



Network Security Simulator 2 (*NeSSi²*)



Installation Guide

Version 2.0.0-beta.2



CC ACT
Agent Core Technologies



CC COG
Cognitive Architectures



CC IRML
Information Retrieval
& Machine Learning



CC NEMO
Network & Mobility



CC NGS
Next Generation Services



CC SEC
Security

Editors:

Karsten Bsufka and Rainer Bye

Contributors:

Arik Messerman

Katja Luther

Stephan Schmidt

Joël Chinnow

Dennis Grunewald

Thomas Hauschild

Sebastian Linkiewicz

Thorsten Rimkus

Jakob Strafer

DAI-Labor der Technischen Universität Berlin

Prof. Dr.-Ing. habil. Sahin Albayrak

Technische Universität Berlin / DAI-Labor

Institut für Wirtschaftsinformatik und Quantitative Methoden

Fachgebiet Agententechnologien in betrieblichen Anwendungen und der Telekommunikation

Sekretariat TEL 14

Ernst-Reuter-Platz 7

10587 Berlin

Telefon: (030) 314 74000

Telefax: (030) 314 74003

E-mail: Sekretariat@dai-labor.de

WWW: <http://www.dai-labor.de>

Contents

1	Installation of the Back End	1
1.1	Installation	1
1.2	Starting	1
1.3	Pitfalls	1
2	Installation of the Front End	3
3	Installation of the Database Management System	5
4	Configuration	7
4.1	Connect the front end to the back end	7
4.2	Database Configuration for the Front End	8
4.3	Database Configuration of the Back End	8
	List of Figures	11

1 Installation of the Back End

The back end is the component of *NeSSi²* that in fact executes the simulations. It generally needs not to reside on the same host as the front end.

1.1 Installation

To install the back end component, you have to

1. download the newest version from `www.nessi2.de`
2. extract the files from the downloaded zip file

1.2 Starting

The back end is started by executing the `NeSSiAgentSimulation` script residing in the `bin` folder

- `NeSSiAgentSimulation.bat` on Windows systems
- `NeSSiAgentSimulation.sh` on Systems with shell environment (Unix, Linux, Mac OS X, ...)

1.3 Pitfalls

1. For simulation of large networks it can be necessary to adapt the memory parameter in the start script:
 - `-Xms512M` for the minimal required memory
 - `-Xmx1G` for the maximal possible memory
2. Plugins should be in `conf/NeSSi2Plugins`
3. If the backend runs on a different host than the GUI see section 4.1 for a detailed description

2 Installation of the Front End

The front end is the GUI component of *NeSSi2*² (Graphical User Interface). You will need this component to create networks, profiles, scenarios and sessions and to start and monitor running simulations.

To install the front end component, you have to

1. download the version applicable for the host architecture and Operating System from `www.nessi2.de`
2. extract the files from the downloaded zip file

The front end is started by executing the `NeSSi2` application:

- `NeSSi2.exe` on Windows systems
- `NeSSi2` on Linux/GTK

3 Installation of the Database Management System

The data incoming during simulations is stored within a database. Although SQLite is supported since Version 2.0.0-beta2, it is recommended to install a “real” DBMS – such as MySQL – for productive use.

To set up a DBMS for use with *NeSSi*², you have to

1. install the DBMS as usual
2. create a database user with read and write access (maybe also from other hosts)
3. optionally create a database user with read access only (maybe also from other hosts)
4. do not forget to flush the new privileges if that is necessary

At the moment, the following Database Management Systems are supported by *NeSSi*²:

- MySQL
- SQLite

If you would like to use an SQLite database, it is recommended to run the front end and the back end on the same host.

4 Configuration

Before you can start, you have to configure the database settings for the front end. You may also optionally configure the database connections for the backend, but this should not be necessary in most cases.

4.1 Connect the front end to the back end

You need to set the name of the host running the simulation back end within the GUI preferences (“Window” → “Preferences” → “NeSSi2” → “Simulation Settings”). The default setting for this is `localhost`.

