# Application Note

# Zabbix Integration Guide

Advantech Czech s.r.o., Sokolska 71, 562 04 Usti nad Orlici, Czech Republic
Document No. APP-0089-EN, revised on March 23, 2026.

# Used symbols

**Important**

**Important** — Indicates a risk to personal safety or potential damage to the router. Follow these instructions precisely to prevent injury or equipment damage.

**Warning**

**Warning** — Highlights conditions that may cause malfunction, loss of data, or unexpected behavior in specific situations. Read carefully before proceeding.

**Info**

**Info** — Provides helpful tips, context, or references that improve understanding but are not strictly required to complete the task.

**Code Example**

```
Code Example – Copy-pasteable configuration snippets or CLI commands.
```

# Contents

# List of Figures

# List of Tables

# 1. Quick Setup Guide

This chapter provides a minimal set of steps required to start monitoring an Advantech router using Zabbix. For detailed configuration options, refer to the corresponding chapters.

> **Info**
>
> For simple monitoring, use SNMP. For advanced metrics and custom checks, use the *Zabbix Agent* Router App.

1. **Install Zabbix**
   Deploy and start a Zabbix server (for example, using the official virtual appliance). Ensure that the Zabbix web interface is accessible.
   Refer to Chapter *3.1 Server Installation*.

2. **Import templates**
   Import the Advantech YAML templates into the Zabbix web interface. These templates provide predefined items, triggers, and discovery rules.
   Navigate to *Data collection* → *Templates* → *Import*.
   Refer to Chapter *3.2 Advantech Templates*.

3. **Configure router**
   Enable and configure the monitoring interface on the router:

   - SNMP (for basic monitoring), or
   - Zabbix Agent Router App (for advanced monitoring).

   Ensure that network communication between the router and the Zabbix server is allowed.
   Refer to Chapter *4.1 Router Configuration (SNMP)* or Chapter *5.1 Router Configuration (Zabbix Agent)*.

4. **Add host**
   Create a new host in the Zabbix web interface, assign the appropriate template(s), and configure the correct interface type (SNMP or Agent).
   Refer to Chapter *4.2 Configuring a Host in Zabbix* or Chapter *5.2 Configuring a Host in Zabbix*.

5. **Verify data**
   Confirm that monitoring data is being collected:

   - Verify that the host availability indicator is green.
   - Review collected metrics in *Monitoring* → *Latest data*.

   If no data is displayed, verify connectivity, configuration, and update intervals.
   Refer to Chapter *4 Using SNMP Polling* or Chapter *5 Zabbix Agent Router App*. For common troubleshooting procedures, see Chapter *6 Troubleshooting*.

# 2. Zabbix Basics

## 2.1 Introduction

Zabbix is an enterprise-grade, open-source monitoring platform designed to track the performance and availability of IT infrastructure, including networks, servers, virtual machines, and cloud services.

It uses a client-server architecture, where a centralized Zabbix server collects, processes, and evaluates data from monitored devices. It can monitor a wide range of network parameters and the overall health and integrity of servers. For more information, visit *zabbix.com*.

The Zabbix server typically runs on a Linux operating system and supports a wide range of data collection protocols, including ICMP Ping, SNMP, SSH, and custom Zabbix agents.

Remote monitoring is the process of supervising IT systems from a central management server. In general, monitoring improves the reliability and security of your network because it facilitates the early detection of erroneous conditions.

> **Info**
>
> For an introduction to remote monitoring and a list of other monitoring tools, please refer to the *Remote Monitoring* Advantech Application Note.

## 2.2 Possible Interfaces

For gathering Zabbix data from an Advantech router, the following interfaces can be used:

- **SNMPv1/v2** – Retrieves predefined values via the SNMP protocol. This is the simplest way to implement data monitoring. For details, refer to Chapter *4 Using SNMP Polling*.
- **Agent** – Allows retrieval of additional and custom-defined items compared to the SNMP protocol. It requires the *Zabbix Agent* Router App to be installed on the router. For details, refer to Chapter *5 Zabbix Agent Router App*.

When to use SNMP vs Zabbix Agent?

- **SNMP** → simple, low overhead, read-only
- **Agent** → advanced metrics, custom checks, active mode

Agent-based monitoring can operate in two modes:

- **Passive mode** – The Zabbix server initiates the connection and polls the agent. This increases server load and may not work well for devices behind NAT or firewalls.
- **Active mode** – The agent initiates the connection to the Zabbix server. This approach reduces server load and works reliably in NAT or firewall-restricted environments.

