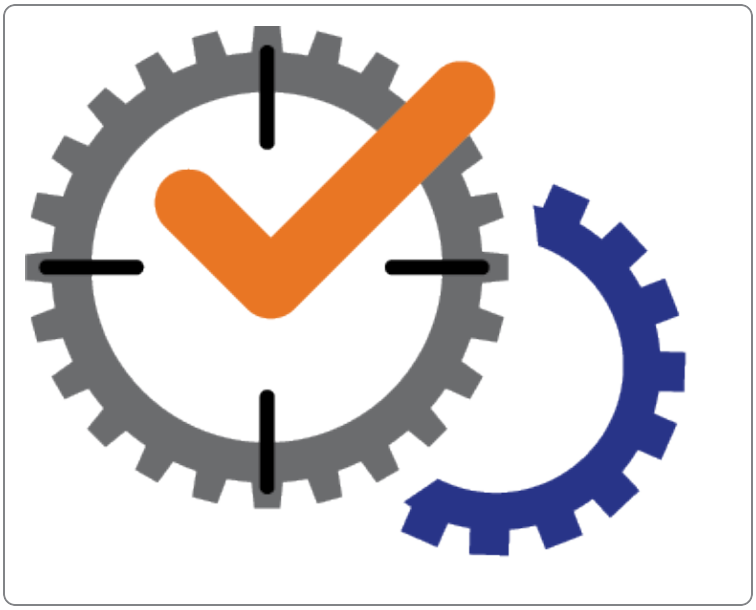
spectracom

PRISMA Compliance®

User Guide



Document Part No.: 1234-5000-0050
Revision: 2.0
Date: 20-Dec-2018

spectracom.com

The information in this document has been carefully reviewed and is believed to be accurate and up-to-date. Spectracom assumes no responsibility for any errors or omissions that may be contained in this document, and makes no commitment to keep current the information in this manual, or to notify any person or organization of updates. This User Guide is subject to change without notice. For the most current version of this documentation, please see our web site at spectracom.com.

Spectracom reserves the right to make changes to the product described in this document at any time and without notice. Any software that may be provided with the product described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of those agreements.

Orolia USA, Inc. dba Spectracom

• 1565 Jefferson Road, Suite 460, Rochester, NY 14623 USA
• 3, Avenue du Canada, 91974 Les Ulis Cedex, France

Do you have questions or comments regarding this User Guide?
➔ E-mail: techpubs@spectracom.com

Warranty Information

For a copy of Spectracom's Limited Warranty policy, see the Spectracom website: http://spectracom.com/support/warranty-information.

Blank page.

CONTENTS

CHAPTER 1

Theory of Operation 1

1.1 Introduction 2

1.1.1 Real-Time Reporting and True UTC Traceability 2

1.1.2 Basic Structure 3

1.1.3 Transmitter Compatibility 4

1.2 Operating Principle 4

1.3 Database Architecture 5

1.3.1 Compliance Server Node IDs 6

1.4 Replication 7

1.5 Licensing 8

CHAPTER 2

Installation 11

2.1 Network and System Requirements 12

2.1.1 Compliance Transmitter Requirements 12

2.1.2 Compliance Server Database Node Requirements 12

2.1.3 Timing Network Requirements 13

2.2 Installation Summary 14

2.2.1 Compliance Servers 14

2.2.2 Compliance Transmitters 14

2.3 Installing a Compliance Server Package 14

2.4 Installing a Compliance Transmitter 18

2.4.1 Linux/Unix Installation 18

2.4.2 Windows Installation 19

CHAPTER 3

Setup 21

3.1 Configuring the Servers 22

3.1.1 Adding Compliance Server Nodes 22

3.1.2 Configuring Nodes within a Cluster 23
3.1.3 Taking Down a Database Node 24
3.1.4 Recovering from a Node Failure 25
3.2 Configuring the Transmitter 25
3.2.1 Compliance Transmitter Commands 26
3.2.2 Scheduling Transmitter Functions 29
3.2.3 Configuring Log Rotation 30
3.2.3.1 Installing ptp4logger 30
3.2.3.2 Installing timelogger 31

CHAPTER 4
Using PRISMA Compliance. 33
4.1 Accessing the Compliance Package 34
4.2 Generating Reports 35

APPENDIX
Appendix 37
5.1 Technical Support 38
5.1.1 Regional Contact 38
5.2 Sample Storage Measurements 38
5.3 List of Tables 39
5.4 List of Images 39
5.5 Document Revision History 40

INDEX

CHAPTER 1

Theory of Operation

The Chapter presents an overview of the PRISMA Compliance, its capabilities, main technical features and specifications.

The following topics are included in this Chapter:

1.1 Introduction 2
1.2 Operating Principle 4
1.3 Database Architecture 5
1.4 Replication 7
1.5 Licensing 8

1.1 Introduction

The **PRISMA Compliance** timing solution offers efficient data collection and logging, resilient storage, flexible data retrieval, and analysis. PRISMA Compliance is a complete and universal software tool to document the ongoing changes in your timing network's fidelity to UTC, covering all timing elements, and thus satisfying current MiFID II, ESMA, MiFIR, and FINRA compliance regulations, as well as internal compliance needs.

Spectracom's PRISMA Compliance solution comprises:

- a QuasarDB® fast database that is designed to manage large amounts of data
- a timing data transmitter software that feeds the relevant timing data from timing network grandmasters and clients to the database
- customized retrieval software allowing the user to access data and generate reports.

1.1.1 Real-Time Reporting and True UTC Traceability

PRISMA Compliance was built from the ground up using a highly scalable, low latency QuasarDB® time series database, capable of collecting and organizing your data in real time. Customer interaction with PRISMA Compliance software is fast and efficient; you can generate a report in seconds, rather than hours.

PRISMA Compliance provides full UTC (Coordinated Universal Time) traceability by collecting and aggregating all relevant information within your timing chain to provide you with the true offset to UTC (beyond the offset from master to client). This traceability to UTC is required by the latest regulations.