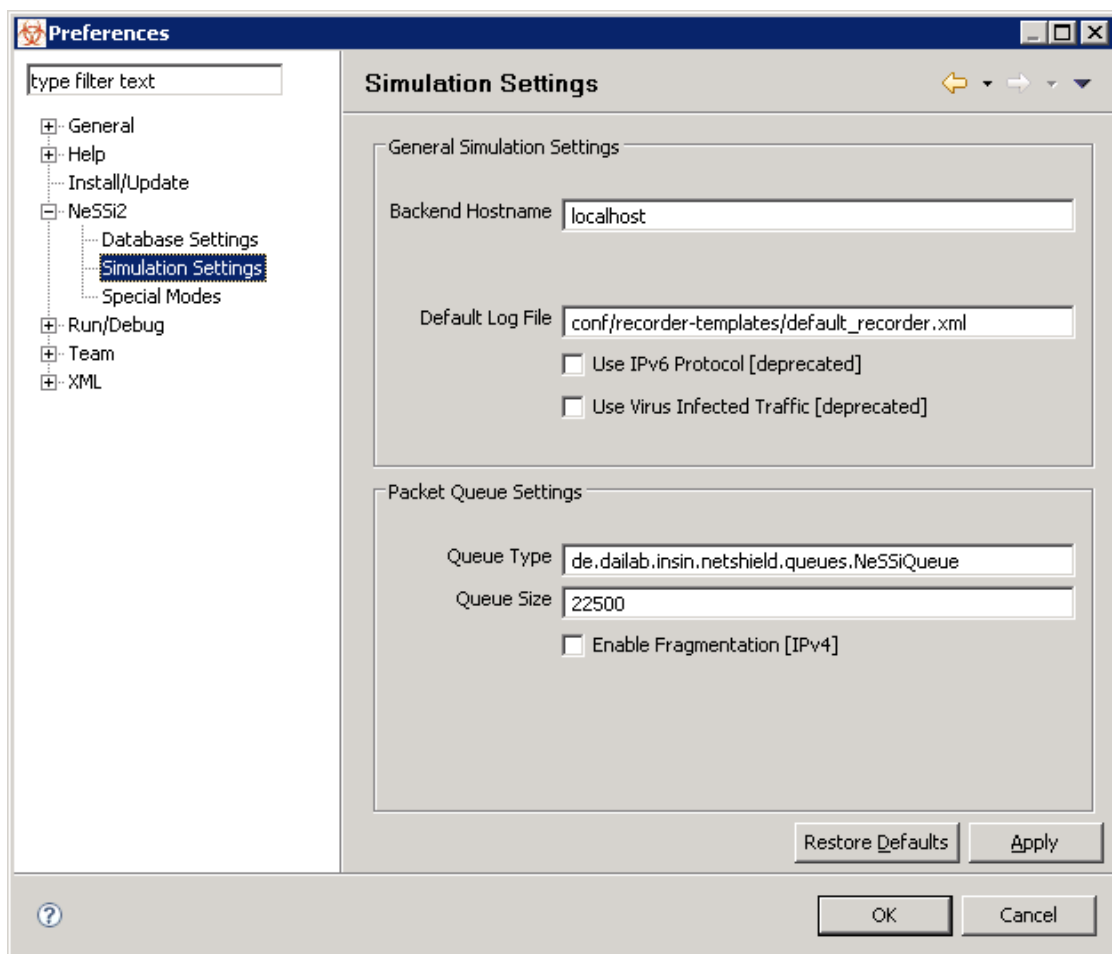


Figure 4.1: Simulation Settings

4.2 Database Configuration for the Front End

In the *NeSSi*² database preferences (“Window” → “Preferences” → “NeSSi2” → “Database Settings”) you need to set the username, his password, the connection URL, the name of the database schema and the driver. An example setting is depicted in figure 4.2