## 2.3 Advantech Integration Tools

There are two free tools designed to simplify Zabbix integration on Advantech routers. While neither is mandatory for Zabbix data collection to work, they can be highly beneficial:

1. **Zabbix Templates File:** A YAML configuration file containing preconfigured items and triggers based on the *Conel MIB*, as well as templates for the Agent interface. For details, refer to Chapter *3.2 Advantech Templates*.
2. ***Zabbix Agent* Router App:** An application that can be installed directly on the router to enable advanced, agent-based monitoring in both passive and active checks. For details, refer to Chapter *5 Zabbix Agent Router App*.

# 3. Zabbix Server

## 3.1 Server Installation

The Zabbix server is the central component of the Zabbix monitoring software. It is responsible for polling and trapping data, calculating triggers, and sending notifications. Regardless of your chosen scenario, you will need a Zabbix server installed and running in your environment.

There are multiple options for acquiring and installing a Zabbix server[1]. For small environments or testing purposes, the easiest method is to deploy the pre-configured Zabbix virtual appliance.

For example, you can use VirtualBox[2] as a free virtualization tool. Download the pre-configured Zabbix virtual appliance[3] in the Open Virtualization Format (.ovf). An official manual[4] is available for the Zabbix appliance.

Once downloaded, open the installed VirtualBox application and import the downloaded appliance by navigating to *File* → *Import Appliance...*.

**Info**

- If the downloaded appliance is archived (for example, in a `.tar.gz` format), you need to extract it first. This will provide a directory containing the `.ovf` and `.vmdk` files.
- The default network setting for the virtual machine is usually NAT. Depending on your network environment, you may need to configure port forwarding or change the adapter type to *Bridged Adapter*.
- If needed, use the `ip addr` command in the guest console (Zabbix server) to find out the IP address assigned to the running server.

Start the virtual machine and allow the guest operating system to boot up. While it is usually not necessary, if you need to log in to the server console directly on the running guest, use the following default OS credentials:

- **Username:** `root`
- **Password:** `zabbix`

The primary goal is to access the Zabbix web interface (frontend). In your web browser, navigate to `http://<ip_address>`, where `<ip_address>` is the IP address of your Zabbix server. The default login credentials for the web interface are:

- **Username:** `Admin` (case-sensitive)
- **Password:** `zabbix`

[1] https://www.zabbix.com/download
[2] https://www.virtualbox.org
[3] https://www.zabbix.com/download_appliance
[4] https://www.zabbix.com/documentation/current/manual/appliance

## 3.2 Advantech Templates

These templates are highly useful in the Zabbix web interface, as they contain predefined definitions of SNMP entries of Advantech routers, including their OIDs and data types. Using these templates eliminates the need for manual data entry.

The templates are stored in the `zbx_icr_templates.yaml` configuration file. These templates are based on the *Conel MIB* (Management Information Base) and include preconfigured items for receiving via SNMP. This configuration file can be downloaded from the *Zabbix Agent* Router App page. The Conel MIB structure is described in detail in the *SNMP Object Identifiers* Application Note.

The YAML templates configuration file contains the following templates:

A) Retrievable via SNMP polling only (no *Zabbix Agent* Router App is required, refer to Chapter *4 Using SNMP Polling* ):

- **Advantech ICR Basic SNMP** – Defines basic system, hardware, and network metrics.
- **Advantech ICR Mobile 1 SNMP** – Defines metrics related to the mobile connection, such as signal strength, data usage, and SIM card details.

B) Retrievable via *Zabbix Agent Router App* (refer to Chapter *5 Zabbix Agent Router App*):

- **Advantech ICR Resources by Agent** – Defines system resource metrics (e.g., memory and storage usage) collected passively by the Zabbix server.
- **Advantech ICR Resources by Agent active** – Defines the same system resource metrics as the passive template, but utilizes active checks.

The following tables detail the contents of all the mentioned templates. The term *Populated Inventory* in the third column indicates that the value retrieved by the given Zabbix item is automatically written to the system inventory of the respective host in Zabbix.