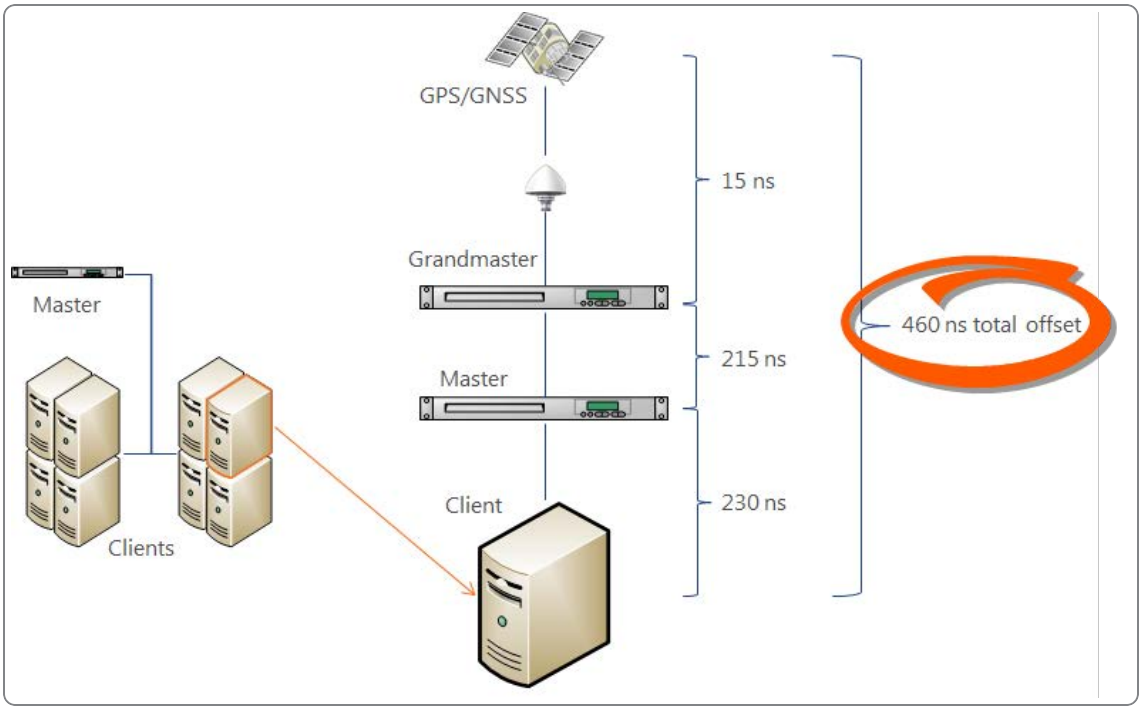
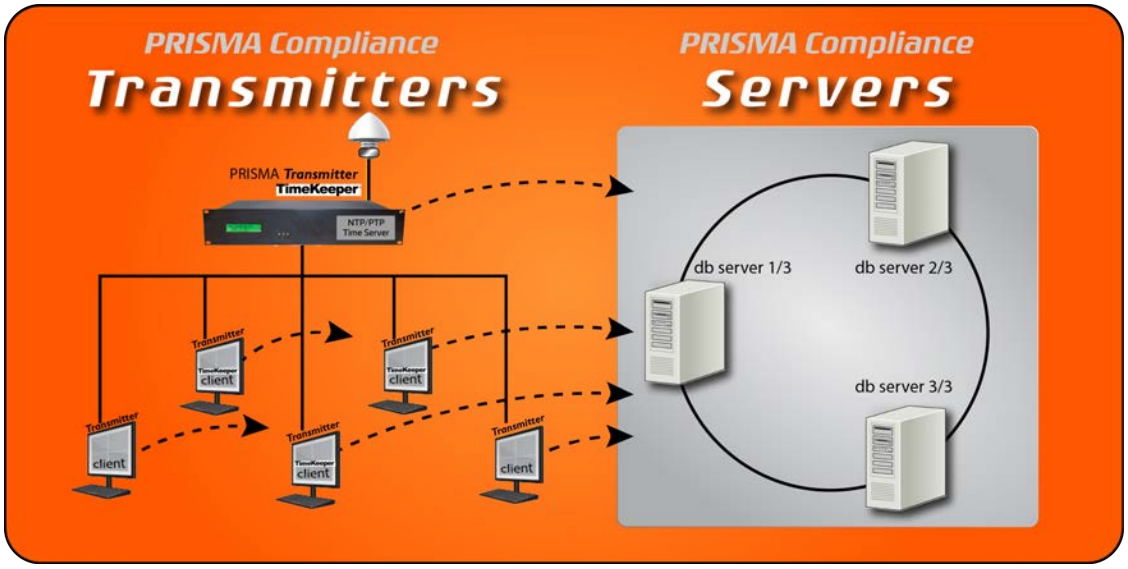


Figure 1-1: Timing chain offset

1.1.2 Basic Structure

Figure 1-2: PRISMA Compliance Transmitters and Servers

- Network timing data is aggregated by the PRISMA Compliance Transmitter installed on timing grandmasters, masters and clients within a local network.
- The Transmitter pushes the captured timing data to a PRISMA Compliance Server where the data is replicated onto other Compliance Servers.
- Users can access the stored data via an interface to generate customized reports (see "Generating Reports" on page 35).



1.1.3 Transmitter Compatibility

PRISMA Compliance Transmitters run on Linux and Windows and are fully compatible with NTPd, Timekeeper, chrony, ptpd, ptp4l, and sfptpd. This makes PRISMA Compliance a drop-in timing compliance solution for most timing networks, whether it's an older legacy NTP network or a new, low latency PTP deployment. See "Network and System Requirements" on page 12 and the "Installation Summary" on page 14 for more information.

1.2 Operating Principle

PRISMA Compliance is comprised of two basic parts. On the receiving end, at least one QuasarDB database **Compliance Server** instance is installed on a server (also referred to as **node**). On the sending end, PRISMA Compliance also uses at least one **Compliance Transmitter** installed on a timing server or client that pushes the logged compliance data to the Server.

For standard applications, you will use two Compliance Servers, to allow for redundancy (a replication factor greater than 1), and one Compliance Transmitter installed for each timing client (to allow for truthful monitoring). There is no theoretical upper limit for the number of time Compliance Server nodes and Compliance Transmitters (while observing the licensing requirements (see "Licensing" on page 8).

Compliance Transmitters are programs that you install in your timing network. The Transmitters collect synchronization data from the timing network on which they are installed, and send this data to the Compliance Server(s), where the data is archived and can be accessed by users.

» The data transmission can be scheduled during off-hours or throughout the day (see "Scheduling Transmitter Functions " on page 29).

- The Compliance Server(s) collect the timing data and store it in the high-performance QuasarDB database.
- A database replication automatically occurs between all Compliance Server nodes, provided they are balanced properly (see "Database Architecture" below).
- Reports are generated from a local application on a Compliance Server, or via an API (see "Generating Reports" on page 35).

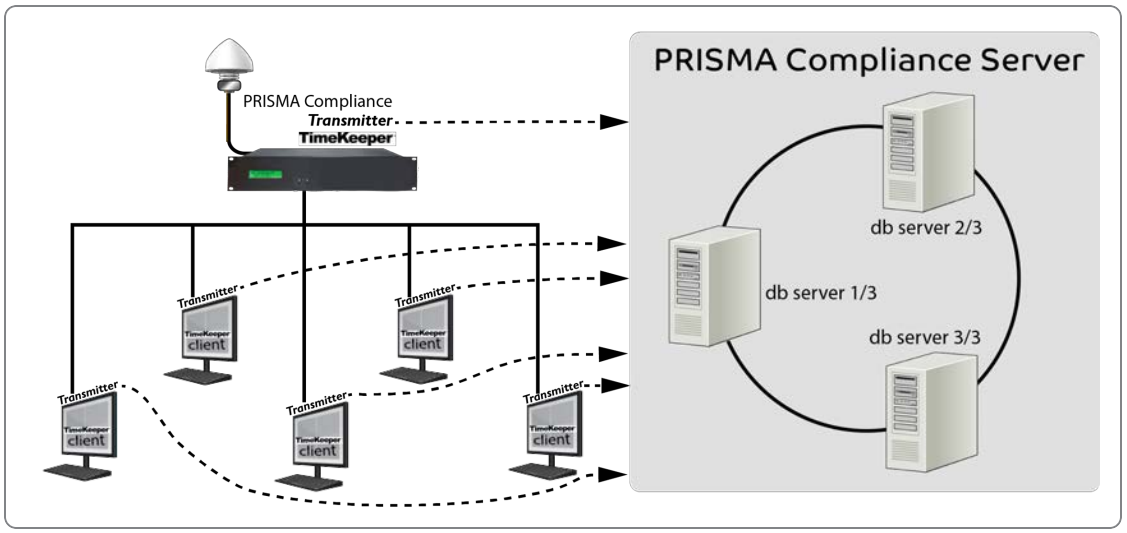


Figure 1-3: Transmitting timing data

1.3 Database Architecture

Compliance Servers store compliance data that has been pushed by the Compliance Transmitter which are installed on timing network clients or servers. A Compliance database Server running a QuasarDB instance is called a database **node**. Nodes can be installed in one geographic site, or across sites in different locations (depending on the license purchased; for more information, see "Licensing" on page 8). While one node will suffice to store all timing data from a monitored network, it is more efficient and safer from a redundancy standpoint to use multiple nodes which are linked together in a **cluster**. In such a peer-to-peer cluster, the database nodes are self-sufficient by sharing data and handling all client requests, thus providing a scalable, concurrent, and fault-tolerant database.

