**Summary:** This document provides implementation steps to implement and validate time accuracy with Software Timestamping on Windows 10 and Server 2019 scenarios.

RS5 Validation Guide

Table of Contents

[Overview 3](#_Toc517247835)

[Description 3](#_Toc517247836)

[Troubleshooting and Feedback 3](#_Toc517247837)

[Test Activities 3](#_Toc517247838)

[Activities 3](#_Toc517247839)

[Activity 1: Module and DSC Resource Installation 3](#_Toc517247840)

[Activity 2: Identify adapters for timestamping 4](#_Toc517247841)

[Activity 3: Enable Timestamping 4](#_Toc517247842)

[Activity 4: Disable Timestamping 4](#_Toc517247843)

[Activity 5: Verify Timestamping is Enabled 4](#_Toc517247844)

[Activity 6: Use DSC to configure your system 5](#_Toc517247845)

# Overview

Many sectors require accurate time. The financial sector requires accurate time so that transactions are properly timestamp to 50ms, 1ms or even 100µs accuracy. Windows Server 2019 and the corresponding Windows 10 release continues to improve Windows’ accuracy by leveraging new feature, Software Timestamping improving our ability to maintain a precise and accurate system clock benefitting both physical and virtual workloads.

# Description

When NTP or PTP receive a timing packet through the network from their time server it must be processed by the OS’ networking stack prior to being consumed in the operating system. Each component in the networking stack introduces a variable amount of latency that negatively affects the accuracy of the timing measurement. With Software Timestamping, you can calculate the latency introduced by networking stack and make the appropriate adjustments.

Operating System Requirements

Software timestamping is currently supported only on Windows Server 2019 and Windows 10 (v1809). This feature is available on current Insider Builds.

# Troubleshooting and Feedback

Please submit an issue at: <https://github.com/Microsoft/SDN/issues> and add the [Time](https://github.com/microsoft/sdn/issues?q=is%3Aissue+is%3Aopen+label%3Atime) tag

# Test Activities

Below are the activities included to validate this feature:

1. Module and DSC Resource Installation
2. Identify adapters for timestamping
3. Enable Timestamping
4. Disable Timestamping
5. Verify Timestamping is configured
6. Use DSC to configure your system

# Activities

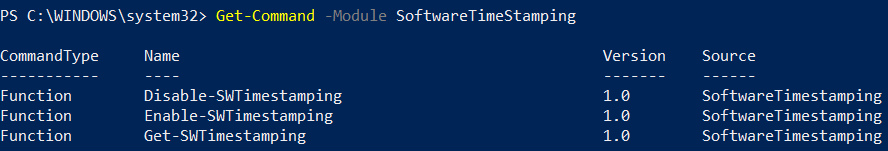
## Activity 1: Module and DSC Resource Installation

This activity will help you install the PowerShell Module and DSC Resource that can be configured for timestamping. Timestamping can be configured on physical and virtual machines.

1. Download the **SoftwareTimestamping** module from the Microsoft SDN GitHub repo: <https://github.com/Microsoft/SDN/tree/master/Time>
2. Extract and install the **SoftwareTimestamping** module into the following path on the system to be configured:

C:\Program Files\WindowsPowerShell\Modules

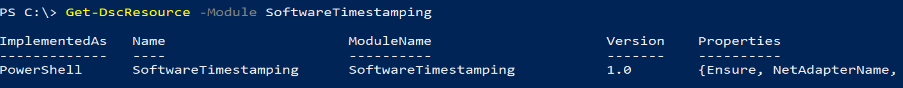
1. Open an elevated PowerShell prompt and run the following command to verify the module was installed properly:



If the module does not show the necessary cmdlets, please close and re-open PowerShell (alternatively, import the module with ‘Import-Module SoftwareTimeStamping’

If the module still does not show, it is possible that you improperly extracted the module or placed it into an incorrect folder.

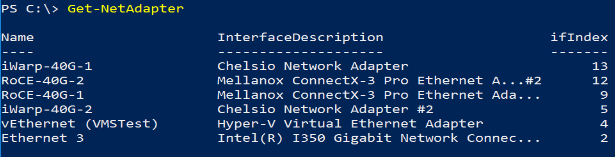
1. In the elevated PowerShell prompt, run the following command to verify the DSC Resource was installed properly:



## Activity 2: Identify adapters for timestamping

This activity will help you identify adapters that can be configured for timestamping. Timestamping can be configured on physical and virtual NICs.