| Zabbix Item | Description | Populated Inventory |
|---|---|---|
| Firmware | Information about the device firmware version. | OS |
| GNSS satellites | Number of GNSS satellites currently visible to the router. | |
| ICMP loss | Percentage of lost ICMP echo request packets. | |
| ICMP ping | Availability of the device via ICMP ping (0 = unreachable, 1 = reachable). | |
| ICMP response time | Response time to ICMP ping requests (in seconds). | |
| Location altitude | Altitude of the device location above sea level (in meters). | |
| Location latitude | Geographic latitude of the device location (in degrees). | Location latitude |
| Location longitude | Geographic longitude of the device location (in degrees). | Location longitude |
| Part number | Product part number assigned by the manufacturer. | Model |
| Product name | Product designation or model name. | Type |
| RTC battery | Status of the RTC backup battery (unknown, ok, empty). | |
| Serial number | Unique serial number of the device. | Serial number A |
| SNMP agent availability | Availability status of SNMP communication (0 = unavailable, 1 = available, 2 = unknown). | |
| SNMP traps (fallback) | Collects all SNMP traps that are not matched by other trap items. | |
| System contact details | Contact information for the administrator of the device. | Contact |
| System description | Textual description including hardware type, OS, and software details. | |
| System location | Physical location of the device (e.g., rack, room, building). | Location |
| System name | Administratively assigned device name (typically FQDN). | Name |
| System object ID | Vendor-specific identifier of the managed device type (OID). | |
| Temperature | Internal device temperature (in °C). | |
| Uptime (hardware) | Time since the device was last started (hardware uptime). | |
| Uptime (network) | Time since the SNMP agent (network management) was last restarted. | |
| Voltage | Input power supply voltage (in volts). | |

Table 1: Advantech ICR Basic SNMP template items

| Zabbix Item | Description | Populated Inventory |
|---|---|---|
| Mobile card | Index of the active SIM card (0 = primary, 1 = secondary, 2 = tertiary). | |
| Mobile connections 1 | Number of connections established using the primary SIM card (daily counter). | |
| Mobile connections 2 | Number of connections established using the secondary SIM card (daily counter). | |
| Mobile inbound data 1 | Amount of data received via the primary SIM card (in bytes, daily total). | |
| Mobile inbound data 2 | Amount of data received via the secondary SIM card (in bytes, daily total). | |
| Mobile offline time | Total time the mobile interface was offline (daily value). | |
| Mobile online time 1 | Total time connected using the primary SIM card (daily value). | |
| Mobile online time 2 | Total time connected using the secondary SIM card (daily value). | |
| Mobile operator | Name of the mobile network operator currently in use. | |
| Mobile outbound data 1 | Amount of data transmitted via the primary SIM card (in bytes, daily total). | |
| Mobile outbound data 2 | Amount of data transmitted via the secondary SIM card (in bytes, daily total). | |
| Mobile registration | Current network registration status (e.g., idle, searching, home, roaming). | |
| Mobile signal average | Average signal strength over the monitored period. | |
| Mobile signal level | Signal strength indicator (CSQ value in range 0–31). | |
| Mobile signal max | Maximum recorded signal strength over the monitored period. | |
| Mobile signal min | Minimum recorded signal strength over the monitored period. | |
| Mobile signal quality | Signal quality of the selected cell (in dB). | |
| Mobile signal strength | Instantaneous signal strength (in dBm). | |
| Mobile technology | Currently used mobile network technology (e.g., LTE, UMTS, 5G). | |
| Mobile uptime | Time since the current mobile connection was established. | |
| Modem IMEI | Unique IMEI identifier of the modem. | Serial number B |
| Modem SIM ICCID | ICCID (SIM card identifier) of the inserted SIM card. | |
| Modem SIM IMSI | IMSI (subscriber identity) of the SIM card. | |
| Strength threshold Fair | Signal strength threshold defining a fair-quality connection (technology-dependent). | |
| Strength threshold Weak | Signal strength threshold defining a weak connection (technology-dependent). | |

Table 2: Advantech ICR Mobile 1 SNMP template items

The strength thresholds *Fair* and *Weak* are auto-calculated items that depend on the used mobile technology. They are used by the signal strength triggers.

From the `Mobile-2` OID tree, the *Advantech ICR Mobile 1 SNMP* template exclusively uses aggregated values from the `MobileYesterday` table. The `MobileToday` table is not used because it contains incomplete, constantly changing data. Similarly, long-term aggregated tables like `MobileThisWeek` are unnecessary, as Zabbix stores the daily data and calculates its own long-term historical statistics.

| Zabbix Item | Description |
|---|---|
| Checksum `/etc/passwd` | MD5 checksum of the `/etc/passwd` file used to detect unauthorized changes. |
| Storage `/` free | Available disk space on the root filesystem (in bytes). |
| Storage `/opt` free | Available disk space on the `/opt` filesystem (in bytes). |
| Storage `/opt` used | Used disk space on the `/opt` filesystem (in bytes). |
| Storage `/opt` used % | Percentage of used disk space on the `/opt` filesystem. |
| Storage `/` used | Used disk space on the root filesystem (in bytes). |
| Storage `/` used % | Percentage of used disk space on the root filesystem. |
| Storage `/var/data` free | Available disk space on the `/var/data` filesystem (in bytes). |
| Storage `/var/data` used | Used disk space on the `/var/data` filesystem (in bytes). |
| Storage `/var/data` used % | Percentage of used disk space on the `/var/data` filesystem. |
| System memory available | Amount of available system memory (in bytes). |
| System memory used | Amount of used system memory (in bytes). |
| System memory used % | Percentage of used system memory. |