The nodes in a cluster need to be arranged such that they have equal distances to one another (they are "**balanced**"), in order to distribute the data storage load evenly:

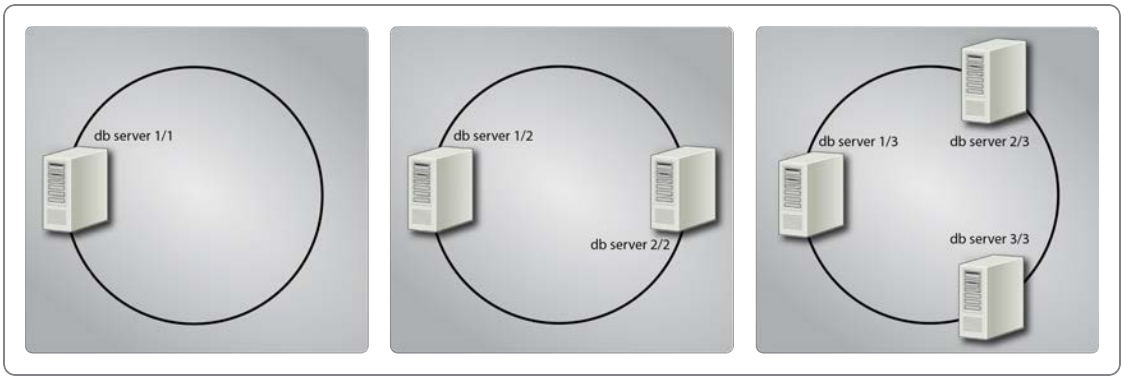


Figure 1-4: Database cluster architectures

Database servers must be indexed by node ID's which look like fractions: The server ID is the numerator and the # of servers in the cluster in the denominator. (For instance, an ID of ¾ corresponds to the 3rd server in a cluster of 4 servers.) For balanced operation, where roughly the same amount of data is installed on each server, it is recommended that each position in the cluster have a server associated with it.

1.3.1 Compliance Server Node IDs

The syntax of the node ID's is: [`current_node`]/[`total_node`] (e.g., 3/8 for the third node of a cluster with 8 nodes).

QuasarDB will assign an indexed node ID automatically, using the relative position of the new node. For example, if you have a cluster with 4 nodes, each node should be assigned the following ID:

- node 1: 1/4
- node 2: 2/4
- node 3: 3/4
- node 4: 4/4

If you want to reserve ID space to allow the cluster to grow to 32 nodes without changing all IDs later, you should assign the following numbers:

- node 1: 1/32
- node 2: 9/32
- node 3: 17/32
- node 4: 25/32

(See also: "Configuring the Servers" on page 22)

1.4 Replication

Regulations in both the US, and Europe require that all relevant logs from reportable events are stored for at least 5 years and can be retrieved when requested. By nature, the Compliance Servers only store each piece of information once; they avoid duplication of information because the Compliance Transmitters do not retrieve data that is recognized as already existing in the node.

To allow for data to be present within storage in more than one place, PRISMA Compliance supports **data replication**, allowing for the logs to be stored on more than one Compliance Server for resiliency purposes. Replication is recommended, as it ensures that data will not be lost if a single Compliance Server node fails. Once replication has been set up and the database cluster has been stabilized, replication occurs automatically between Compliance Servers.

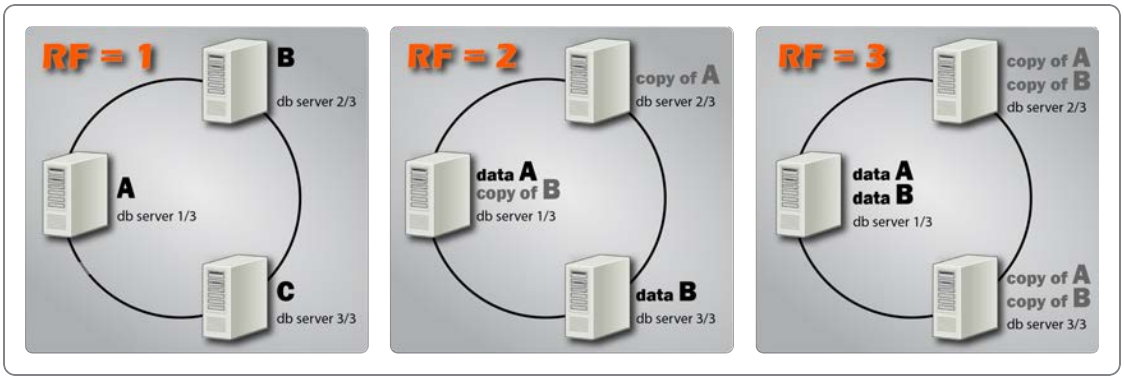


Figure 1-5: Replication principle: No/partial/full replication

The concept of data being stored in more than one place in the database cluster is referred to as replication. A replication factor of 1 means that each piece of data will only be stored on a single node of the cluster (zero redundancy). If data is lost on a single node of the cluster, that data cannot be recovered.

A replication factor of 2 means that each piece of data will be stored on 2 nodes in the cluster. The cluster must have at least 2 nodes in order to support this level of replication. If a single node fails, then once that node is replaced (with the same node ID) it will be re-populated with data to support the necessary replication factor.

Replication factors of 3 and 4 and so on are also supported, with each piece of data being stored on 3 or 4 nodes, respectively.

It is necessary to have at least as many nodes installed as your replication factor setting. To reduce the amount of data on each node, you may also choose to install more nodes than your replicated data, so that each node only holds part of one full copy of the data.

The nodes in a database cluster are arranged in a ring-shaped database (see figure above). If a timing log file is added to node 1/3, it will be replicated to node 2/3 (replication factor = 2), and to 3/3 (replication factor = 3), etc. Thus, replication linearly increases disk and memory usage.

Data that is transmitted to a database with replication enabled will not be made available in reports until replication is complete.

In the event that a node fails or when entries are otherwise unavailable, client requests will be served by the successor node(s) containing the duplicate data. In order for an entry to become unavailable, all nodes containing the duplicate data need to fail *simultaneously*.

In each of the following examples, the network has three nodes (Compliance Servers) installed. The data distribution occurs differently within each node depending on the replication factor. As the replication factor increases, so does the amount of data on each node.

Table 1-1: Replication Factor = 1 (letters represent individual pieces of data)

Node 1	Node 2	Node 3
A		
	B	
		C

Table 1-2: Replication Factor = 2

Node 1	Node 2	Node 3
A	A	
	B	B
C		C

Table 1-3: Replication Factor = 3

Node 1	Node 2	Node 3
A	A	A
B	B	B
C	C	C

For a general idea of data storage sizes, see "Sample Storage Measurements" on page 38.

1.5 Licensing

Licenses are issued by site per organization, with one site being a geographic location (postal address).

A Single Site license covers:

- One database cluster with an unlimited number of Compliance Servers
- An unlimited number of Compliance Clients ("transmitters" within this site. If applicable, the clients can be installed on different networks located at this site

An Enterprise license covers:

- Up to 25 sites

An Unlimited license covers:

» More than 25 sites.

Licenses are issued for one year from date of purchase. Software upgrades are included during the license period.

A click-through renewal reminder will be issued automatically upon login 60, 30, and 15 days before the expiration date. Once a license has expired, you will no longer be able to run the software.

Contact your salesperson or Spectracom Tech Support for license renewal.

BLANK PAGE.

CHAPTER 2

Installation

The Chapter describes the different aspects of PRISMA Compliance installation.

The following topics are included in this Chapter:

2.1 Network and System Requirements 12
2.2 Installation Summary 14
2.3 Installing a Compliance Server Package 14
2.4 Installing a Compliance Transmitter 18

2.1 Network and System Requirements

2.1.1 Compliance Transmitter Requirements

The Transmitter software that is installed on the network timing clients and grandmaster require one of the following timing clients on each node.

Timing networks running NTPd, TimeKeeper, chrony, PTPd, PTP4L, and sfptpd are supported.

CPU:

- Intel Westmere (2010) or better microarchitecture (AMD Bulldozer or better) – database API requirements

OS:

- Linux (deb/rpm/tgz): Requires **glibc 2.12** and x64 (2010) or later
- Windows:
  - **x64** (32-bit not recommended)
  - **Windows 8 or later**; not tested on earlier versions
  - **Windows Server 2012 or later**; not tested on earlier versions

2.1.2 Compliance Server Database Node Requirements

PRISMA Compliance database software can run on virtual servers and physical servers. The ISO installation bundle will install a full CentOS7 version on each server.

