**Waarp R66 Setup Document**

**Required url’s for the project:**

* Git url for the Waarp server -<https://github.com/t3ctechnologies/WaarpR66>
* Commands to setup Waarp server - [https://waarp.github.io/Waarp/Waa HYPERLINK "https://waarp.github.io/Waarp/WaarpR66Command.html" HYPERLINK "https://waarp.github.io/Waarp/WaarpR66Command.html" HYPERLINK "https://waarp.github.io/Waarp/WaarpR66Command.html"r HYPERLINK "https://waarp.github.io/Waarp/WaarpR66Command.html" HYPERLINK "https://waarp.github.io/Waarp/WaarpR66Command.html" HYPERLINK "https://waarp.github.io/Waarp/WaarpR66Command.html"pR66Command.html](https://waarp.github.io/Waarp/WaarpR66Command.html)

**Prerequisites:**

* Eclispse -Mars or any above versions
* MySql Workbench - version 6.3.8
* Apache Maven - version 3.5.0 or Other
* Waarp Project - version 3.0.8

**Update Configuration files:**

Files to be updated before building source,

* config-serverA.xml
* All the referred file paths in this file should be updated with the absolute source path (All the files referred in this file are part of Waarp project source code)
* Update serverhome tag with the path where Waarp should store all the files.
* Update schema name and credentials of mysql under db tag.
* OpenR66-authent-A.xml – Update all the paths to the latest source file paths.
* config-clientA.xml – All the paths for the files in this file should be updated with the exact source path
* All the referred file paths in this file should be updated with the absolute source path (All the files referred in this file are part of Waarp project source code)
* Update serverhome tag with the path where Waarp should store all the files.

Building Waarp server steps:

* Initialize the database – This step will create all the required tables into the database schema.

File to run: ServerInitDatabase.java

Arguments(2): src\main\config\config-serverA.xml –initdb

Result: This will create all the required tables for the Waarp server like configuration, hostconfig, hosts, multiplemonitor, rules, runner, sequences.

* Loading business configurations into database – This step will load the business configuration into the hostconfig table.

File to run: ServerInitDatabase.java

Arguments(3): src\main\config\config-serverA.xml –loadBusiness src\main\config\config-serverA.xml

Result: This will update hostconfig table with business config details.

* Loading Aliases – This step will load all the aliase configurations into the database.

File to run: ServerInitDatabase.java

Arguments(3): src\main\config\config-serverA.xml –loadAlias

src\main\config\config-serverA.xml

Result: This will update hostconfig table with aliases details.

* Loading Roles – This step will load role configurations into the database.

File to run: ServerInitDatabase.java

Arguments(3): src\main\config\config-serverA.xml –loadRoles

src\main\config\config-serverA.xml

Result: This will update hostconfig table with role details.

* Loading Rules – This step will load rule configurations into the database.

File to run: ServerInitDatabase.java

Arguments(3): src\main\config\config-serverA.xml –dir

\*\*\*Directory path of rules\*\*\*

Result: This will update rules table with all the rule details.

* Loading Authentication – This step will load authentication details into the database.

File to run: ServerInitDatabase.java

Arguments(3): src/main/config/config-serverA.xml -auth src/main/config/OpenR66-authent-A.xml

Result: This will update hosts table with all the server details.

* Loading Bandwidth details – This step will load bandwidth details into the database.

File to run: ServerInitDatabase.java

Arguments(3): src/main/config/config-serverA.xml -limit src/main/config/limitConfiga.xml

Result: This will update configuration table with all the bandwidth details

* Upgrading Database – This step will upgrade the database schema.

File to run: ServerInitDatabase.java

Arguments(2): src/main/config/config-serverA.xml -limit

Result: This will update configuration table with all the bandwidth details

* Launch Waarp Server –

File to run: R66Server.java

Arguments(1): src/main/config/config-serverA.xml

Result: This will start the Waarp server. In console it will display the status of server.

Once Waarp starts successfully, we can access the warp admin tool and monitoring tool through browser.

Waarp R66-Web-Administration: The administrator allows to take action on transfers (stop, restart, ...), export logs, handle hosts and rules,

and some specific internal functions. It is accessed [https://host\_address:8067](https://host_address:8067/).

8) Waarp R66-Web-Monitoring : The supervision enables only to show information on transfers (active or not) and is dynamically reloaded every 10 seconds. It is accessed [http://host\_address:8066](http://host_address:8066/).

Other Configurations:

R66 Rule Configuration: Create one folder with a name rules under R66

server home directory, and place it all the six rule config files. \* Depends on mode of transfers(1=SEND 2=RECV 3=SEND+MD5 4=RECV+MD5), transmission of file will done.  
\* You can configure type of task, which you want to perform after or before transfer.(pretask, posttask)

Ex: rulerecv.rule.xml, rulerecv-any.rule.xml, rulerecvthrough-any.rule.xml, rulesend-any.rule.xml, rulesend.rule.xml, rulesendthrough-any.rule.xml