Table 3: Advantech ICR Resources by Agent template items

The *Advantech ICR Resources by Agent active* template contains the exact same items as the *Advantech ICR Resources by Agent* template. The difference is that its items are configured as active checks (*Zabbix agent (active)*).

A trigger is a logical expression that evaluates data collected by items and defines a specific condition or threshold. When the incoming data meets this condition, the trigger changes its state (e.g., from `OK` to `Problem`) to generate an alert. The templates listed above define the following triggers:

| **Template** | **Trigger Name** | **Condition** |
|---|---|---|
| *Advantech ICR Basic SNMP* | System name has changed<br>Host has been restarted<br>No SNMP data collection | System name differs from previous<br>Uptime < 10m<br>SNMP unavailable for 5m |
| *Advantech ICR Basic SNMP* (ICMP) | Unavailable by ICMP ping<br>High ICMP ping loss<br>High ICMP ping response time | ICMP ping fails (timeout)<br>{$ICMP_LOSS_WARN} < ICMP loss < 100%<br>ICMP response time > {$ICMP_RESPONSE_TIME_-WARN} |
| *Advantech ICR Basic SNMP* (Network Interfaces) | Interface {#IFNAME}: Link down<br>Interface {#IFNAME}: High bandwidth usage<br>Interface {#IFNAME}: High error rate<br>Interface {#IFNAME}: Ethernet has changed to lower speed | Operational status is down<br>Utilization > {$IF.UTIL.MAX}%<br>Errors > {$IF.ERRORS.WARN} for 5m<br>Reported speed is lower than previous |
| *Advantech ICR Mobile 1 SNMP* | Fair mobile signal<br>Weak mobile signal | Weak threshold < signal strength ≤ Fair threshold<br>signal strength ≤ Weak threshold |
| *Advantech ICR Resources by Agent* (Active & Passive) | /etc/passwd changed<br>{#STFILE} changed | MD5 checksum differs from previous<br>MD5 checksum differs from previous |

Table 4: Defined Zabbix triggers

## 3.3 Zabbix Web Configuration Structure

This section outlines the core components of a Zabbix configuration and how they interact to provide monitoring. Their relationships are illustrated in Figure 1.

At the top level, monitored devices are defined as **Hosts**, which can be organized into **Host groups** for easier management of large infrastructures.

Individual checks performed on a host are defined as **Items**. Each item represents a specific metric (numeric or textual), collected using a defined method (e.g., SNMP, SSH, or agent) and update interval. Each item is identified by a unique key, such as `system.cpu.load` .

Items are organized using **Tags** (formerly known as **Applications**), which simplify filtering and categorization (e.g., Info, Status, Interfaces). Some items also populate host **Inventory** fields automatically (e.g., Name, OS, Serial Number).

Collected data is evaluated by **Triggers**, which define conditions for generating alerts. For visualization, Zabbix uses **Graphs** to display item values over time.

To simplify configuration, related entities (items, triggers, graphs, or discovery rules) can be grouped into a **Template**. Templates can be linked to hosts or nested within other templates to ensure consistent configuration across devices.