Table 2-1: Database node hardware and space requirements

Hardware	Minimum	Typical
CPU	Intel Westmere or better microarchitecture (AMD Bulldozer or better), 2 cores	Intel Westmere or better microarchitecture (AMD Bulldozer or better), 4 cores
HDD	10 GB + Space required per node* + 20 %	10 GB + Space required per node + 30 %
RAM	4GB	32 GB

*Space required per node = (Total size of data in cluster * Replication factor)/Number of nodes
*Your space requirements will vary depending on the amount of data collected and length of installation.

Note: Some real measurements can be found here: "Sample Storage Measurements" on page 38.

If a client has a single source with one record per second, then it would require 1.5 GB to store one year worth of data.

- Size of Client Data in Cluster = ~1.5 GB * Records per second * Years * Sources amount
- Total Size of Data in Cluster = SUM (Size of Client Data in Cluster)

EXAMPLE:

If the requirement is to store 7 years of data for a client with 3 sources (1 record per second), and a server with one source (2 records per second):

Size of Client Data = 1.5 GB * 1 rec/sec * 7 years * 3 sources = 31.5 GB

Size of Server Data = 1.5 GB * 2 rec/sec * 7 years * 1 sources = 21 GB

Total Size of Client Data + Server Data (with a replication factor of 1) = 31.5 GB + 21 GB = 52.5 GB

In this example, the HDD requirements for a single node cluster would be:

- Minimum = 10 GB + 105 GB + 20% = 76 GB
- Typical = 10 GB + 105 GB + 30% = 82 GB

The HDD requirements for a cluster with 4 nodes and a replication factor of 2 would be:

- Minimum = 10 GB + 52.5 GB * Replication Factor of 2 / 4 nodes + 20% = 10 GB + 26.25 GB + 20% = 51 GB per node
- Typical = 10 GB + 52.5 GB * Replication Factor of 2 / 4 nodes + 30% = 10 GB + 26.25 GB + 30% = 55 GB per node

Note: The total amount of data a single grid may handle is 16 EiB (that is 18,446,744,073,709,551,616 bytes).

Information on cluster configuration can be found under "Database Architecture" on page 5.

2.1.3 Timing Network Requirements

- All Compliance Servers (database nodes) must be time synchronized. It is **necessary to have functioning NTP on your network** to achieve accurate records.
- If your system is using Windows, it is necessary to have **Windows 8 and Windows Server 2012 or better** for the operating system to deliver highly accurate timestamps, which prevent transaction conflicts and provide entries with accurate metadata.

2.2 Installation Summary

2.2.1 Compliance Servers

The **PRISMA Compliance** Server software is distributed as an ISO package based on CentOS 7. The ISO will install a full Linux system on your virtual machine or on a physical server. The PRISMA Compliance Server software includes the database software and the Spectracom software that generates the logs. Your license file is delivered separately.

Once the installation is complete, you can then add the appropriate amount of nodes for your network (see "Configuring the Servers" on page 22). For standard applications you need to configure only a few database settings. See "Installing a Compliance Server Package" below for more information.

2.2.2 Compliance Transmitters

PRISMA Compliance Transmitters are installed on the network timing clients, masters and grandmasters. Available for free download as a companion to the Compliance Server, Compliance Transmitter files gather timing data in your network. You will install the Transmitter that is designed for your timing setup, and enter settings to ensure data collection. See "Installing a Compliance Transmitter" on page 18 and "Configuring the Transmitter" on page 25 for more information.

For Technical Support contact information see "Technical Support" on page 38.

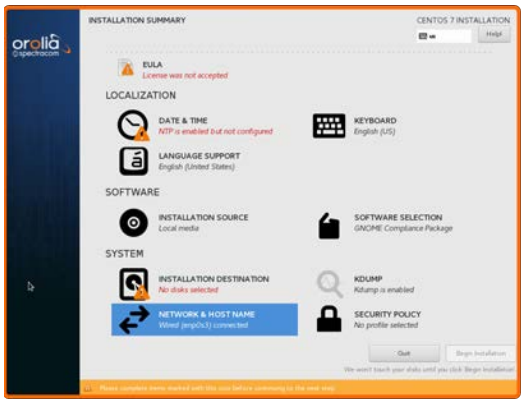
2.3 Installing a Compliance Server Package

Note: The following instructions are designed to apply to installation of both your first Compliance Server (database node) and your additional servers. If you plan on installing multiple Compliance Servers, you must install, configure, and start the first server prior to installing additional servers. See "Configuring the Servers" on page 22 for more information.

The Compliance Package is provided as an ISO file. The ISO file can be installed on a virtual machine (VM), or on bare metal hardware. Attach a hard drive of sufficient size.

1. Verify that your network, and your server meet the minimum requirements outlined under "Network and System Requirements" on page 12.
2. Prepare the target device to boot from the provided ISO file. For VM's, attach the ISO directly to the virtual machine instance. For physical servers, write the ISO to media the server can boot from.

3. Boot the physical machine or VM.

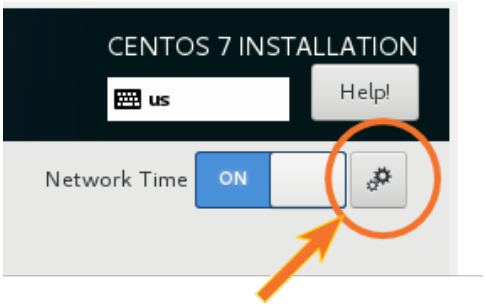


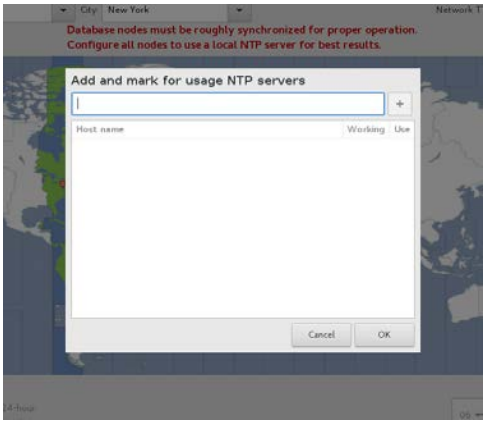
   - » Before continuing the installation process, you must resolve the red-highlighted sections

4. Click **DATE & TIME** and select the current timezone.

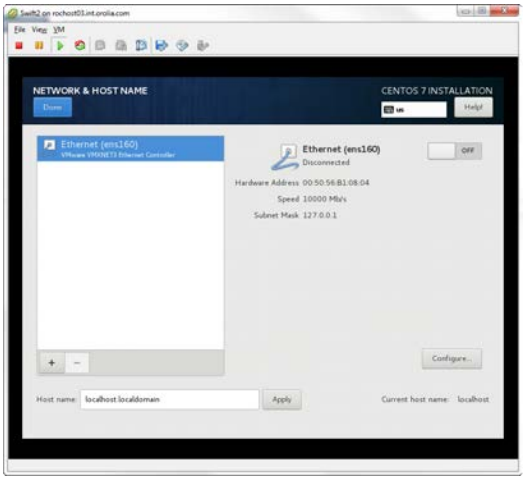
Connect to your NTP system

   - » NTP is required to achieve rough synchronization of multiple compliance servers, but is not used for compliance tracking.
   - » The node must be configured with at least one active NTP server. **For best results, a user-installed local NTP server is recommended**.
   - » In the **DATE & TIME** submenu, click the gear icon next to the **Network Time** toggle. Enter your NTP server information.



