**Пошаговое руководство по установке**

Развертывание в Linux

Откройте терминал и выполните следующие команды:

Копировать

sudo rpm -ivh timebeat<version-number/full-file-name>.rpm

Если операция выполнена правильно, вы должны увидеть на терминале примерно такой же вывод, как показано ниже.

Копировать

Verifying... ################################# [100%]

Preparing... ################################# [100%]

Updating / installing...

1:timebeat-1.2.1-1 ################################# [100%]

Это всё, что нужно для установки агента Timebeat. Теперь нам нужно настроить файл конфигурации, чтобы ваше приложение Timbeat могло получать данные в формате UTC и отправлять их в Elasticsearch. Для этого нам нужно отредактировать следующий файл:

sudo nano /etc/timebeat/timebeat.yml

Для получения подробной информации о конфигурации, пожалуйста, ознакомьтесь с руководствами по настройке.

Предположим, вы настроили файл timebeat.yml и поместили его в нужную папку:

/etc/timebeat/timebeat.yml      
Далее нам нужно убедиться, что ваш лицензионный файл находится в правильном месте. Мы рекомендуем размещать его в следующем месте:

/etc/timebeat/timebeat.lic

Хотя это расположение необязательно, оно используется по умолчанию в файле конфигурации timebeat.yml. Поэтому, если вы размещаете лицензию в другом месте, не забудьте обновить путь в файле конфигурации.

После завершения работы лицензия будет доставлена на ваш почтовый ящик в считанные секунды.

Если ваши зависимости настроены и файл конфигурации обновлен, вы готовы запустить службу Timebeat. Мы рекомендуем запускать службу на переднем плане после внесения любых изменений в конфигурацию. Выполнение этой команды позволит выявить любые ошибки, которые могут присутствовать в файле конфигурации, чтобы вы могли быстро внести изменения и перезапустить службу.

sudo /usr/share/timebeat/bin/timebeat -c /etc/timebeat/timebeat.yml -path.home /usr/share/timebeat -path.config /etc/timebeat -path.data /var/lib/timebeat -path.logs /var/log/timebeat -e

Как только вы убедитесь, что служба работает должным образом, вы можете запустить её

sudo service timebeat start

Чтобы убедиться, что Timebeat запущен, вы можете проверить его состояние с помощью:

sudo service timebeat status

На этом этапе также рекомендуется выполнить следующую команду, чтобы включить Timebeat при запуске системы и не остаться без синхронизации при перезагрузке системы.

sudo systemctl enable timebeat

Развертывание в Windows

Предположим, что вы загрузили программу Timebeat в каталог на локальном устройстве и можете работать с правами администратора:

Распакуйте содержимое архивированного файла Timebeat

Скопируйте распакованную папку и поместите ее в Program Files

Откройте окно PowerShell от имени администратора и измените каталог на Program files

Здесь вам может потребоваться изменить политику выполнения в зависимости от текущей конфигурации Windows. Чтобы запустить установщик Timebeat, вам нужно настроить политику выполнения на «Без ограничений».  
Выполните следующую команду в папке Program Files

Set-ExecutionPolicy Unrestricted

При правильном выполнении вы получите следующий результат в окне PowerShell, после чего выберите [Y]. В зависимости от версии операционной системы Windows могут быть доступны другие варианты. В этом случае выберите вариант, который лучше всего соответствует желаемому результату.

PS C:\Program Files> set-ExecutionPolicy Unrestricted

Execution Policy Change

The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might expose

you to the security risks described in the about\_Execution\_Policies help topic at

http://go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the execution policy?

[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"):

Теперь войдите в папку Timebeat и выполните следующие команды:

.\install-service-timebeat.ps1  
Если вы неправильно настроили ExecutionPolicy, вы увидите в окне PowerShell вывод, похожий на приведенный ниже.

PS C:\Program Files\timebeat-1.2.1-windows-x86\_64> .\install-service-timebeat.ps1

.\install-service-timebeat.ps1 : File C:\Program Files\timebeat-1.2.1-windows-x86\_64\install-service-timebeat.ps1

cannot be loaded. The file C:\Program Files\timebeat-1.2.1-windows-x86\_64\install-service-timebeat.ps1 is not

digitally signed. You cannot run this script on the current system. For more information about running scripts and

setting execution policy, see about\_Execution\_Policies at http://go.microsoft.com/fwlink/?LinkID=135170.

At line:1 char:1

+ .\install-service-timebeat.ps1

+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

+ CategoryInfo : SecurityError: (:) [], PSSecurityException

+ FullyQualifiedErrorId : UnauthorizedAccess

Если ExecutionPolicy настроен правильно, вы увидите следующий вывод PowerShell:

PS C:\Program Files\timebeat-1.2.1-windows-x86\_64> .\install-service-timebeat.ps1

Security warning

Run only scripts that you trust. While scripts from the internet can be useful, this script can potentially harm your

computer. If you trust this script, use the Unblock-File cmdlet to allow the script to run without this warning

message. Do you want to run C:\Program Files\timebeat-1.2.1-windows-x86\_64\install-service-timebeat.ps1?

[D] Do not run [R] Run once [S] Suspend [?] Help (default is "D"):

На этом этапе выберите [R], чтобы продолжить процесс. После выбора [R] появится следующий результат:

Status Name DisplayName

------ ---- -----------

Stopped timebeat timebeat

Чтобы завершить установку, вам нужно будет ещё раз запустить установочный скрипт PowerShell, который будет выглядеть следующим образом (обратите внимание, что вам нужно будет снова нажать [R]:

PS C:\Program Files\timebeat-1.2.1-windows-x86\_64> .\install-service-timebeat.ps1

Security warning

Run only scripts that you trust. While scripts from the internet can be useful, this script can potentially harm your

computer. If you trust this script, use the Unblock-File cmdlet to allow the script to run without this warning

message. Do you want to run C:\Program Files\timebeat-1.2.1-windows-x86\_64\install-service-timebeat.ps1?

