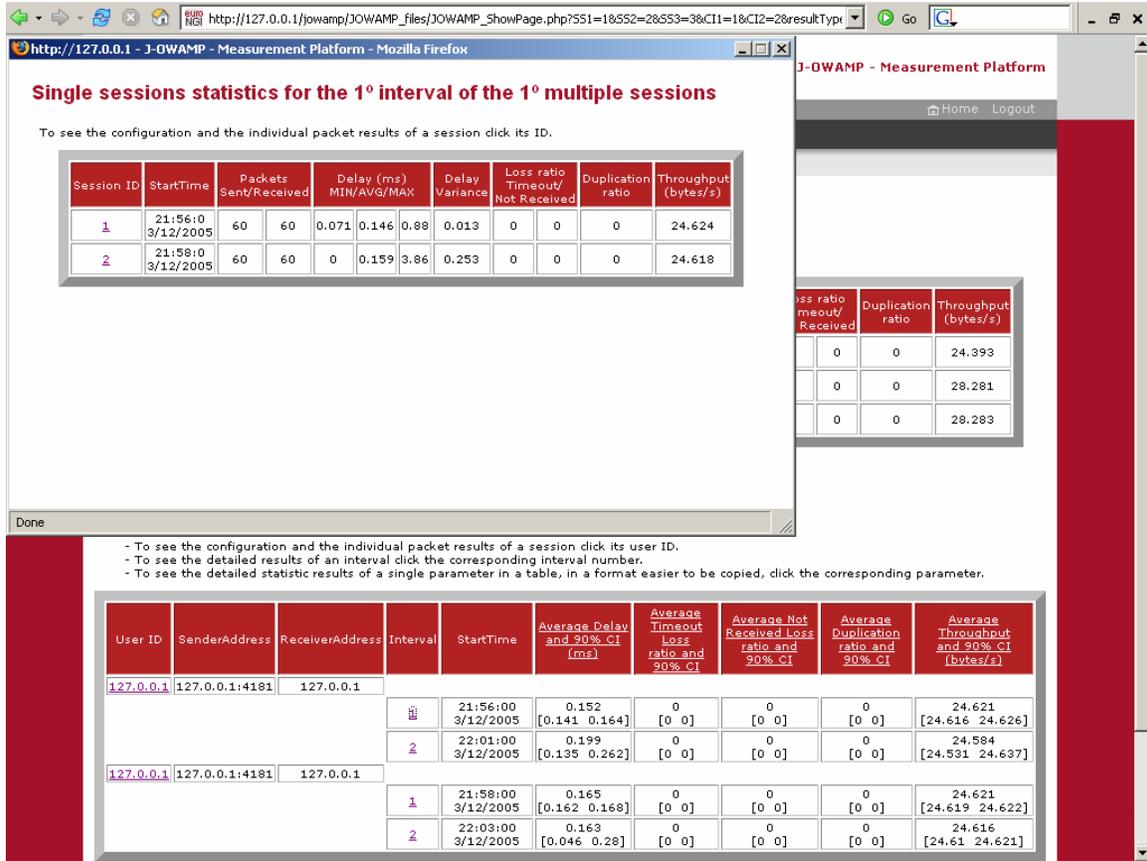


Figure 19. Single sessions statistics for an interval of a Confidence Interval session.