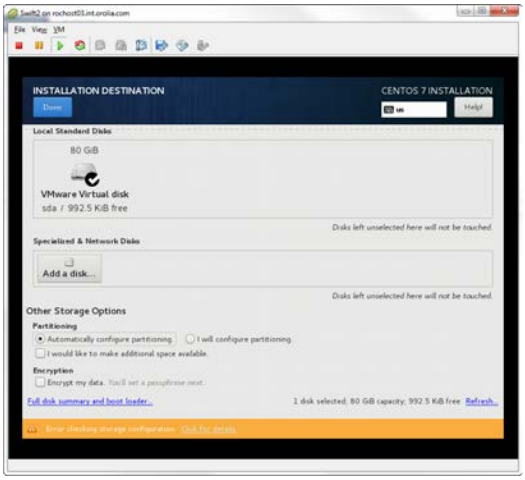


5. Click **NETWORK & HOSTNAME** to enable networking.



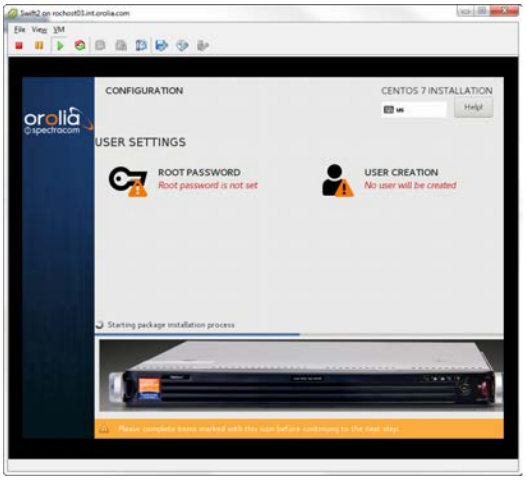
- Click **Switch** to enable Ethernet
- Enter hostname and click **Apply**.
- Click **Done**.

6. Click **INSTALLATION DESTINATION**.



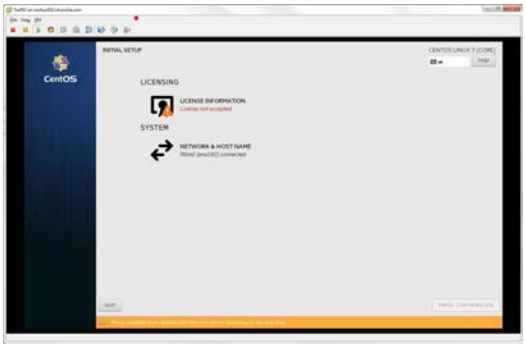
   - » Ensure that the proper disk is clicked by default.
   - » Select **Automatically configure partitioning** to use the whole disk.

7. Other settings (keyboard, language support, installation source, software selection, kdump, security policy) do not need to be changed.
8. Click **Begin Installation**.
9. While the install is in progress, you can create the main user logins:



   We recommend adding a user as an "administrator". This user will have sudo access, hence there will be no need for a root password.

10. Reboot the machine, making sure to boot from the hard drive this time.
11. On reboot, you will see an **INITIAL SETUP** page where in order to proceed you need to click on **License information** and acknowledge the EULA:



12. Click **FINISH CONFIGURATION** and boot into the VM itself.

2.4 Installing a Compliance Transmitter

Compliance Transmitter is the client application that is installed in networks that are to be monitored using the PRISMA Compliance system. They extract the log files of the timing source and push timing data into the database. The Compliance Transmitter package is available to download from our website and can be installed on either a Linux/Unix system or on Windows.

Following installation, it will be necessary to configure your Compliance Transmitter to communicate with your timing log files (see "Configuring the Transmitter" on page 25).

2.4.1 Linux/Unix Installation

Navigate to https://files.spectracom.com and copy the link for the file that corresponds to your Linux distribution. Or, use the links below. You can download using the `wget` tool, or download to a different system using your preferred method (ftp/sftp, USB, cd, etc.) to transfer the file.

Download the .rpm package using wget (example):

```
$ sudo wget https://files.spectracom.com/latest-files/-
direct/prisma_compliance_transmitter_rpm_current.tar.gz
```

- **RPM**: To install in Red Hat-based distributions (.rpm):

  https://files.spectracom.com/latest- files/direct/prisma_ compliance_ transmitter_ rpm_current.tgz

  example command for version x.x.x installation:

  - `$ sudo rpm --install prisma_compliance_transmitter_ rpm_x.x.x.tgz`

- **DEB**: To install in Debian-based distributions (.deb):

https://files.spectracom.com/latest-files/direct/prisma_compliance_transmitter_debian_current.tgz

example command for version x.x.x installation:

```
$ sudo dpkg --install prisma_compliance_transmitter_debian_x.x.x.tgz
```

- **General Linux**: To install in any other standalone Linux/Unix based system, including time servers, such as PRISMA VelaSync:

  https://files.spectracom.com/latest-files/direct/prisma_compliance_transmitter_linux64_current.tgz

  example command for version x.x.x installation:

  ```
  $ sudo tar --extract prisma_compliance_transmitter_linux64_x.x.x.tgz
  ```

2.4.2 Windows Installation

Navigate to https://files.spectracom.com and download the Windows installer file.

- Or, a direct link can be found here:

  https://files.spectracom.com/latest-files/direct/prisma_compliance_transmitter_win64_current.zip

- The Windows installer installs all files in the "C:\Program Files (x86)\Orolia\compliance transmitter" directory. The files are:
  - settings.cfg : the Compliance Transmitter configuration file
  - transmitter.exe : the Windows Compliance Transmitter executable
  - transmitter.bat : a short batch file that calls transmitter.exe and then pauses to let you analyze the output
  - uninstall.exe : the uninstaller for the Windows Transmitter
- Paths can be listed as either Unix-style paths (/tmp/foo) or Windows-style paths (C:\tmp\foo).
- Once the settings file is correct, runing transmitter.exe (or transmitter.bat) will read the logfiles once and send them to the Compliance Server in the settings.config file.

BLANK PAGE.

CHAPTER 3

Setup

The Chapter describes how to configure PRISMA Compliance for your system needs.

The following topics are included in this Chapter:

3.1 Configuring the Servers 22
3.2 Configuring the Transmitter 25

3.1 Configuring the Servers

Following initial installation of your first Compliance Server, it is necessary to design your system to your specifications.

This section is designed to help you structure your Compliance software effectively, with the appropriate amount of and names for Compliance Servers (also referred to as database nodes) within your cluster, and your ideal replication factor.

Note: If you plan on installing several Compliance Servers, you must install, configure, and start the first server prior to installing additional servers. See "Installing a Compliance Server Package" on page 14 for more information.

Note: For more information on the replication practice with QuasarDB, see "Database Architecture" on page 5, "Compliance Server Node IDs" on page 6, or "Replication" on page 7.

Note: It is advisable to have a homogenous hardware configuration within a cluster.

3.1.1 Adding Compliance Server Nodes

You can add (and remove) Compliance Servers. There is no practical limit as to how many servers can be installed. There are two recommended methods for adding new servers to preserve balanced operation:

1. You can double the number of servers by multiplying the numerator and denominator of the existing server ID's by 2, and then adding servers to fill in the gaps. Existing servers do not have to be rebooted.

   EXAMPLE:

   An existing cluster has 1/3, 2/3, 3/3 nodes. These nodes are equivalent to 2/6, 4/6, and 6/6. New nodes can be added at 1/6, 3/6, and 5/6.

2. You can add servers by changing the "number of servers" in the denominator (see "Configuring Nodes within a Cluster" on the facing page), but first existing database daemons should be shut down one at a time in order to apply new ID's.

EXAMPLE:

An existing cluster has 1/3, 2/3, 3/3 nodes. These nodes are powered down, one at a time, and re-numbered to 1/5, 2/5, and 3/5. After that, new nodes can be added at 4/5 and 5/5.

If a node attempts to join a cluster and a node with a similar ID is found, the new node will exit the cluster.

To add a Compliance Server database node, follow the steps in "Installing a Compliance Server Package" on page 14 **again on another instance of virtual machine or physical server**. Then follow the steps in "Configuring Nodes within a Cluster" below.

3.1.2 Configuring Nodes within a Cluster

Identify your ideal cluster architecture:

1. Determine the desired **replication factor** for your data
   - » Can be 1-4 (recommended setting is at least 2)
   - » Must be less than or equal to your number of nodes.
2. Decide how many **database nodes** (Compliance Servers) you want overall
3. Then, determine the names of the nodes based on (**node # / total # of nodes**).
   - » For instance, if you want 3 nodes total, the node IDs will be "1/3", "2/3", "3/3".
   - » If you want to leave space for your system to grow, you can name the first nodes with a larger total number of nodes (denominator).
   - » It may be necessary to take down your first-installed node briefly in order to rename with your new naming convention.