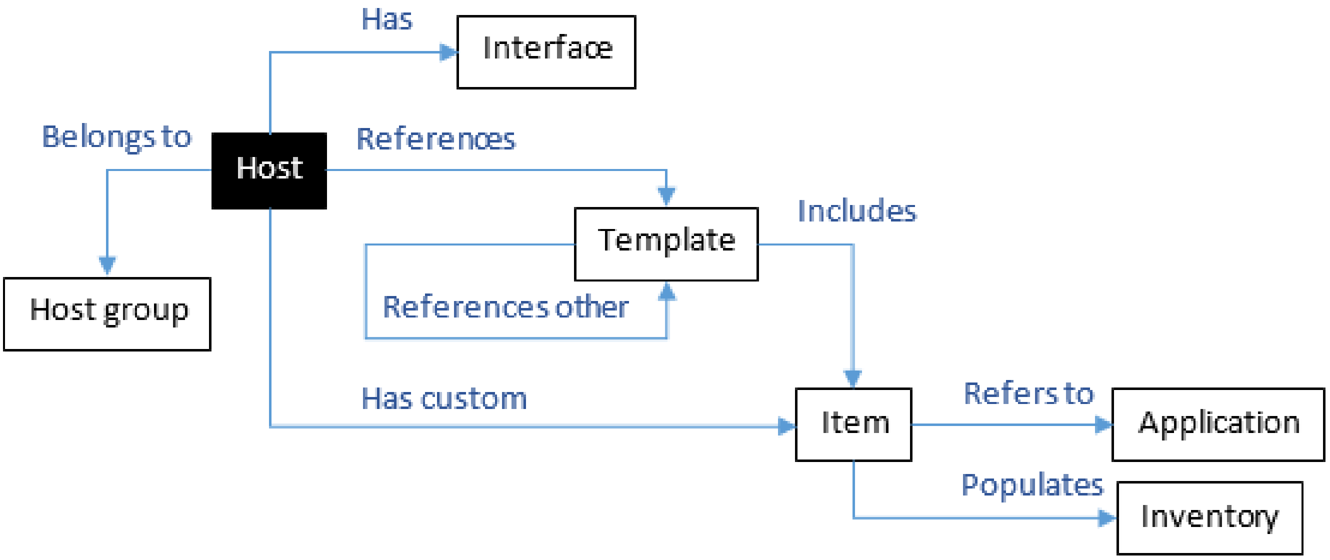


Figure 1: Logical schema of a Zabbix configuration

### Importing Templates

To import templates into the Zabbix web interface, navigate to *Data collection* → *Templates* and click *Import*. Select the YAML template file and proceed with the import.

**Info**

You can create custom templates tailored to your router and monitoring requirements. Existing items from Advantech templates can be reused to simplify configuration.

# 4. Using SNMP Polling

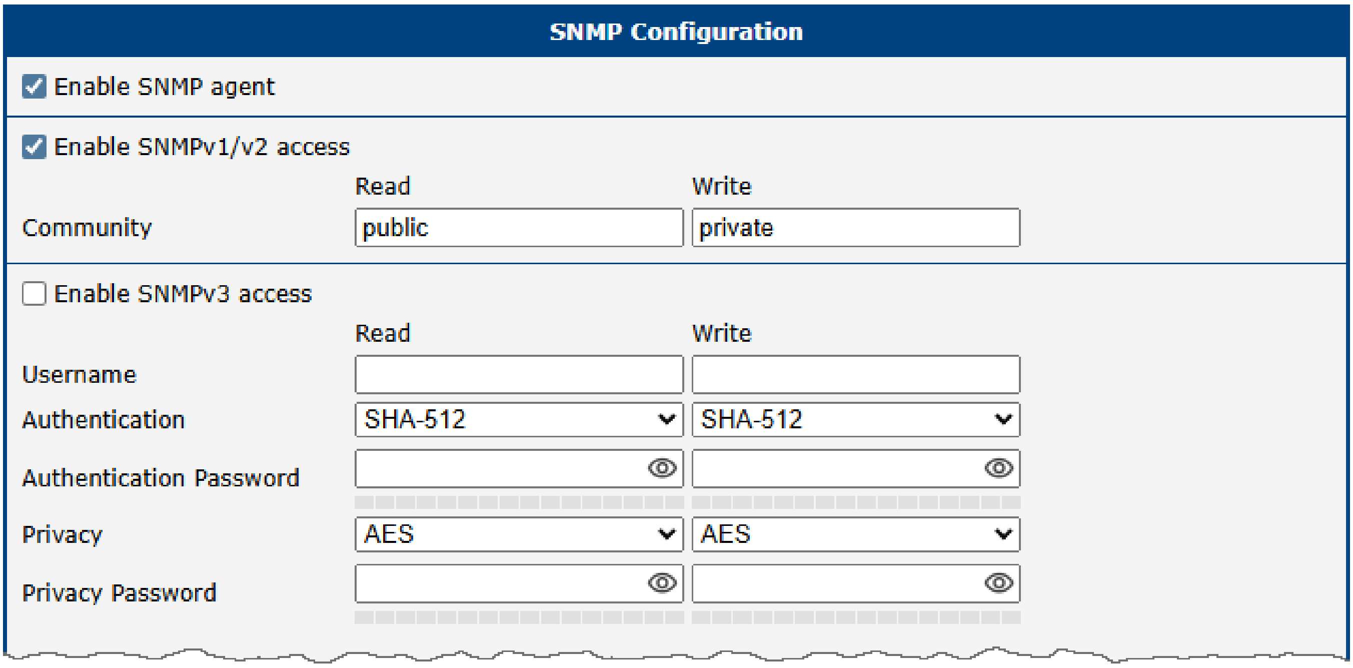
## 4.1 Router Configuration (SNMP)

For the Zabbix server to successfully receive data via the SNMP protocol, the SNMP service must be enabled and properly configured on the router. Additionally, related network settings, such as firewall rules, must be configured to ensure that SNMP traffic can freely pass between the Zabbix server and the router.