On the host where the **SoftwareTimestamping** module was installed run the following command to identify network adapters in the system:



Note the **Name** property for the network adapter you wish to enable. This will be used in future activities.

## Activity 3: Enable Timestamping

To enable Software Timestamping on one or more NICs, run the following command. For NetAdapterName, refer to the name from Activity 2.



Next restart the NIC using the command:



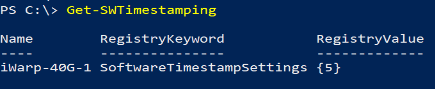
## Activity 4: Disable Timestamping

To disable Software Timestamping on one or more NICs, use the following command:



## Activity 5: Verify Timestamping is Enabled

To verify Software Timestamping on one or more NICs, refer to the name in Activity 2 and run the following command:



To see the actual measurements evaluated by the timestamping feature, enable the w32tm debug log with the following command:



Now look for entries in the log like this. Note the ETD (Estimated Time Delay) and ATD (Actual Time Delay) values. Time Delay values are shown in FileTime units which are 100-nanosecond units. Therefore, there were 315, 100-nanosecond file time units or 31.5 microseconds ((315 \* 100 nanoseconds) / 1000) of delay introduced by the networking stack for this timestamp.



## Activity 6: Use DSC to configure your system

Desired State Configuration can be used to deploy, manage, and maintain the software timestamping configuration across many servers. A simple example is provided in the module downloaded for this exercise however this can be expanded and included in your CI/CD pipeline as necessary. You could also use this to target groups of servers.

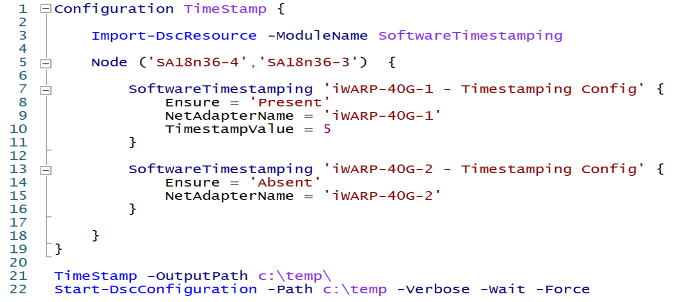
Open the PowerShell ISE. Next, open *Configuration.ps1* included in the **DSCExamples** folder included in this guide. If you placed the module in the recommended location included in this guide, then the example configuration will be located at:

*C:\Program Files\WindowsPowerShell\Modules\SoftwareTimestamping\DSCExamples\Configuration.ps1*

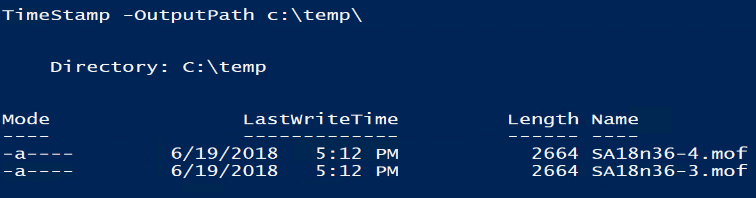
Use **Activity 2: Identify adapters for timestamping** to identify adapters that you would like to be configured and maintained.

Create a **SoftwareTimeStamping** instance for each adapter.

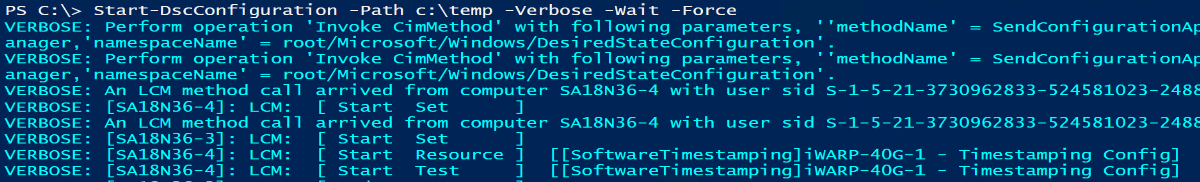
Modify the **Ensure**, **NetAdapterName**, and **TimestampValue** as necessary.



In this example, two nodes (SA18N36-3 and SA18N36-4) are selected and so two MOF files are created for deployment.



Now run Start-DSCConfiguration to push the configuration to the nodes.



Please note, more complex scenarios are outside of the scope of this document. Also, this DSC resource is implemented in a Class-based resource and requires at least WMF5.