Add nodes, adjust node ID's, and set your replication factor:

4. While connected to the new server location, follow the steps in "Installing a Compliance Server Package" on page 14 to **add a database node**.
5. On the Compliance Server that you need to configure or add to your cluster, access the console of the server or VM. Note the IP address of the node.
6. Copy the license file to the node (scp is recommended).
7. **Apply the license file** by executing "`sudo compliance-license-installer ./<license_filename>`" in a command shell.
8. As the superuser, edit the `/etc/qdb/qdbd.conf` file to **adjust the node name**:
   - » Add the Node ID to the "Node ID" line:
     - » example: `"node_id": "2/3",`
   - » If this is not the first node, add the IP Address and Database Port of any other database node:

- example: `"bootstrapping_peers": ["10.10.128.12:2836"],`
- this will connect a new node to the cluster

9. As the superuser, edit the `/etc/qdb/qdbd.conf` file to **adjust the replication factor**:
   - example: `"replication_factor": 2,`

Finalize your changes:

10. **Restart the database** with the new parameters:
    - `sudo service qdbd restart`
11. If there are problems, the qdbd log file can be found at `/var/log/qdb/qdbd.log`. That file can only be read by the superuser.

3.1.3 Taking Down a Database Node

Note: It is possible to take down a database server temporarily e.g., to change its node ID (which will be necessary if you want to add an additional server). Note, however, that any reports generated during this down time will be incomplete if your replication factor is 1.

Caution: When a node is removed from the database, other nodes will reorganize the data, so you must account for some additional space on those nodes. The space requirements can be computed depending on the network configuration and the expected amount of data of the machines removed from the cluster.

Once you remove a node via a clean shutdown, it will inform the other nodes in the ring during the shutdown. The removal of the node will result in another node storing more data until the missing node is replaced. If a node disappears due to technical failure, the cluster will detect the failure during the next periodic balance verification check, and then automatically restabilize the cluster.

Entries are not migrated when a node leaves the cluster, only when a node enters the cluster.

To shut down a node, use one of the following methods:

- shutdown the VM the node (Compliance Server) is installed on
- Use the "service" commands:
  - `sudo service qdbd start`
  - `sudo service qdbd stop`
  - `sudo service qdbd restart`

3.1.4 Recovering from a Node Failure

When a node recovers from failure, it needs to reference a node within the ring to rejoin the cluster. The configuration for the first node in a ring generally does not reference other nodes, thus, if the first node of the ring fails, you may need to adjust its configuration file to refer to an operational node.

If following a major network failure, a cluster forms two disjointed rings, the two rings will be able to unite again once the underlying failure is resolved. This is because each node "remembers" past topologies.

The detection and re-stabilization process surrounding node failures can add a lot of extra work to the affected nodes. Frequent failures will severely impact node performance.

A node can also "fail" if it is fully saturated, as it will no longer be able to accept and store data.

Note: A cluster operates best when more than 90% of the nodes are fully functional. Anticipate traffic growth and add nodes before the cluster is saturated.

Note: The information above is an excerpt of the online QuasarDB documentation, where you can also find additional instructions, if needed: https://doc.quasardb.net/2.0.0/concepts/data_storage.html#data-migration.

3.2 Configuring the Transmitter

Following installation, it is necessary to configure the Compliance Transmitter to identify your timing log files. The method for this configuration varies with the timing sources in your network (see "Compliance Transmitter Commands" on the next page).

For certain systems, you may also need to set up the Transmitter to check your rotated log files for uncaptured data (see "Configuring Log Rotation" on page 30).

It is also recommended to schedule your Transmitter's functionality (see "Scheduling Transmitter Functions " on page 29).

A Compliance Transmitter will communicate with the log files from the following timing sources:

- **ntpd**
- **chrony**
- **ptpd**
- **sfptpd**
- **ptp4l** (requires additional **ptp4logger** utility)

- **Windows Time Service** (requires additional **timelogger** utility)
- **Timekeeper**

The Compliance Transmitter is configured using a `compliance-transmitter.cfg` file that is included in the installation process (for Windows, this file is `settings.cfg`).

Depending on your installation process, the settings file may be in one of the folders `/etc` , or `/opt`, or in the directory where the Compliance Transmitter is also installed.

The transmitter will function as long as the application has access to the clients' log files.

- If ntpd is used, it must be configured to write out the `peerstats` log.
- If ptp4l is used, the ptp4logger application must also be used (see "Installing ptp4-logger" on page 30).
- If the Windows Time Service is used, the timelogger appplication must be used (see "Installing timelogger" on page 31).

3.2.1 Compliance Transmitter Commands

Settings are organized in groups. Each group has a **Group Name**, followed by a list of settings. Customers should make sure the Group Name is uncommented if they want to use any settings from that group.

Settings are disabled by commenting them out using "#".

To use the default values for all settings, uncomment a Group Name, but leave all settings commented.

The following commands can be used to configure the `compliance- transmitter.config` file (`settings.cfg` for Windows):

- `sources:` — configures the source of timing data on that node
  - `timekeeper:` uncommenting this enables timekeeper log parsing
    - `mode: transmit clients data`
      - If this is not commented, the transmitter will parse the `timekeeperclients` folder and transmit information on all clients connected to this Timekeeper instance. Most useful on Timekeeper-based masters, like a Timekeeper-based Velasync.
    - `log path: /path/to/logs`
      - If this is not commented, Timekeeper logs will be taken from the specified path (if it is commented, logs will be taken from the default `/var/log` path).
  - `ntpd:` uncommenting this will enable NTP log parsing
    - `log path: /path/to/logs`

- If this is not commented, ntpd logs will be taken from the specified path (if it is commented, logs will be taken from the default path for chrony).

- `chrony`: uncommenting this will enable chrony log parsing
  - `log path: /path/to/logs`
    - If this is not commented, chrony logs will be taken from the specified path (if it is commented, logs will be taken from the default path for chrony).
- `ptpd`: must be configured to store **ptpd2.stats** and **ptpd2.status**.
  - A typical ptpd config file:

    global:log_statistics=y

    global :statistics_file=/var/log/ptpd2.stats

    ; log file in KB

    global:statistics_file_max_files=5

    global:log_status=y

    global:status_file=/var/log/ptpd2.status
  - `log pathj: /var/log`
    - If this is not commented, ptpd logs will be taken from the default path of `/var/log`.
  - experimental:
    - Should only be used with version 2.3.1 to work around a bug in reporting.
- `solarflare ptp`:
  - Solarflare PTP should be configured to store the Statistics log as `/var/log/sfptpd/log` using the following line in the solarflare PTP configuration: stats_`log/var/log/sfptpd.log`. After log rotation, `sfptpdctl logrotate` should be called to cause sfptpd to reopen log files.
  - `log path: /var/log`
    - If this is uncommented, logs will be taken from the default path `/var/log`. This should match the path in the `stats_log` directive in the config file.
- `ptp4l`:
  - ptp4l should be used with the ptp4logger application (see "Installing ptp4logger" on page 30).

- `log path: /var/log`
  - ptp4logger logs will be taken from this path.
- `w32time:`
  - `w32time` should be used with the timelogger application (see "Installing timelogger" on page 31
  - `log path: /var/log`
    - timelogger data will be taken from this path. If the path is commented out, data will be taken from the default path, C:\Program Files(x86)\Orolia\timelogger\Logs

- `database:` — configures the connection to the database
  - `address: x.x.x.x:2836` — must be set to the IP address of one of the database nodes. Note the database IP port of 2836.
    If there are multiple database nodes desired, leave `address:` on its own line and add lines underneath it, like this:

```
address:
    - 10.32.1.24.2836
    - 10.32.1.25:2836
```

- `debug:` — configures the path of the transmitter log file
  - `log path: /tmp/compliance-transmitter.log`
    - If this is uncommented, logs will be written to the specified file. The log file path must be writeable by the transmitter.
- `device:` — configures device hostname
  - The transmitter will try to assign a hostname by doing a DNS lookup on the current host's IP address, but if that fails, it will simply use the IP address directly (or, in the case of ptpd, it may use the PTP Port Identity instead.) If that is not desired, the device hostname may be specified here.
  - `identity: hostname`
    - assigns the hostname.
- `interfaces:` — configures miscellaneous network settings
  - `ignore:`
    - Is set to interfaces that can be safely ignored. This is set to multiple interfaces, just like the `address` setting. It is recommended to leave this setting enabled with the default settings of `docker0` and `virbr0`.