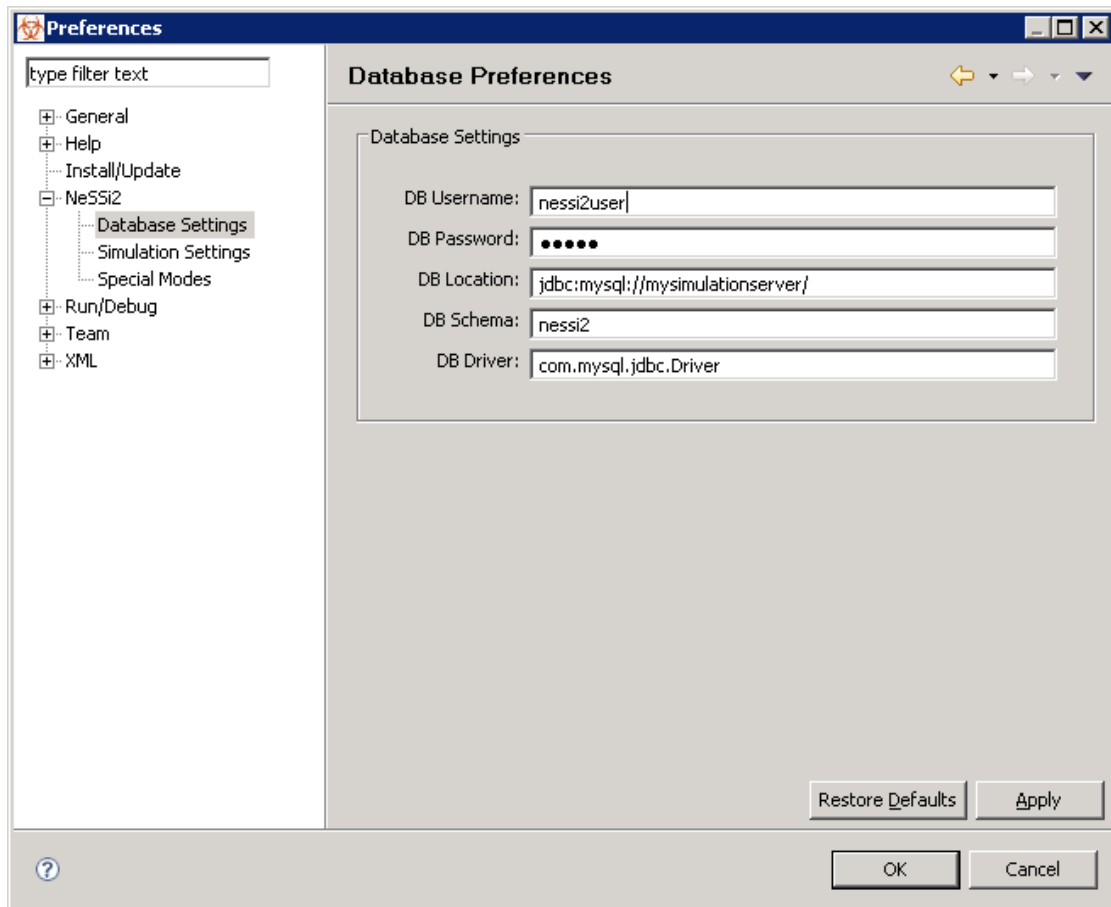


Figure 4.2: Example settings for connecting to a MySQL database on the server with the hostname “mysimulationserver”

Connections to SQLite databases have to be specified in another way which is depicted in figure 4.3. The contents of the username and password fields are irrelevant. The DB Location should only be set to “jdbc:sqlite:”, since the name of the SQLite database file goes in the DB Schema field. It is strongly recommended to specify the filename with its full path.

4.3 Database Configuration of the Back End

The back end gets its configuration from the recorder settings XML file, which is specified when a session is created (see *HowTo* document). Missing database connection settings default to those made within the front end preferences (see 4.2).