Figure 2 below illustrates a minimal workable configuration of the SNMP service on the router for SNMPv1/v2 access. The community string is set to `public` for Read access.

**Warning**

The default `public` community string should not be used in production environments.



**SNMP Configuration**

☑ Enable SNMP agent

☑ Enable SNMPv1/v2 access

| | Read | Write |
|---|---|---|
| Community | public | private |

□ Enable SNMPv3 access

| | Read | Write |
|---|---|---|
| Username | | |
| Authentication | SHA-512 | SHA-512 |
| Authentication Password | | |
| Privacy | AES | AES |
| Privacy Password | | |

Figure 2: SNMP router configuration

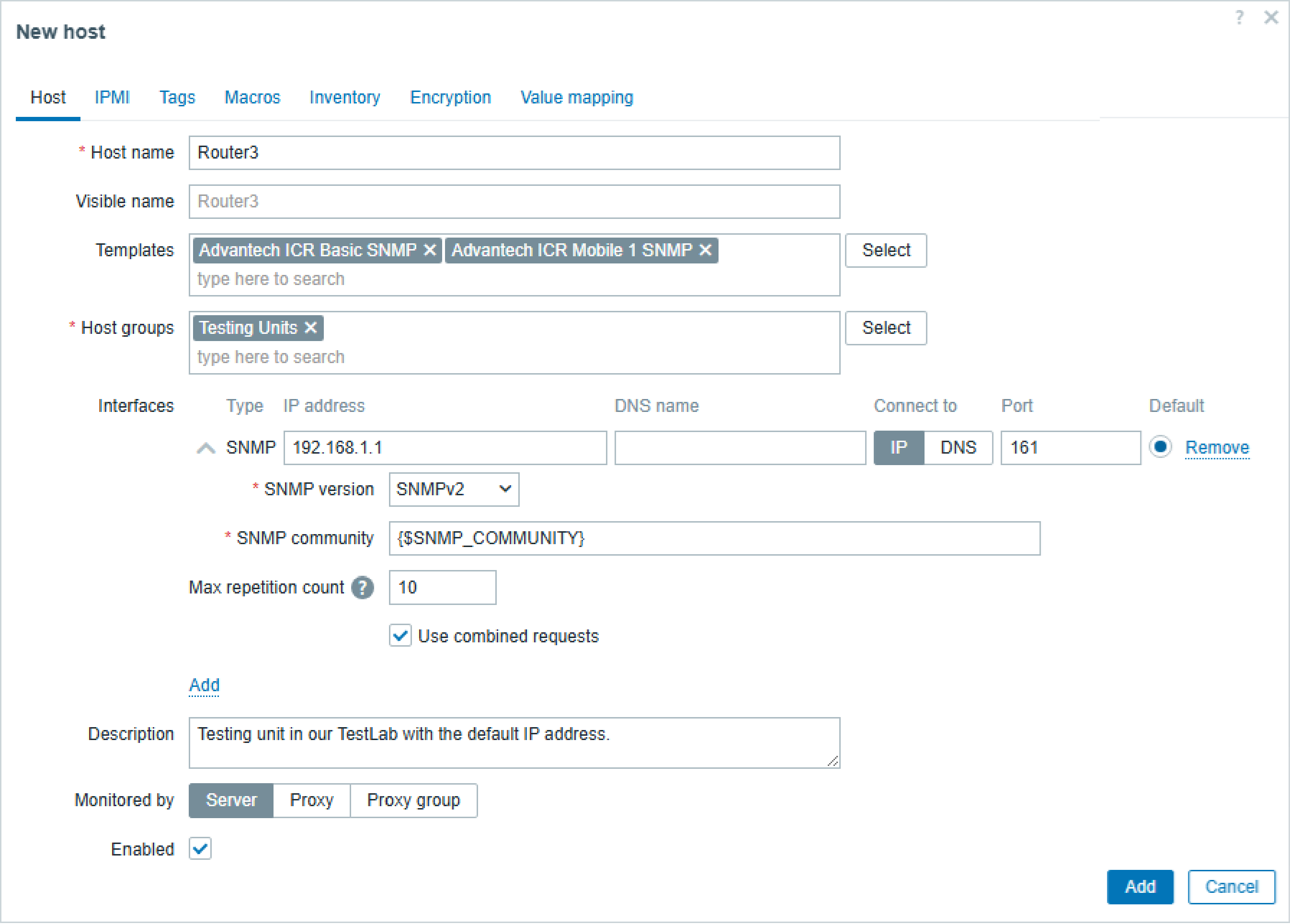
## 4.2 Configuring a Host in Zabbix

**Info**

This chapter describes a quick way to get Advantech router monitoring using SNMP interface. There are many options for configuring the Zabbix server and frontend; please refer to the official manuals on the *Zabbix Documentation* pages for more detailed information.

The core logical configuration structure was described in Chapter *3.3 Zabbix Web Configuration Structure*. Here is the simplest way to add a router to the monitored hosts in the Zabbix web interface:

- We assume that you have already imported the templates from the Advantech YAML file into the Zabbix web interface under *Data collection* → *Templates*, as described in Chapter *3.2 Advantech Templates*.
- In the Zabbix web interface, navigate to *Monitoring* → *Hosts* and click the *Create host* button. A *New host* configuration form will appear. Figure 3 illustrates the configuration of a new host for SNMP data polling.



New host

* Host name: Router3

Visible name: Router3