Transmitter configuration file (example):

```
sources:
   timekeeper:
      #mode: transmit clients data # uncomment to transmit TK
clients data
      #log path: /tmp/tk_logs # if not set, then will be used
system default
   #ntpd:
      #log path: /tmp/ntpd_logs # if not set, then will be used
default /var/log/ntpstats
   #chrony:
      #log path: /tmp/chrony_logs # if not set, then will be
used default /var/log/chrony

database:
   address: 10.10.128.25:2836 # if not set, then will be used
default "127.0.0.1:2836"
      #address: # example with multiple nodes
   # - 127.0.0.1:2836
   # - 10.32.1.24:2836
debug:
   log path: /home/admin/compliance-transmitter.log # if not
set, then will be used system default
interfaces:
   ignore:
      - docker0
      - virbr0
```

Figure 3-1: Example Compliance Transmitter configuration file

3.2.2 Scheduling Transmitter Functions

In an unconfigured state, the Compliance Transmitter operates once, then stops. In order to send data automatically and periodically, it is necessary to schedule a repeating task to run the Transmitter.

Transmitters will find out how much data is already in the database and will only send new data. Data will persist in the database even if the original log files they were based from are rotated out (see "Configuring Log Rotation" on the next page).

You can set up the Compliance Transmitters to transmit on a schedule that is appropriate for your network. For example:

- You can transmit frequently to make sure data is copied to the database often.
- Or, you can wait until the end of the day so that PRISMA Compliance traffic can be limited on the network to appropriate low-volume times.

On Linux systems, a good tool for this scheduling is **cron**.

On Windows, scheduling can be done via the **Windows Task Scheduler**

3.2.3 Configuring Log Rotation

In addition to the **log files** themselves, the Transmitter will also look for **rotated log files** and automatically add necessary files to the database.

Timekeeper, ntpd, and chrony all do their own log rotation and the Compliance Transmitter will look for those files and will include in the database storage any new data within them that has not been transmitted.

However, ptpd, sfptpd, and ptp4l do not automatically perform log rotation and users should install an additional log rotation package, depending on your timing source (such as logrotate on linux). Rotated logs should use the format <logfile>.XX (or .XX.gz for compressed logs), where XX is a one or two digit number.

- For solarflare ptp only, after log rotation "`sfptpdctl logrotate`" should be called to cause sfptpd to reopen the log files for writing.

3.2.3.1 Installing ptp4logger

When using ptp4l/linuxptp with the PRISMA Compliance software, more information is required than the standard ptp4l log behavior, in order to fully track the timing data. The ptp4logger application obtains this information from the UNIX domain socket at /var/run/ptp4l.

ptp4logger can be configured entirely form the command line, if desired:

- [`c FILE_PATH`] configuration file path
- [`domain DOMAIN`] ptp domain
- [`-path PATH`] path to store logs
- [`h | --help |-?`] show help

If a config file is used, there are only two config options:

`domain=x`

- sets the PTP domain. This is the same setting as the –domain switch on the command line.

`log_path=/var/log`

- sets the path to store the ptp4logger logs. This is the same setting as the –path switch on the command line. This should match the "log path:" setting of the "ptp4l:" entry in the Compliance Transmitter configuration file.

ptp4logger should be run continuously whenever ptp4l is running.

3.2.3.2 Installing timelogger

In order to fully track timing and traceability data while using the Windows Time Service, the timelogger application must be used. Timelogger obtains required information from the Windows Time Service and stores it in a log file to make it available to the PRISMA Compliance transmitter.

The timelogger utility is installed automatically at the same time as the Windows PRISMA Compliance Transmitter installation. The file can be found in the C:\Program Files (x86)\Orolia\timelogger directory. Upon installation, a Windows service named **Time Logger** is automatically started which will run the utility in the background. The timelogger utility should be run continuously. Logfiles are stored in C:\Program Files (x86)\Orolia\timelogger\Logs

Should it be necessary to uninstall either the timelogger package or the PRISMA Compliance Transmitter for Windows, uninstallers are located in the install directories of each utility. Logfiles will not be removed if timelogger is uninstalled.

BLANK PAGE.

CHAPTER 4

Using PRISMA Compliance.

The Chapter describes how user can access the software, and how to generate reports.

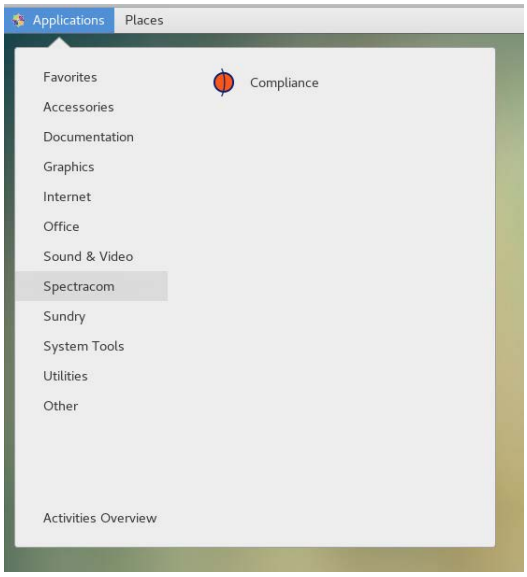
The following topics are included in this Chapter:

4.1 Accessing the Compliance Package 34
4.2 Generating Reports 35

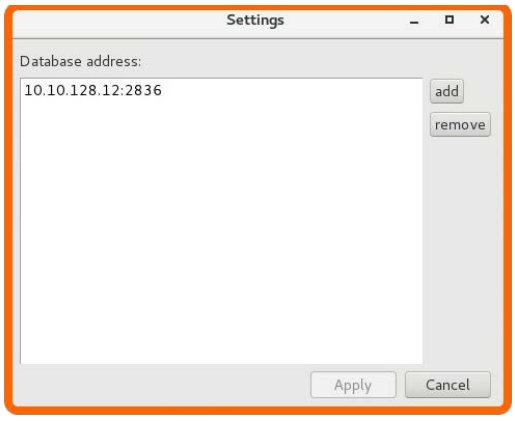
4.1 Accessing the Compliance Package

For the average user to generate reports, it is necessary to log into the Compliance interface.

1. Navigate to the instance of the Compliance Package's Centos Linux machine installed on your system.
2. Log in to the software using your previously created password (see "Installing a Compliance Server Package" on page 14).
3. Select **Applications** > **Spectracom** > **Compliance**



4. When the client is run for the first time, it must be configured with the path to the database. Under **File > Settings**, include the IP address of any of the nodes in the database in the file. (Include the port of the database, too: normally, it is **2836**).