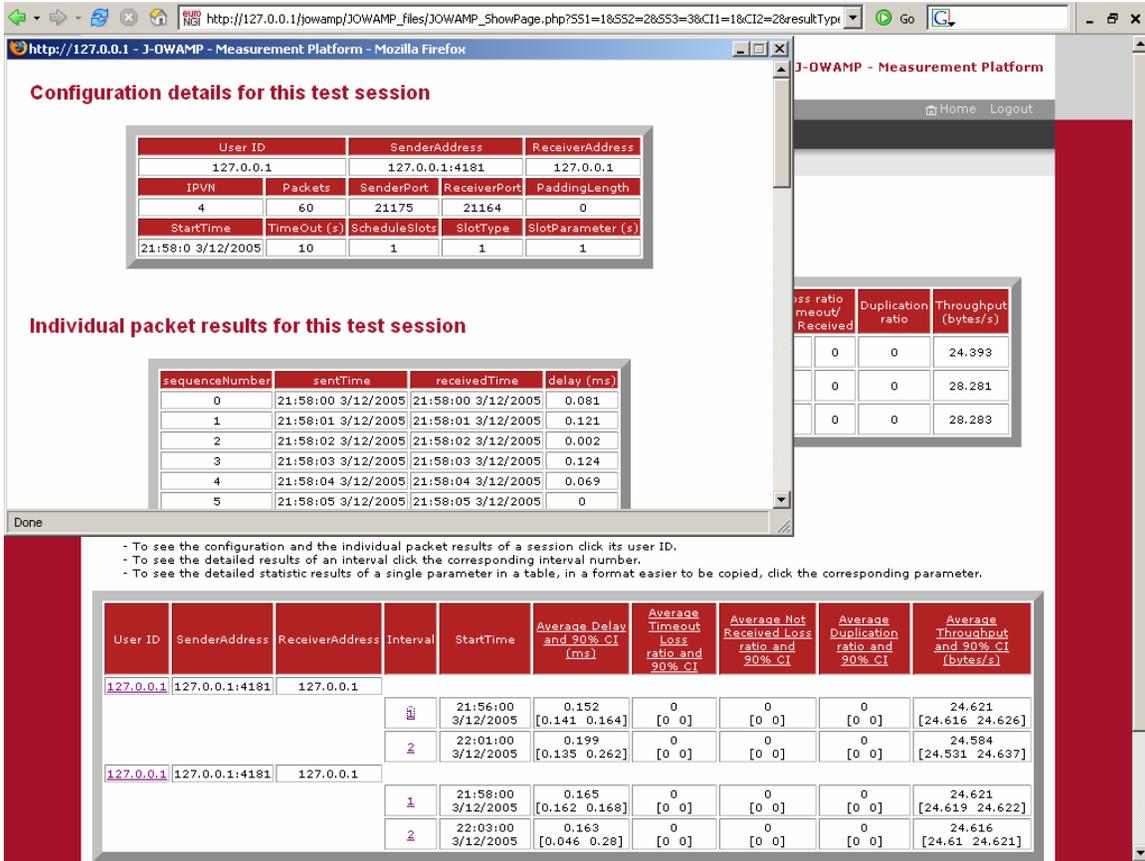


Figure 20. Configuration details and individual packet statistics of a single session of an interval of a Confidence Interval session.

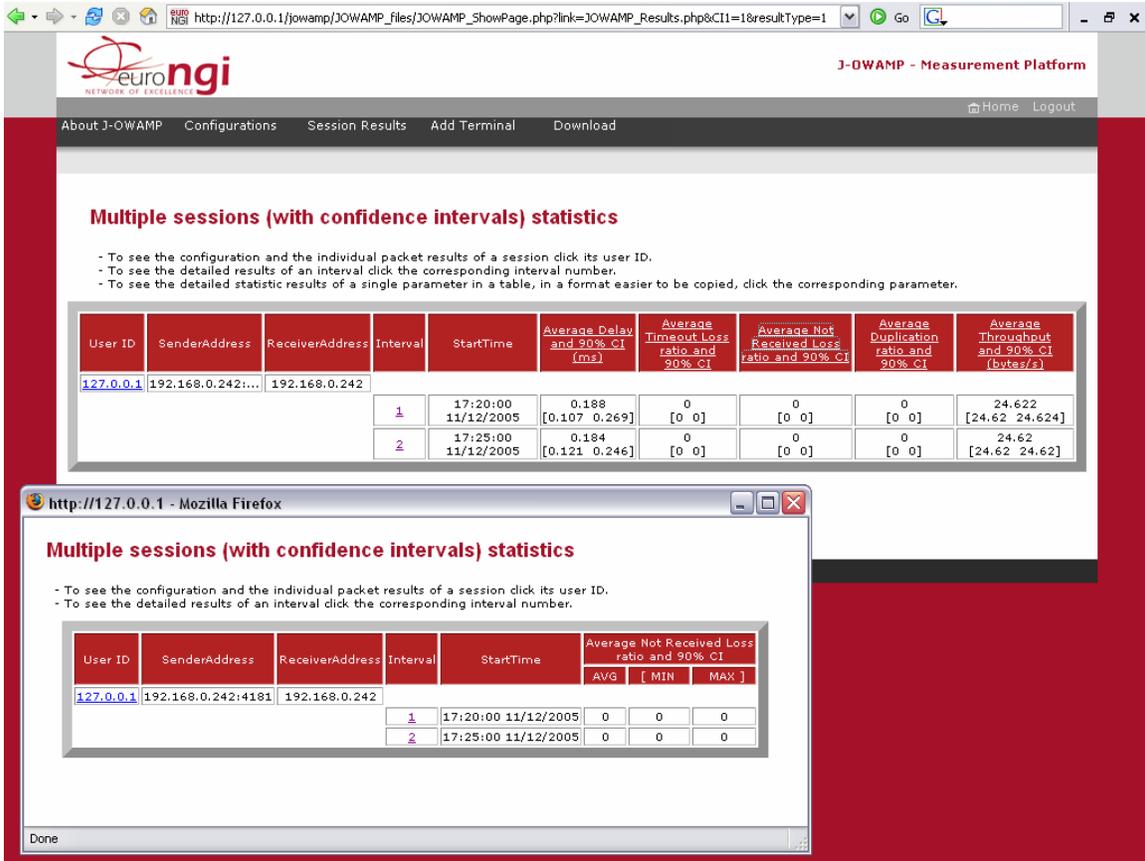


Figure 21. Detailed statistic results of a single parameter of a Multiple Sessions test.