[D] Do not run [R] Run once [S] Suspend [?] Help (default is "D"): R

\_\_GENUS : 2

\_\_CLASS : \_\_PARAMETERS

\_\_SUPERCLASS :

\_\_DYNASTY : \_\_PARAMETERS

\_\_RELPATH :

\_\_PROPERTY\_COUNT : 1

\_\_DERIVATION : {}

\_\_SERVER :

\_\_NAMESPACE :

\_\_PATH :

ReturnValue : 5

PSComputerName :

\_\_GENUS : 2

\_\_CLASS : \_\_PARAMETERS

\_\_SUPERCLASS :

\_\_DYNASTY : \_\_PARAMETERS

\_\_RELPATH :

\_\_PROPERTY\_COUNT : 1

\_\_DERIVATION : {}

\_\_SERVER :

\_\_NAMESPACE :

\_\_PATH :

ReturnValue : 0

PSComputerName :

Status : Stopped

Name : timebeat

DisplayName : timebeat

Это всё, что нужно для установки агента Timebeat. Теперь нам нужно настроить файл конфигурации, чтобы ваше приложение Timebeat могло получать данные в формате UTC и отправлять их в Elastic. Для этого нам нужно отредактировать следующий файл:

Копировать

C:\\Program Files\timebeat<version-number/full-file-name>\timebeat.yml

Далее нам нужно убедиться, что ваш лицензионный файл находится в правильном месте. Мы рекомендуем размещать его в следующем месте:

Копировать

C:\\Program Files\timebeat<version-number/full-file-name>\timebeat.lic

Хотя это расположение необязательно, оно рекомендуется для удобства. Вам нужно будет обновить расположение по умолчанию в файле конфигурации timebeat.yml, поэтому, если вы размещаете лицензию в другом месте, не забудьте обновить путь в файле конфигурации. В отличие от установок в Linux, путь должен быть указан в одинарных кавычках (это относится ко всем расположениям путей). Пример выглядит следующим образом:

Копировать

'C:\Program Files\timebeat-1.2.1-windows-x86\_64\timebeat.lic'

Если ваши зависимости настроены и файл конфигурации обновлен, вы готовы запустить службу Timebeat. Мы рекомендуем запускать службу на переднем плане после внесения любых изменений в конфигурацию. Выполнение этой команды позволит выявить любые ошибки, которые могут присутствовать в файле конфигурации, чтобы вы могли быстро внести изменения и перезапустить службу.

Следующая команда запустит Timebeat на переднем плане в окне PowerShell, если вы находитесь в каталоге с файлами Timebeat. Если вы не находитесь в каталоге с файлами Timebeat, просто добавьте полный путь к timebeat.exe в начале:

Копировать

.\timebeat.exe -d “\*” -e

Как только вы убедитесь, что служба работает должным образом, вы можете запустить её обычным способом с помощью следующих команд:

Start-Service timebeat

Аналогичным образом вы можете перейти на вкладку «Службы» в диспетчере задач и запустить службу оттуда.

Чтобы убедиться, что Timebeat запущен, вы можете проверить его состояние с помощью:

* Откройте диспетчер задач, найдите Timebeat на вкладке «Службы» и убедитесь, что он работает

Подводные камни / задачи, которые нужно запомнить

Иногда вы выполняете описанные выше действия, но Timebeat всё равно не работает.

Ниже мы приводим некоторые примеры распространённых ошибок, которые часто упускают из виду или о которых забывают.

**Firewalld:**У вас есть возможность настроить Firewalld, но в большинстве случаев мы рекомендуем просто отключить его с помощью приведенных ниже команд.

Копировать

sudo systemctl stop firewalld

sudo systemctl disable firewalld

**Брандмауэр Windows / Защитник Windows:** аналогично, в Windows вам нужно отключить (или настроить) брандмауэр Windows и Защитник Windows.  
Это делается на панели управления, просто убедитесь, что нужные флажки сняты.

**Лицензионный ключ:**это решение подходит для многих элементов в конфигурации Timebeat. Всегда проверяйте, что имена путей указаны точно и правильно, вы не всегда можете полагаться на значения по умолчанию, даже если в большинстве случаев это нормально. Убедитесь, что ваш лицензионный ключ имеет правильное имя файла и соответствует тому, что вы настроили в файле timebeat.yml.

**По умолчанию используются пути с косой чертой:**иногда это требует проб и ошибок, поэтому мы всегда рекомендуем запускать Timebeat в терминале при первом запуске после внесения изменений в конфигурацию, особенно при перемещении файлов конфигурации между операционными системами. По умолчанию Timebeat в Linux использует пути с косой чертой, а в Windows — с обратной косой чертой. Аналогично, в Linux используются двойные кавычки, а в Windows — одинарные. Убедитесь, что они указаны правильно. Снова используя приведённую выше команду отладки (с флагом «\*»), вы можете определить проблемную строку в файле конфигурации.

**YAML:**При редактировании файла YAML всегда помните, что нужно использовать пробелы, а не табуляцию. Если вы будете использовать табуляцию, файл будет прочитан неправильно, и, как говорят мудрецы, табуляцию практически невозможно найти, так как она выглядит так же, как пробелы.

Как настроить домен PTP

Последнее обновление 2 года назад

Конфигурация PTP по умолчанию:

Копировать

# PTP Config example

#- protocol: ptp

#domain: 0

#serve\_unicast: true

#max\_unicast\_subscribers: 0

#serve\_multicast: true

#server\_only: true

#announce\_interval: 1

#sync\_interval: 0

#delayrequest\_interval: 0

#unicast\_master\_table: ['1.2.3.4','2.3.4.5', '3.4.5.6']

#delay\_strategy: e2e # other options which can be used are: 'p2p'

#hybrid\_e2e: false # Send delay requests as unicast (enterprise profile)

#priority1: 128

#priority2: 128

#monitor\_only: false

#use\_layer2: false # Use ptp over ethernet instead of IP/UDP

#interface: enp2s0

#profile: 'G.8275.2' # other options which can be used are: 'G.8275.1', 'G.8265.1' and 'enterprise-draft'

#logsource: 'Grandmaster Clock in NY4' # configurable name for source included in all logging records for source

#asymmetry\_compensation: 0 # In nanoseconds. Static compensation for know asymmetry (+/- as the case maybe)

#max\_packets\_per\_second: 0 # If inbound packet rate exceeds this a WRED algorithm is used

Чтобы настроить домен PTP, измените приведённую ниже строку на нужный вам домен.