4.2 Generating Reports

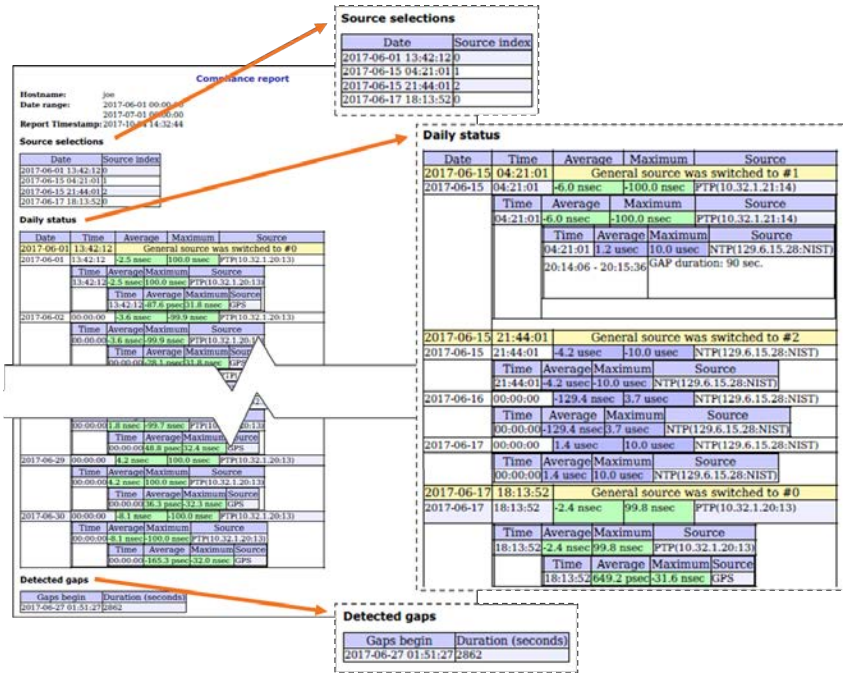


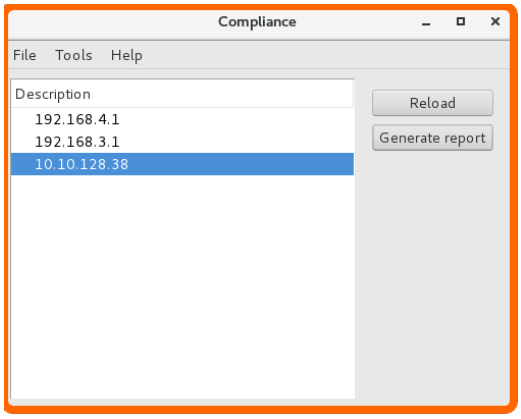
Figure 4-1: Compliance reporting

PRISMA Compliance supports the generation of different report types:

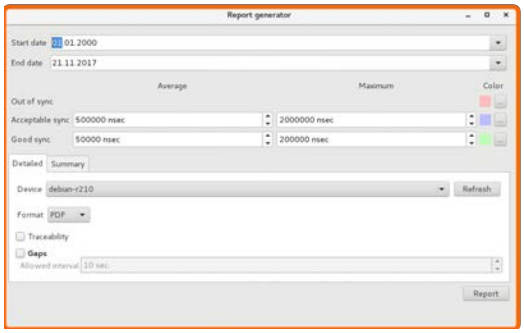
- Consolidated Statistics – Mean, Std Dev, Max
- Adjustable Periods – Year, Month, Day, etc.
- Client Traceability – Complete trace from selected client to UTC source
- Data Gaps – Missing data reporting

To Create a Report:

1. To operate the Compliance client, you must be logged into the console GUI of any compliance database node (see "Accessing the Compliance Package" on the previous page).
2. Under **File** > **Settings**, you will see client nodes listed by Hostname, or if Hostname is not available, by IP address



3. To generate a report on a selected node, click the **Generate Report** button. The report generator will pop up.



4. Fill out all desired fields:
   - » Set the **Start date** and **End date**.
   - » Assign thresholds for **Acceptable sync** and **Good sync**. (Data that exceeds the **Acceptable sync** threshold is considered "out of sync".)
   - » You can also change the colors of all 3 categories in the report.
   - » When **Traceability** is checked, the report will also include data on any upstream devices that the compliance package has information on, and combine the data from the entire chain. If the entire chain goes back to a device that is synchronized to UTC, the report will have full traceability.
   - » When **Gaps** is checked, the report will also indicate if there is missing data.
5. Click **Report**.
6. Determine your end file name and destination, and click **Save**.

APPENDIX

Appendix

The following topics are included in this Chapter:

5.1 Technical Support 38
5.2 Sample Storage Measurements 38
5.3 List of Tables 39
5.4 List of Images 39
5.5 Document Revision History 40

5.1 Technical Support

To request technical support for your PRISMA Compliance software, please go to the "Support" page of the Spectracom Corporate website:

- submit a support request
- find additional technical documentation

Please include information about your system configuration and database setup.

Phone support is available during regular office hours under the telephone numbers listed below.

Thank you for your cooperation.

5.1.1 Regional Contact

Spectracom operates globally and has offices in several locations around the world. Our main offices are listed below:

Table 5-1: Spectracom contact information

Country	Location	Phone
China	Beijing	+86-10-8231 9601
France	Les Ulis, Cedex	+33 (0)1 6453 3980
USA	Rochester, NY	+1.585.321.5800

Additional regional contact information can be found on the Contact Us page of the Spectracom corporate website.

5.2 Sample Storage Measurements

The table below provides some real world data storage measurements for a setup with a **replication factor of 1**. Note that these numbers are the result of one single test, and hence are intended only as examples.

The QuasarDB database will periodically **trim** its contents during low-usage durations. This results in data size reduction; not because information is being lost, but because the database is compacting the data more economically. So the ultimate size of data is better represented after trimming.

An average use case follows, with expected data storage requirements:

Table 5-2: Real-world data: average

Description	Initial size of data in cluster	Total size of data after trimming
1 source, 1 record per second, 1 year Transfer 60k records per request	2.8 G	508 M

The following data represents a worst-case, in which the source name is different for each transaction, and estimated offsets are stored (complex timing systems, such as TimeKeeper, will produce data in these quantities):

Table 5-3: Real-world data: large

Description	Initial size of data in cluster	Total size of data after trimming
1 source, 1 record per second, 1 year Transfer 100k records per request	5.5 G	1.5 G
2 sources, 1 record per 2 seconds, 1 year Transfer 100k records per request	3.8 G	1.5 G
1 source, 1 record per second, 10 years Transfer 100k records per request	50 G	15 G

5.3 List of Tables

Table 1-1: Replication Factor = 1 (letters represent individual pieces of data) 8
Table 1-2: Replication Factor = 2 8
Table 1-3: Replication Factor = 3 8
Table 2-1: Database node hardware and space requirements 12
Table 5-1: Spectracom contact information 38
Table 5-2: Real-world data: average 38
Table 5-3: Real-world data: large 39

5.4 List of Images

Figure 1-1: Timing chain offset 3
Figure 1-2: PRISMA Compliance Transmitters and Servers 3
Figure 1-3: Transmitting timing data 5
Figure 1-4: Database cluster architectures 6
Figure 1-5: Replication principle: No/partial/full replication 7
Figure 3-1: Example Compliance Transmitter configuration file 29
Figure 4-1: Compliance reporting 35

5.5 Document Revision History

Rev	ECO	Description	Date
1.0		Initial release of the PRISMA Compliance User Guide.	Jul 2018
2.0		Added transmitters, errata updates	Dec 2018

INDEX

A

Access Compliance interface 34
Add node 23
Add server 23

B

Balanced architecture 5

C

chrony 27
Cluster 5
Commands, Compliance Transmitter 26
Compliance server 14
Compliance Server Installation 14
Compliance transmitter 14, 25
Compliance Transmitter Commands 26
Compliance Transmitter Installation 25
Configuration file, transmitter 29
Configuring Log Rotation 30
contact, Spectracom 38

D

data replication 7
Data size sample 38
debug 28

G

Generate report 36

H

HDD, requirements 13
hostname 28

I

ISO file 14

L

license renewal 9
License, Enterprise 8
License, Single Site 8
License, Unlimited 8
Linux/Unix Installation, transmitter 18
Log Rotation, Configuration 30

Login, how to 34

M

Multiple Nodes 23

N

network, requirements 13
Node 5
Node failure, recovery 25
Node ID 6
Node name, adjust 23
Node, configuring 23
Node, requirements 12
Node, taking down 24
ntpd 26

P

ptp4l 27, 30
ptp4logger 30
ptpd 27

R

Real-time reporting 2
Remove node 24
Replication 7
Replication factor 7
Replication factor, adjust 24
Reports 35
Requirement 12
Requirements 12

S

Sample Storage Measurements 38
Scheduling 29
Server Installation 14
Server package 17
Server, requirements 12
solarflare 27
sources 26

T

Technical support 38
timekeeper 26
timelogger 31
timing, requirements 13
Transmitter, compliance 18, 25
transmitter, configuration 25
Transmitter, download 18
Transmitter, requirements 12
Transmitter, schedule 29

U

User access 34
user login 17
User login 34
UTC traceability 2

W

w32time 28
Windows Installation, transmitter 19