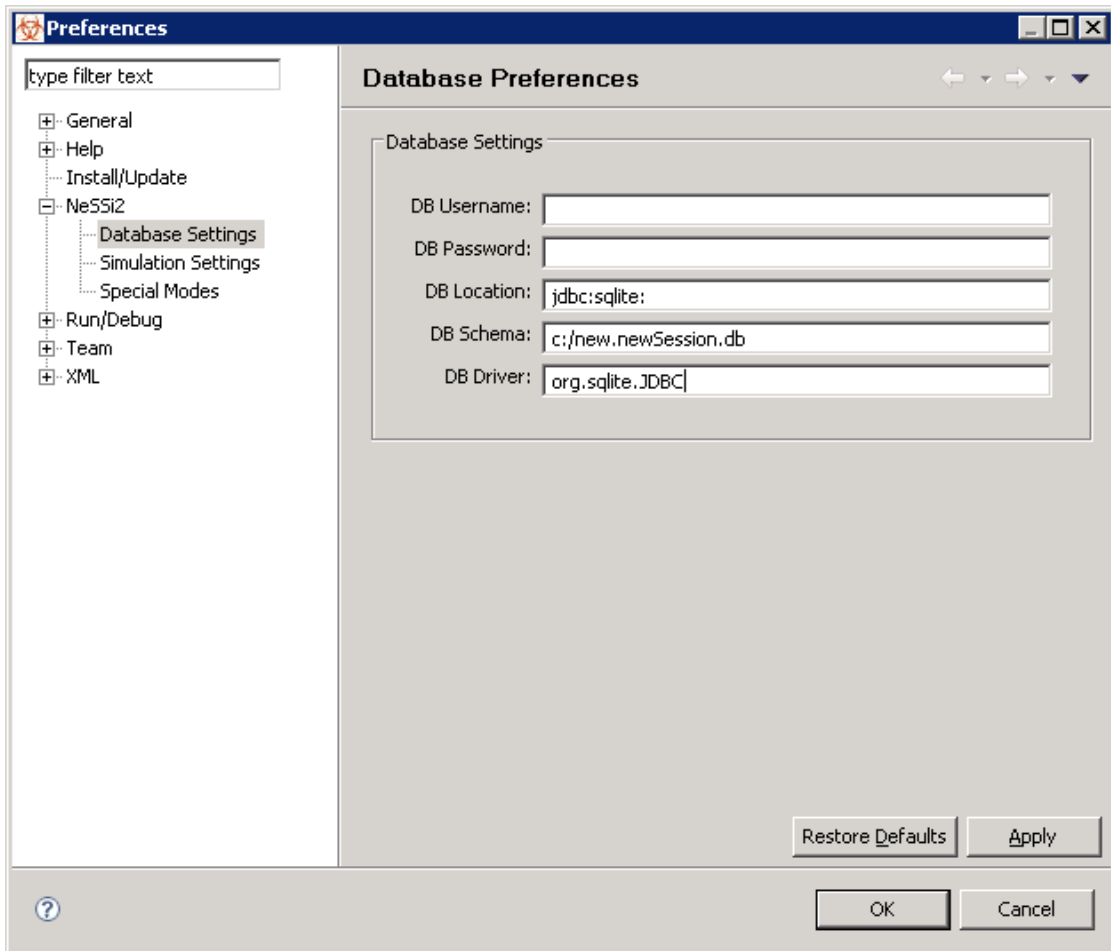


Figure 4.3: Example settings for connecting to an SQLite database stored in the file “C:\new.newSession.db”

If you need to have different settings in front and back end, you have to edit the configuration file.

Expand the nodes of your *NeSSi²* project Configuration → Recorder Templates → Global Recorder Templates and copy the contents of *default_recorder.xml* to a new file. Comment in the parameters in the appender definition block for *DBAppender* and set the appropriate values. Figure 4.4 shows a sample configuration.

For further information about the back end configuration files refer to the Apache log4j documentation.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">
3 <log4j:configuration xmlns:log4j='http://jakarta.apache.org/log4j/'>
4
5   <appender name="Console" class="org.apache.log4j.ConsoleAppender">
6     <param name="Target" value="System.out"/>
7     <layout class="org.apache.log4j.PatternLayout">
8       <param name="ConversionPattern"
9         value="%dABSOLUTE %-5p [%c1] %m%n"/>
10    </layout>
11  </appender>
12
13  <appender name="DBAppender"
14    class="de.dailab.insin.netshield.recording.types.DBAppender">
15    <param name="driver" value="com.mysql.jdbc.Driver"/>
16    <param name="url" value="jdbc:mysql://localhost/">
17    <param name="schema" value="nessi2"/>
18    <param name="user" value="root"/>
19    <param name="password" value="Start01"/>
20  </appender>
21
22  <logger name=
23    "de.dailab.insin"
24    additivity="false">
25    <level value="DEBUG"/>
26    <appender-ref ref="DBAppender"/>
27  </logger>
28
29  <logger name=
30    "de.dailab.insin.netshield.agents.network.DummyConnection"
31    additivity="false">
32    <level value="DEBUG"/>
33    <appender-ref ref="DBAppender"/>
34  </logger>
35
36  <logger name=
37    "de.dailab.insin.netshield.protocols.core.layer.InternetLayer"
38    additivity="false">
39    <level value="DEBUG"/>
40    <appender-ref ref="DBAppender"/>
41  </logger>
42
43  <logger name=
44    "de.dailab.nessi2.mobileNode"
45    additivity="false">
46    <level value="DEBUG"/>
47    <appender-ref ref="DBAppender"/>
48  </logger>
49
50  <logger name="NetworkLogger"
51    additivity="false">
52    <level value="DEBUG"/>
53    <appender-ref ref="DBAppender"/>
54  </logger>
55 </log4j:configuration>
```

Figure 4.4: recorder_settings.xml for configuring the *NeSSi²* backend

List of Figures

4.1	Simulation Settings	7
4.2	Example settings for connecting to a MySQL database	8
4.3	Example settings for connecting to an SQLite database	9
4.4	recorder_settings.xml	10