Копировать

domain: 0

Итак, конфигурация будет выглядеть следующим образом:

Копировать

# PTP Config example

- protocol: ptp

domain: 0

Как обслуживать одноадресный PTP

Последнее обновление 2 года назад

Конфигурация PTP по умолчанию:

Копировать

# PTP Config example

#- protocol: ptp

#domain: 0

#serve\_unicast: true

#max\_unicast\_subscribers: 0

#serve\_multicast: true

#server\_only: true

#announce\_interval: 1

#sync\_interval: 0

#delayrequest\_interval: 0

#unicast\_master\_table: ['1.2.3.4','2.3.4.5', '3.4.5.6']

#delay\_strategy: e2e # other options which can be used are: 'p2p'

#hybrid\_e2e: false # Send delay requests as unicast (enterprise profile)

#priority1: 128

#priority2: 128

#monitor\_only: false

#use\_layer2: false # Use ptp over ethernet instead of IP/UDP

#interface: enp2s0

#profile: 'G.8275.2' # other options which can be used are: 'G.8275.1', 'G.8265.1' and 'enterprise-draft'

#logsource: 'Grandmaster Clock in NY4' # configurable name for source included in all logging records for source

#asymmetry\_compensation: 0 # In nanoseconds. Static compensation for know asymmetry (+/- as the case maybe)

#max\_packets\_per\_second: 0 # If inbound packet rate exceeds this a WRED algorithm is used

Чтобы настроить одноадресные линии передачи PTP:.

Копировать

serve\_unicast: true

server\_only: true

unicast\_master\_table: ['insert IP address of distribution interface']

Итак, конфигурация будет выглядеть следующим образом:

Копировать

# PTP Config example

- protocol: ptp

domain: 0

serve\_unicast: true

server\_only: true

unicast\_master\_table: ['insert IP address of distribution interface']

Как обслуживать многоадресную рассылку PTP

Последнее обновление 2 года назад

Конфигурация PTP по умолчанию:

Копировать

# PTP Config example

#- protocol: ptp

#domain: 0

#serve\_unicast: true

#max\_unicast\_subscribers: 0

#serve\_multicast: true

#server\_only: true

#announce\_interval: 1

#sync\_interval: 0

#delayrequest\_interval: 0

#unicast\_master\_table: ['1.2.3.4','2.3.4.5', '3.4.5.6']

#delay\_strategy: e2e # other options which can be used are: 'p2p'

#hybrid\_e2e: false # Send delay requests as unicast (enterprise profile)

#priority1: 128

#priority2: 128

#monitor\_only: false

#use\_layer2: false # Use ptp over ethernet instead of IP/UDP

#interface: enp2s0

#profile: 'G.8275.2' # other options which can be used are: 'G.8275.1', 'G.8265.1' and 'enterprise-draft'

#logsource: 'Grandmaster Clock in NY4' # configurable name for source included in all logging records for source

#asymmetry\_compensation: 0 # In nanoseconds. Static compensation for know asymmetry (+/- as the case maybe)

#max\_packets\_per\_second: 0 # If inbound packet rate exceeds this a WRED algorithm is used

Чтобы настроить домен многоадресной рассылки PTP feed, измените приведенные ниже строки:

Копировать

serve\_multicast: true

server\_only: true

interface: insert interface id

Итак, конфигурация будет выглядеть следующим образом:

Копировать

# PTP Config example

- protocol: ptp

domain: 0

serve\_multicast: true

server\_only: true

interface: ens1

**Конфигурация NMEA по умолчанию:**

Копировать

*# NMEA-GNSS input config example (can provide majortime for PPS sources)*

*# Eight data bits, no parity bit, and one stop bit (8N1) is assumed*

*# (It is not recommended to configure a NMEA source if you have an alternative secondary source)*

*#- protocol: nmea*

*# device: '/dev/ttyS0' # Serial device path*

*# baud: 9600 # Serial device baud rate*

*# offset: 0 # Static offset of RMC line*

*# monitor\_only: false*

*# Other vendor specific GNSS receivers input config example (can provide majortime for PPS sources)*

*# Eight data bits, no parity bit, and one stop bit (8N1) is assumed*

*# (It is not recommended to configure a Mini-JLT source if you have an alternative secondary source)*

*#- protocol: timecard-mini # "mini-jlt" (Jackson Labs) is also an option*

*# device: '/dev/ttyUSB0' # Serial device path*

*# baud: 115200 # Serial device baud rate*

*# offset: 0 # Static offset of RMC line*

*# monitor\_only: false*

Чтобы настроить исходный код NMEA, измените приведенные ниже строки:

Копировать

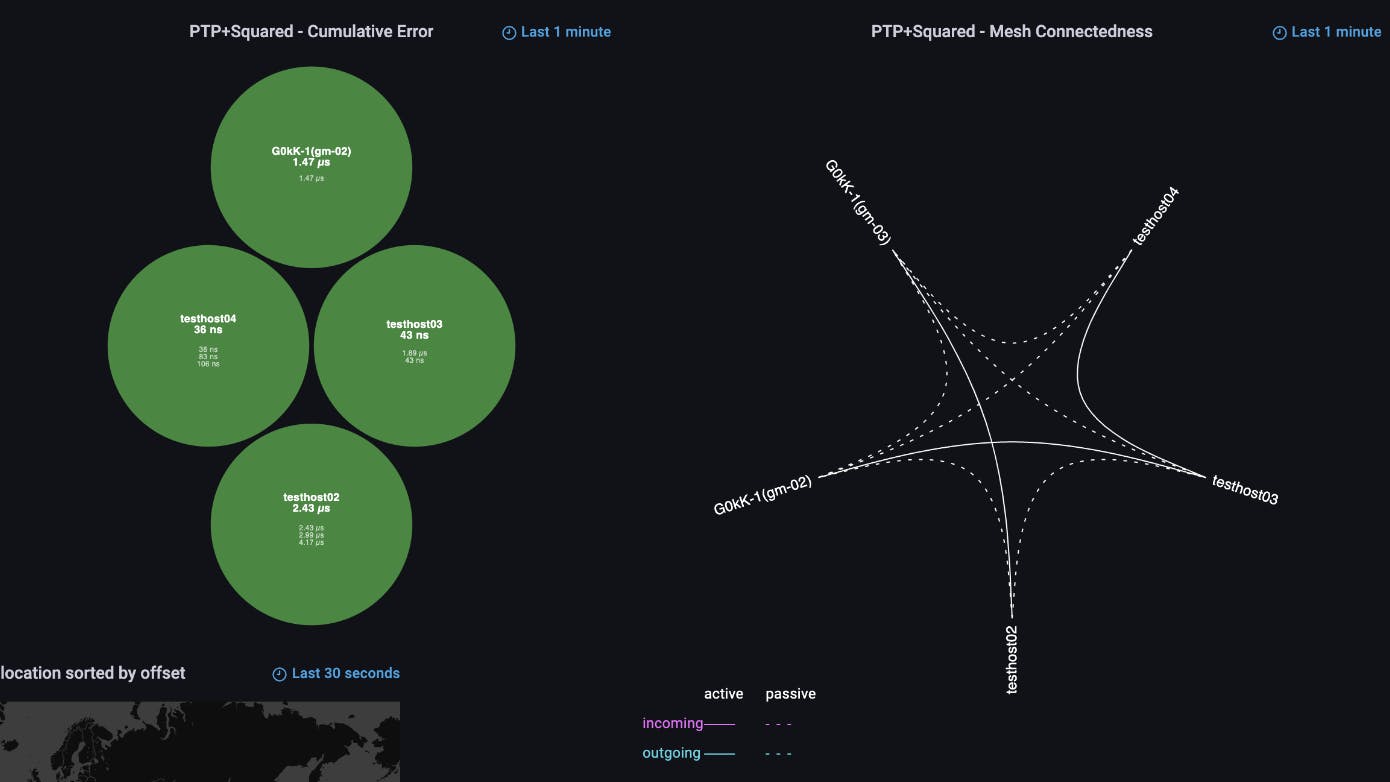
- protocol: timebeat\_opentimecard\_mini *# "mini-jlt" (Jackson Labs) is also an option*