Templates: Advantech ICR Basic SNMP ✕ Advantech ICR Mobile 1 SNMP ✕ type here to search

* Host groups: Testing Units ✕ type here to search

Interfaces:

| Type | IP address | DNS name | Connect to | Port | Default |
|---|---|---|---|---|---|
| SNMP | 192.168.1.1 | | IP | 161 | ● Remove |

* SNMP version: SNMPv2

* SNMP community: {$SNMP_COMMUNITY}

Max repetition count: 10

☑ Use combined requests

Description: Testing unit in our TestLab with the default IP address.

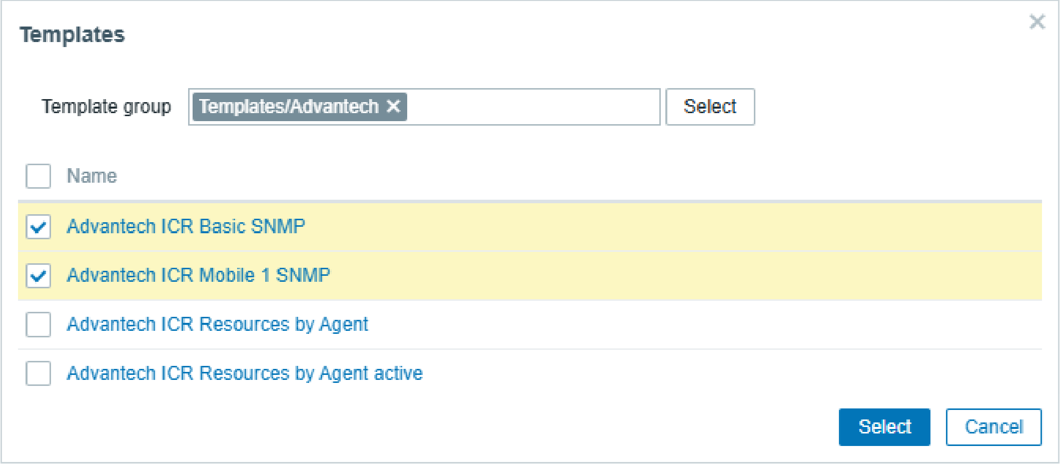
Monitored by: Server

Enabled: ☑

Figure 3: New host form with SNMP interface

- Fill in the following fields in the form:
  - *Host name* – An arbitrary but unique name for the host. This name does not need to match the router's actual hostname. You can optionally use the *Description* field below for a detailed host description.

  - *Templates* – Choose the imported **Advantech ICR Basic SNMP** and/or **Advantech ICR Mobile 1 SNMP** template for SNMP data polling (you can filter for Advantech templates as shown in Figure 9):



Templates

Template group: Templates/Advantech ✕

□ Name

☑ Advantech ICR Basic SNMP

☑ Advantech ICR Mobile 1 SNMP

□ Advantech ICR Resources by Agent

□ Advantech ICR Resources by Agent active

Figure 4: Choose template for SNMP

  - *Host groups* – Select from existing host groups. You can create your own group in *Data collection* → *Host groups*.
  - *Interfaces* – Click on *Add* and choose **SNMP**. The SNMP version should match the version you have configured on the router. The SNMP community string is set to the `{$SNMP_COMMUNITY}` macro, which is configured to `public` by default. This matches our router configuration shown in Figure 2.