device: '/dev/ttyS0' *# Serial device path*

baud: 9600 *# Serial device baud rate*

Краткое начало: PTP + Squared - основы построения сетки

PTP+Squared — это динамическая, самовосстанавливающаяся функция распределения и синхронизации PTP в Timebeat. На первый взгляд эта система может показаться сложной и потенциально трудной в настройке и использовании, но если вы будете следовать этому руководству, то быстро поймёте, что это не так.



Прежде чем мы начнём, необходимо разобраться в двух концепциях, которые сделают настройку простой и практически не требующей усилий при работе в больших масштабах.  
  
Концепция 1:

Корневые узлы:

Корневой узел может выполнять множество различных функций, но в конечном счёте это будет устройство, которое вводит источник PTP в сеть. Если вы используете одно из наших устройств Open Time, мы рекомендуем сделать его корневым узлом. Если нет, то это будет первое устройство, подключённое к вашей текущей GMC.

Концепция 2:

Узлы сетки:

Узлы сетки - это узлы в развертывании, которые не подключены к прямому источнику UTC. Это означает, что они будут находиться на расстоянии как минимум одного перехода от UTC. В зависимости от размера сети и конкретной конфигурации узел сети, скорее всего, подключается напрямую к корневому узлу или находится на расстоянии одного прыжка от корневого узла. Однако это не проблема, поскольку PTP + Squared всегда использует наилучший из возможных источников, поэтому вы всегда получите наилучшую синхронизацию.

Лучше всего начать настройку PTP+Squared с корневого узла.

В отличие от узлов Grid, корневому узлу требуется основной источник времени. Он настраивается так же, как и при использовании базовой конфигурации для основного источника. Мы рассмотрим это подробнее ниже.

Копировать

*# \_\_\_\_\_\_ \_ \_\_ \_\_*

*# /\_ \_\_/(\_)\_\_\_\_ \_\_\_ \_\_\_ / /\_ \_\_\_ \_\_\_\_ \_ / /\_*

*# / / / // \_\_ `\_\_ \ / \_ \ / \_\_ \ / \_ \ / \_\_ `// \_\_/*

*# / / / // / / / / // \_\_// /\_/ // \_\_// /\_/ // /\_*

*# /\_/ /\_//\_/ /\_/ /\_/ \\_\_\_//\_.\_\_\_/ \\_\_\_/ \\_\_,\_/ \\_\_/*

*################### Timebeat Configuration AREA #########################*

timebeat:

*# Location of license key file for premium features*

license.keyfile: '/etc/timebeat/timebeat.lic'

*# Configuration for clock synchronisation*

clock\_sync:

*# Clocksync*

*# Default is true enabling clock synchronisation, setting to false for monitoring purposes only (false will not adjust the clock)*

adjust\_clock: true

*# To remove step limit restrictions comment out the below, this will allow the clock to be stepped indefinitely*

step\_limit: 15m *# "s", "h" "d" can be used to denominate seconds, hours or days for step limit.*

primary\_clocks:

*# PTP Config example*

- protocol: ptp

domain: 0

*# serve\_unicast: true # (Enterprise Feature)*

*# max\_unicast\_subscribers: 0 # (Enterprise Feature)*

*# serve\_multicast: true # (Enterprise Feature)*

*# server\_only: true # (Enterprise Feature)*

announce\_interval: 1

sync\_interval: 0

delayrequest\_interval: 0

*# unicast\_master\_table: ['1.2.3.4','2.3.4.5', '3.4.5.6']*

*# delay\_strategy: e2e # other options which can be used are: 'p2p'*

*# hybrid\_e2e: false # Send delay requests as unicast (enterprise profile)*

*# priority1: 128*

*# priority2: 128*

*# monitor\_only: false*

*# use\_layer2: false # Use ptp over ethernet instead of IP/UDP*

interface: enp2s0

*# profile: 'G.8275.2' # other options which can be used are: 'G.8275.1', 'G.8265.1' and 'enterprise-draft'*

*# logsource: 'Grandmaster Clock in NY4' # configurable name for source included in all logging records for source*

*# asymmetry\_compensation: 0 # In nanoseconds. Static compensation for know asymmetry (+/- as the case maybe)*

*# max\_packets\_per\_second: 0 # If inbound packet rate exceeds this a WRED algorithm is used*

Вы можете использовать приведённый выше фрагмент конфигурации в качестве основного источника. Дальнейшую настройку можно выполнить в соответствии с вашей средой или требованиями, но приведённая выше конфигурация является базовой для начала работы.

Просто убедитесь, что все элементы соответствуют конфигурации вашего источника PTP. Если вы не уверены, ознакомьтесь с другими нашими руководствами по настройке основного источника PTP

Шаг 2:

Квадратная конфигурация для корневого узла:

Поначалу это может показаться сложным, но это просто и легко воспроизвести, если следовать приведённым ниже инструкциям.

Копировать

ptpsquared:

*# Enable PTPSquared functionality*

enable: true

discovery:

*# mdns: true*

dht: true

*# dht\_seed\_list: ["/ip4/10.101.101.23/tcp/65107/p2p/16Uiu2HAmJiQvJQbja8pf5dKAZsSYxWmcDCxZaoYbMUL5X7GnXej9"]*

keypath: "/etc/timebeat/ptp2key.private"

*# domains: [115,116]*

interface: ens5f0

seats\_to\_offer: 4 *# Number of seats available to consume time from this node*

seats\_to\_fill: 3 *# Number of seats this node seeks to fill from other nodes to consume time*

concurrent\_sources: 1 *# Number of concurrent PTP sources used to steer our clock(s)*

*# active\_sync\_interval: -3 # sync interval for active sources*

*# active\_delayrequest\_interval: -3 # delay request interval for active sources*

*# monitor\_sync\_interval: 0 # sync interval for sources we monitor*

*# monitor\_delayrequest\_interval: 0 # delay request interval for sources we monitor*

*# capabilities: ["hqosc-1500"] # high-quality oscillator 1.5us drift per 24 hours*

*# preference\_score: 0 # Used when requesting capacity from reservations*

*# reservations: ["1500:50%:115,116", "750:25%"] # List containing client seat reservations:*

*# :*

*# :*

*#*

*# debug: false # debug logging specifically for ptpsquared*

*# advanced:*

*# asymmetry\_compensation: 0 # In nanoseconds. Static compensation for know asymmetry (+/- as the case maybe). All interfaces*

*# is\_better\_factor: 1.4 # How much "better" does a source need to be to be selected instead of a current source*

*# eos\_weight: 1.0 # Weight attributed to error of source std dev calculation in determining hop cost*

*# All costs below default to zero which infers that cumulative error of source is the only factor*

*# determining the cost of a particular time source path*

*# base\_hop\_cost: 0.0 # The default cost of a single hop to downstream peers*

*# swts\_cost: 0.0 # The cost of using software timestamping*

*# hwts\_cost: 0.0 # The cost of using hardware timestamping*

*# Enable latency analysis functionality*

*# latency\_analysis\_enable: false*

Для начала работы можно использовать базовую конфигурацию, описанную выше. Опять же, убедитесь, что у вас настроен правильный интерфейс. Это будет интерфейс, который отправляет PTP на узлы сети.

Все остальные элементы в разделе конфигурации PTP+squared можно изменить по своему усмотрению, но приведенная выше конфигурация по умолчанию поможет вам начать работу с корневым узлом.

(Обратите внимание, что если ваш корневой узел также является вашим главным узлом, например, Open Time Server, вы можете изменить количество одновременных источников на ноль, чтобы сетка не влияла на синхронизацию, если вы потеряете GNSS или прямой источник UTC).

Настройка узла сетки:

Этот метод немного сложнее при включении конфигурации узла Grid, но его гораздо проще воспроизвести в больших масштабах.

Чтобы настроить узел Grid, сначала выполните действия, описанные в разделе конфигурации корневого узла PTP+Squared выше. (Обратите внимание, что для узла Grid не требуется настраивать основной источник времени.)

Копировать

ptpsquared:

*# Enable PTPSquared functionality*

enable: true

discovery:

*# mdns: true*

dht: true

dht\_seed\_list: ["/ip4/10.101.101.23/tcp/65107/p2p/16Uiu2HAmJiQvJQbja8pf5dKAZsSYxWmcDCxZaoYbMUL5X7GnXej9"]

keypath: "/etc/timebeat/ptp2key.private"

*# domains: [115,116]*

interface: ens5f0

seats\_to\_offer: 4 *# Number of seats available to consume time from this node*

seats\_to\_fill: 3 *# Number of seats this node seeks to fill from other nodes to consume time*

concurrent\_sources: 1 *# Number of concurrent PTP sources used to steer our clock(s)*

*# active\_sync\_interval: -3 # sync interval for active sources*

*# active\_delayrequest\_interval: -3 # delay request interval for active sources*

*# monitor\_sync\_interval: 0 # sync interval for sources we monitor*

*# monitor\_delayrequest\_interval: 0 # delay request interval for sources we monitor*

*# capabilities: ["hqosc-1500"] # high-quality oscillator 1.5us drift per 24 hours*

*# preference\_score: 0 # Used when requesting capacity from reservations*

*# reservations: ["1500:50%:115,116", "750:25%"] # List containing client seat reservations:*

*# :*

*# :*

*#*

*# debug: false # debug logging specifically for ptpsquared*

*# advanced:*

*# asymmetry\_compensation: 0 # In nanoseconds. Static compensation for know asymmetry (+/- as the case maybe). All interfaces*

*# is\_better\_factor: 1.4 # How much "better" does a source need to be to be selected instead of a current source*

*# eos\_weight: 1.0 # Weight attributed to error of source std dev calculation in determining hop cost*

*# All costs below default to zero which infers that cumulative error of source is the only factor*

*# determining the cost of a particular time source path*

*# base\_hop\_cost: 0.0 # The default cost of a single hop to downstream peers*

*# swts\_cost: 0.0 # The cost of using software timestamping*

*# hwts\_cost: 0.0 # The cost of using hardware timestamping*

*# Enable latency analysis functionality*

*# latency\_analysis\_enable: false*

Вы заметите, что для узла Grid есть дополнительная строка конфигурации, которую нужно включить в разделе обнаружения:

Копировать

discovery:

*# mdns: true*

dht: true

dht\_seed\_list: ["/ip4/10.101.101.23/tcp/65107/p2p/16Uiu2HAmJiQvJQbja8pf5dKAZsSYxWmcDCxZaoYbMUL5X7GnXej9"]

Список dht\_seed\_list.

Для этого вам нужно будет ввести IP-адрес корневого узла. Он будет находиться на том же интерфейсе, что и выходной интерфейс PTP, указанный в разделе конфигурации PTP+Squared корневого узла.  
  
Вам также потребуется ввести идентификатор корневого узла. Чтобы найти его, вам нужно будет получить доступ к командной строке Timebeat. После настройки вы сможете найти его, выполнив команду «показать квадратные места».

После завершения настройки сетка будет доступна после запуска службы Timebeat на обоих устройствах.

Если у ваших дополнительных узлов сетки одинаковый интерфейс, вы можете скопировать и вставить эту конфигурацию во все из них, и они также присоединятся к сетке.

Конечно, вам следует выполнить и другие действия для настройки тайминга, такие как настройка источника/выхода данных, но в этом руководстве мы не будем рассматривать их.

Шаги командной строки для настройки PTP + Squared:

(Приведённые ниже команды предполагают, что вы работаете от имени пользователя root; если вы не являетесь пользователем root, просто добавьте sudo в начало каждой команды.)

Корневой узел:

Шаг 1. Определите интерфейс, на котором вы получаете PTP, и интерфейс, на котором вы хотите распространить PTP по сети (они могут быть одинаковыми, если ваша сеть настроена таким образом).

Копировать

*# ip a*

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000

2: eno1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq\_codel state DOWN group default qlen 1000

3: enp1s0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc fq\_codel state UP group default qlen 1000

...

Шаг 2:

отредактируйте файл конфигурации timebeat.yml, указав основной источник PTP:

Копировать

*# vi /etc/timebeat/timebeat.yml*

Копировать

ptpsquared:

*# Enable PTPSquared functionality*

enable: true

discovery:

*# mdns: true*

dht: true

*# dht\_seed\_list: ["/ip4/10.101.101.23/tcp/65107/p2p/16Uiu2HAmJiQvJQbja8pf5dKAZsSYxWmcDCxZaoYbMUL5X7GnXej9"]*

keypath: "/etc/timebeat/ptp2key.private"

*# domains: [115,116]*

interface: ens5f0

seats\_to\_offer: 4 *# Number of seats available to consume time from this node*

seats\_to\_fill: 3 *# Number of seats this node seeks to fill from other nodes to consume time*

concurrent\_sources: 1 *# Number of concurrent PTP sources used to steer our clock(s)*

*# active\_sync\_interval: -3 # sync interval for active sources*

*# active\_delayrequest\_interval: -3 # delay request interval for active sources*

*# monitor\_sync\_interval: 0 # sync interval for sources we monitor*

*# monitor\_delayrequest\_interval: 0 # delay request interval for sources we monitor*

*# capabilities: ["hqosc-1500"] # high-quality oscillator 1.5us drift per 24 hours*

*# preference\_score: 0 # Used when requesting capacity from reservations*

*# reservations: ["1500:50%:115,116", "750:25%"] # List containing client seat reservations:*

*# :*

*# :*

*#*

*# debug: false # debug logging specifically for ptpsquared*

*# advanced:*

*# asymmetry\_compensation: 0 # In nanoseconds. Static compensation for know asymmetry (+/- as the case maybe). All interfaces*

*# is\_better\_factor: 1.4 # How much "better" does a source need to be to be selected instead of a current source*

*# eos\_weight: 1.0 # Weight attributed to error of source std dev calculation in determining hop cost*

*# All costs below default to zero which infers that cumulative error of source is the only factor*

*# determining the cost of a particular time source path*

*# base\_hop\_cost: 0.0 # The default cost of a single hop to downstream peers*

*# swts\_cost: 0.0 # The cost of using software timestamping*

*# hwts\_cost: 0.0 # The cost of using hardware timestamping*

*# Enable latency analysis functionality*

*# latency\_analysis\_enable: false*

Шаг 3:

настройте CLI так, чтобы он был включен:

Копировать

cli:

*# Enable the SSH based CLI interface. I.e. to access: ssh -p 65129 admin@127.0.0.1*

enable: true

bind\_port: 65129

bind\_host: 127.0.0.1

*# Server SSH key. If path specified key will be loaded or generated. Otherwise key will be random on startup*

server\_key: "/etc/timebeat/cli\_id\_rsa"

*# Authorised SSH keys (yes... with an s not a z....)*

*# authorised\_keys: "/etc/timebeat/authorised\_keys"*

*# CLI username and password*

username: "admin"

password: "password"

Приведённую выше конфигурацию можно скопировать, но просто раскомментируйте значения по умолчанию, и всё будет готово.

Шаг 4:

Настройте конфигурацию PTP + Squared корневого узла.

Копировать

ptpsquared:

*# Enable PTPSquared functionality*

enable: true

discovery:

*# mdns: true*

dht: true

dht\_seed\_list: ["/ip4/10.101.101.23/tcp/65107/p2p/16Uiu2HAmJiQvJQbja8pf5dKAZsSYxWmcDCxZaoYbMUL5X7GnXej9"]

keypath: "/etc/timebeat/ptp2key.private"

*# domains: [115,116]*

interface: ens5f0

seats\_to\_offer: 4 *# Number of seats available to consume time from this node*

seats\_to\_fill: 3 *# Number of seats this node seeks to fill from other nodes to consume time*

concurrent\_sources: 1 *# Number of concurrent PTP sources used to steer our clock(s)*

*# active\_sync\_interval: -3 # sync interval for active sources*

*# active\_delayrequest\_interval: -3 # delay request interval for active sources*

*# monitor\_sync\_interval: 0 # sync interval for sources we monitor*

*# monitor\_delayrequest\_interval: 0 # delay request interval for sources we monitor*

*# capabilities: ["hqosc-1500"] # high-quality oscillator 1.5us drift per 24 hours*

*# preference\_score: 0 # Used when requesting capacity from reservations*

*# reservations: ["1500:50%:115,116", "750:25%"] # List containing client seat reservations:*

*# :*

*# :*

*#*

*# debug: false # debug logging specifically for ptpsquared*

*# advanced:*

*# asymmetry\_compensation: 0 # In nanoseconds. Static compensation for know asymmetry (+/- as the case maybe). All interfaces*

*# is\_better\_factor: 1.4 # How much "better" does a source need to be to be selected instead of a current source*

*# eos\_weight: 1.0 # Weight attributed to error of source std dev calculation in determining hop cost*

*# All costs below default to zero which infers that cumulative error of source is the only factor*

*# determining the cost of a particular time source path*

*# base\_hop\_cost: 0.0 # The default cost of a single hop to downstream peers*

*# swts\_cost: 0.0 # The cost of using software timestamping*

*# hwts\_cost: 0.0 # The cost of using hardware timestamping*

*# Enable latency analysis functionality*

*# latency\_analysis\_enable: false*

Шаг 5:

Запустите службу timebeat

Копировать

*# service timebeat start*

Шаг 6:

Получите идентификатор корневого узла из командной строки:

Копировать

*# ssh -p 65129 admin@127.0.0.1*

admin@127.0.0.1's password:

Welcome to Timebeat 1.4.8-enterprise CLI

Timebeat# show squared seats

My ID is: 16Uiu2HAmJiQvJQbja8pf5dKAZsSYxWmcDCxZaoYbMUL5X7GnXej9

My RootDistance is: 0

Time sent from other peers to this peer:

Current server peer(s): 0

Unfilled client seats: 0

Max seats: 0

Time sent from this peer to other peers:

Current client peer(s): 0

Unfilled server seats: 0

Max seats: 0

Timebeat# exit

Connection to 127.0.0.1 closed.

(Приведённый выше идентификатор является лишь примером, и у каждого корневого узла он свой).

На этом корневой узел будет завершен.

Узел сетки:

Копировать

ptpsquared:

*# Enable PTPSquared functionality*

enable: true

discovery:

*# mdns: true*

dht: true

dht\_seed\_list: ["/ip4/10.101.101.23/tcp/65107/p2p/16Uiu2HAmJiQvJQbja8pf5dKAZsSYxWmcDCxZaoYbMUL5X7GnXej9"]

keypath: "/etc/timebeat/ptp2key.private"

*# domains: [115,116]*

interface: ens5f0

*# seats\_to\_offer: 4 # Number of seats available to consume time from this node*

*# seats\_to\_fill: 3 # Number of seats this node seeks to fill from other nodes to consume time*

*# concurrent\_sources: 1 # Number of concurrent PTP sources used to steer our clock(s)*

*# active\_sync\_interval: -3 # sync interval for active sources*

*# active\_delayrequest\_interval: -3 # delay request interval for active sources*

*# monitor\_sync\_interval: 0 # sync interval for sources we monitor*

*# monitor\_delayrequest\_interval: 0 # delay request interval for sources we monitor*

*# capabilities: ["hqosc-1500"] # high-quality oscillator 1.5us drift per 24 hours*

*# preference\_score: 0 # Used when requesting capacity from reservations*

*# reservations: ["1500:50%:115,116", "750:25%"] # List containing client seat reservations:*

*# :*

*# :*

*#*

*# debug: false # debug logging specifically for ptpsquared*

*# advanced:*

*# asymmetry\_compensation: 0 # In nanoseconds. Static compensation for know asymmetry (+/- as the case maybe). All interfaces*

*# is\_better\_factor: 1.4 # How much "better" does a source need to be to be selected instead of a current source*

*# eos\_weight: 1.0 # Weight attributed to error of source std dev calculation in determining hop cost*

*# All costs below default to zero which infers that cumulative error of source is the only factor*

*# determining the cost of a particular time source path*

*# base\_hop\_cost: 0.0 # The default cost of a single hop to downstream peers*

*# swts\_cost: 0.0 # The cost of using software timestamping*

*# hwts\_cost: 0.0 # The cost of using hardware timestamping*

*# Enable latency analysis functionality*

*# latency\_analysis\_enable: false*

Шаг 1:

Шаг 1. Определите интерфейс, на котором вы получаете PTP, и интерфейс, на котором вы хотите распространить PTP по сети (они будут одинаковыми).

Копировать

*# ip a*

1: lo: <LOOPBACK,UP,LOWER\_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000

2: eno1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq\_codel state DOWN group default qlen 1000

3: enp1s0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc fq\_codel state UP group default qlen 1000

...

Шаг 2:

отредактируйте файл конфигурации timebeat.yml, добавив источник PTP с квадратичной настройкой:

Копировать

*# vi /etc/timebeat/timebeat.yml*

Смотрите приведенный выше пример конфигурации.

Чтобы создать узел сетки в файле конфигурации, необходимо изменить раздел dht\_seed\_list в соответствии с вашим корневым узлом.

Для этого вам понадобится IP-адрес корневого узла (который мы нашли ранее) и идентификатор корневого узла (также найденный ранее).

Мини-руководство по устранению неполадок Open Timecard

Несколько ошибок могут помешать корректной работе Open Timecard Mini. Вот список наиболее распространённых проблем.

**Неправильно установлены DIP-переключатели**

Возможно, что переключатели на мини-плате тайм-карты застряли в промежуточном положении и для замыкания или размыкания цепи их нужно слегка подтолкнуть в нужном направлении.

***Pi Не загружается / Нет выхода HDMI***

Предположим, что первый переключатель на материнской плате установлен в положение «ВКЛ». В этом случае Raspberry Pi CM4 не будет пытаться выполнить обычную загрузку с eMMC (встроенного хранилища / microSD). Вместо этого он будет отображаться как USB-устройство, доступное через порт micro-USB с надписью «загрузка eMMC» (обычно он используется для перепрошивки eMMC или SD-карты).

***Серийная консоль недоступна.***

Неправильно настроенные переключатели могут препятствовать доступу к консоли. Ознакомьтесь с этой статьей, чтобы узнать, как правильно настроить DIP-переключатели для вашего случая использования, и убедитесь, что они настроены правильно. (Подробнее о доступе к последовательной консоли см. ниже.)

***CM4 отсутствует***

Если вы используете Timecard Mini без Raspberry Pi CM4 (что вполне допустимо), то DIP-переключатели 2-5 должны быть включены.

***Незакрепленные кабели***

Кабели UF, соединяющие перегородку SMA на внешней стороне металлического кронштейна с портами модуля GNSS, могут отсоединиться, поэтому убедитесь, что кабели надежно закреплены.

***Проблемы, связанные с лицензией***

Остановите timebeat следующим образом:

Копировать

systemctl stop timebeat

а затем запустите его с помощью

Копировать

/usr/share/timebeat/bin/timebeat -c /etc/timebeat/timebeat.yml -e

следите за выводами, чтобы не было сообщений, связанных с лицензией:

***Лицензия не найдена***

Невозможно прочитать файл с лицензионным ключом: открыть /etc/timebeat/timebeat.lic: такого файла или каталога нет

***Лицензия недействительна***

Срок действия лицензии истек / не работает в корпоративном режиме

Timebeat не будет работать должным образом без соответствующей лицензии. Если вы видите этот вывод после запуска Timebeat, значит, лицензия не работает в корпоративном режиме и, скорее всего, не будет выполнять то, что вам нужно:

Копировать

2029-02-02T00:00:06.051Z        INFO    [License Manager]       logging/external.go:94  License verified. Timebeat feature set: free

2029-02-02T00:00:06.051Z        INFO    [License Manager]       logging/external.go:94  License issued to yourname@youremail.com (Your Name)

2029-02-02T00:00:06.051Z        INFO    [License Manager]       logging/external.go:94  Enterprise features expired on 1985-10-26 01:21:00 +0000 PST

Если у вас возникла эта проблема, получите бесплатную лицензию на Timebeat [здесь](https://timebeat.app/downloads).

Спутниковые группировки не видны / проблемы с антенной

**Антенна не подключена**

При выполнении следующих команд на хост-компьютере или при входе в систему Raspberry Pi CM4:

Копировать

systemctl stop timebeat

apt-get install minicom

minicom -D /dev/serial0

hit the keys: <CTRL-A> Z P C <ENTER> (to set the baud rate to 9600)

Результат должен выглядеть примерно так:

Копировать

$GPGSV,2,2,08,09,42,200,48,17,39,234,47,19,44,263,27,28,15,035,26,161 $GPGSV,2,1,05,11,01,307,,12,05,325,,21,05,138,,25,02,002,,06F$GPGSV,2,2,05,25,02,002,,057 $GAGSV,2,1,05,05,38,203,45,09,71,105,23,11,19,137,13,34,52,302,24,779$GAGSV,2,2,05,36,69,155,30,74E $GAGSV,1,1,04,04,27,051,,15,03,311,,21,03,013,,27,05,325,,077$GBGSV,4,1,13,06,24,041,17,09,33,055,15,11,25,171,26,14,81,234,40,176 $GBGSV,4,2,13,16,21,037,26,21,18,062,27,26,07,100,16,27,11,322,27,17B$GBGSV,4,3,13,28,49,289,20,33,52,241,38,39,14,031,17,42,70,065,32,176 $GBGSV,4,4,13,43,43,193,48,143

Если результат выглядит похоже на этот:

Копировать

$GNRMC,144311.00,V,,,,,,,210823,,,N,V\*11

$GNVTG,,,,,,,,,N\*2E

$GNGGA,144311.00,,,,,0,00,99.99,,,,,,\*7A

$GNGSA,A,1,,,,,,,,,,,,,99.99,99.99,99.99,1\*33

$GNGSA,A,1,,,,,,,,,,,,,99.99,99.99,99.99,3\*31

$GNGSA,A,1,,,,,,,,,,,,,99.99,99.99,99.99,4\*36

$GNGSA,A,1,,,,,,,,,,,,,99.99,99.99,99.99,5\*37

Если это так, то, скорее всего, антенна отключена или не видно спутников. Антенны не работают стабильно в помещении, хотя могут работать с перебоями.

Если Timebeat работает во время выполнения этой задачи, вы увидите искажённые результаты, поэтому сначала остановите Timebeat.

***Ошибки в кабелях***

Если вы хотите, чтобы PPS работал при использовании CM4, необходимо подключить кабель RJ-45. Кроме того, если кабель не подключен, когда сервер настроен на обслуживание PTP, вы увидите неожиданное поведение и/или сообщения об ошибках.

***Проблемы с конфигурацией***

При запуске тестирования конфигурация CM4 в файлах /boot/cmdline.txt и /boot/config.txt не должна быть изменена. Если в эти файлы внесены изменения, потребуется перезагрузка. Кроме того, настоятельно рекомендуется использовать образ SD-карты CM4, доступный по адресу [https:.//timebeat.app](https://timebeat.app/downloads).

Кроме того, файл конфигурации /etc/timebeat/timebeat.yml требует правильной настройки для вашего случая использования. Ознакомьтесь с нашими руководствами, чтобы узнать, как настроить его так, как вам нужно.

***Режим помех активен***

Если запущена служба, которая устанавливает системные часы или PHC независимо от Timebeat, то Timebeat по умолчанию может работать в «режиме помех». Возможными виновниками этого могут быть **timesyncd**, **ptp4l**, **chronyи ntpd**.

Чтобы узнать, какие службы запущены в текущий момент, используйте:

Копировать

systemctl -type=service -state=running

Чтобы остановить их, используйте:

Копировать

systemctl stop <servicename>

Чтобы остановить и отключить их, используйте:

Копировать

systemctl disable <servicename>--now

***Нет доступа к последовательному порту***

После подтверждения правильности настройки DIP-переключателей

Для доступа к последовательному порту GNSS требуется драйвер MaxLinear, входящий в состав ядра, однако он может не входить в вашу текущую сборку.

Копировать

dmesg | grep tty

Результат должен быть примерно таким:

Копировать

[ 5.730290] 0000:82:00.0: ttyS[3] at MMIO 0xc8000000 (irq = 69, base\_baud = 7812500) is a XR17V35X[ 5.744865] 0000:82:00.0: ttyS[4] at MMIO 0xc8000400 (irq = 69, base\_baud = 7812500) is a XR17V35X

Копировать

lspci | grep XR17V35X

Должно дать:

Копировать

82:00.0 Serial controller: Exar Corp. XR17V3521 Dual PCIe UART (rev 03)

Если драйвер не установлен, его можно найти в исходном коде ядра.

Копировать

cd /usr/src/kernels/kernel-version

Копировать

make menuconfig

Затем найдите SERIAL\_8250\_EXAR, чтобы включить необходимые модули, и убедитесь, что они включены.

Копировать

make modules ; make modules install

Копировать

insmod /usr/lib/modules/6.0.7/kernel/drivers/gpio/gpio-exar.ko /usr/lib/modules/6.0.7/kernel/drivers/gpio/gpio-exar.ko

Как отключить PHC в TAI

**Раздел конфигурации Linux по умолчанию:**

Копировать

linux\_specific:

*# Enable hardware timestamping on Linux SOF\_TIMESTAMPING\_(R|T)X\_HARDWARE*

*# (setting this to false will also disable steering based on PPS)*

*# hardware\_timestamping: true*

*# Enable external software timestamping on Linux SOF\_TIMESTAMPING\_(R|T)X\_SOFTWARE*

*# external\_software\_timestamping: true*

*# Synchronise non-master PHC (nic) clocks on Linux*

*# sync\_nic\_slaves: false*

*# Don't synchronise these clocks on Linux*

*# disable\_adjustment: ['system', 'ens1', 'enp2s0']*

*# Fixed PHC offset strategy (default order of preference is 1st: precise, 2nd: extended,*

*# 3rd: pps, 4th: basic). List of ":"*

*# phc\_offset\_strategy: ["ens1:pps"]*

*# PHC smoothing mechanism. Defaults to "minimum". Other option is "median". List of ":"*

*# phc\_smoothing\_strategy: ["ens1:median"]*

*# Enable the internal modified lucky packet PHC outlier filter*

*# phc\_lp\_filter\_enabled: false*

*# Enable the internal none-Gausian PHC outlier filter*

*# phc\_ng\_filter\_enabled: false*

*# Interface specific PHC samples to determine offset from system clock (default is 15: list of ":")*

*# phc\_samples: ["ens5f0:5"]*

*# Use 1-step on the list of interfaces below if available*

*# phc\_one\_step: ["ens5f0", "ens1"]*

*# Maintain PHC clocks in TAI*

*# phc\_as\_tai: true*

Чтобы отключить синхронизацию PHC с TAI, используйте следующую конфигурацию:

Копировать

*# Maintain PHC clocks in TAI*

phc\_as\_tai: false