- Finally, click the *Add* button at the bottom of the form to save the host.

If everything is configured correctly, after a few minutes you should see:

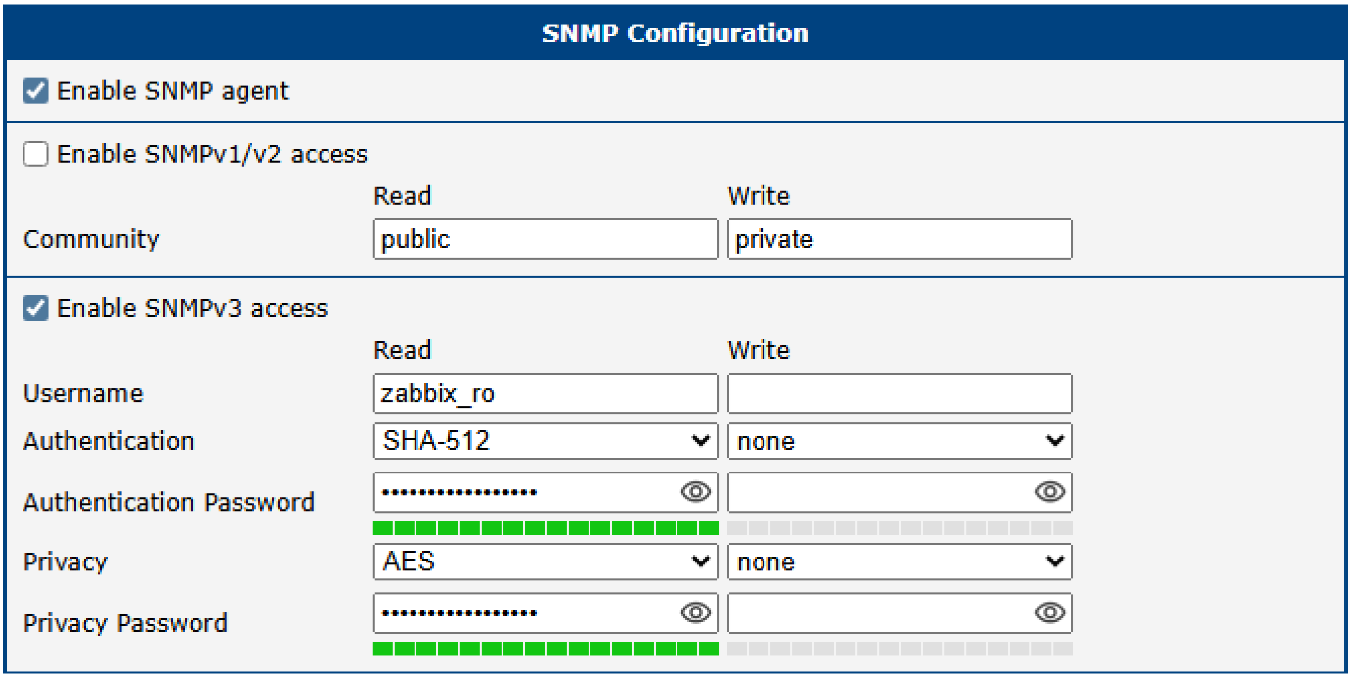
- In *Monitoring* → *Hosts*, the `SNMP` label in the *Availability* column for your host should turn green.
- Retrieved status information under *Monitoring* → *Latest data*. You can filter the results by your specific host name.

**Info**

- Every item has a defined refresh rate, so some items may be populated later than others. These intervals can be configured. If you want to request an immediate update for a specific item, navigate to *Data collection* → *Hosts* and click the *Items* link on your host's line. Select the desired item and click the *Execute now* button.
- If data from your hosts is not collected automatically, the SNMP poller on the Zabbix server may not be running or may be configured with an insufficient number of worker processes. This situation often occurs in the default configuration of the Zabbix appliance. Refer to the official Zabbix documentation for details about the `StartSNMPPollers` parameter in the *Zabbix server* configuration file.

## 4.3 SNMPv3 Security Configuration

For production environments, SNMPv3 with authentication and privacy should always be used instead of SNMPv1/v2, which do not provide encryption. To achieve this, configure SNMPv3 with authentication and privacy on the router, as shown in Figure 5.



**SNMP Configuration**

☑ Enable SNMP agent

☐ Enable SNMPv1/v2 access

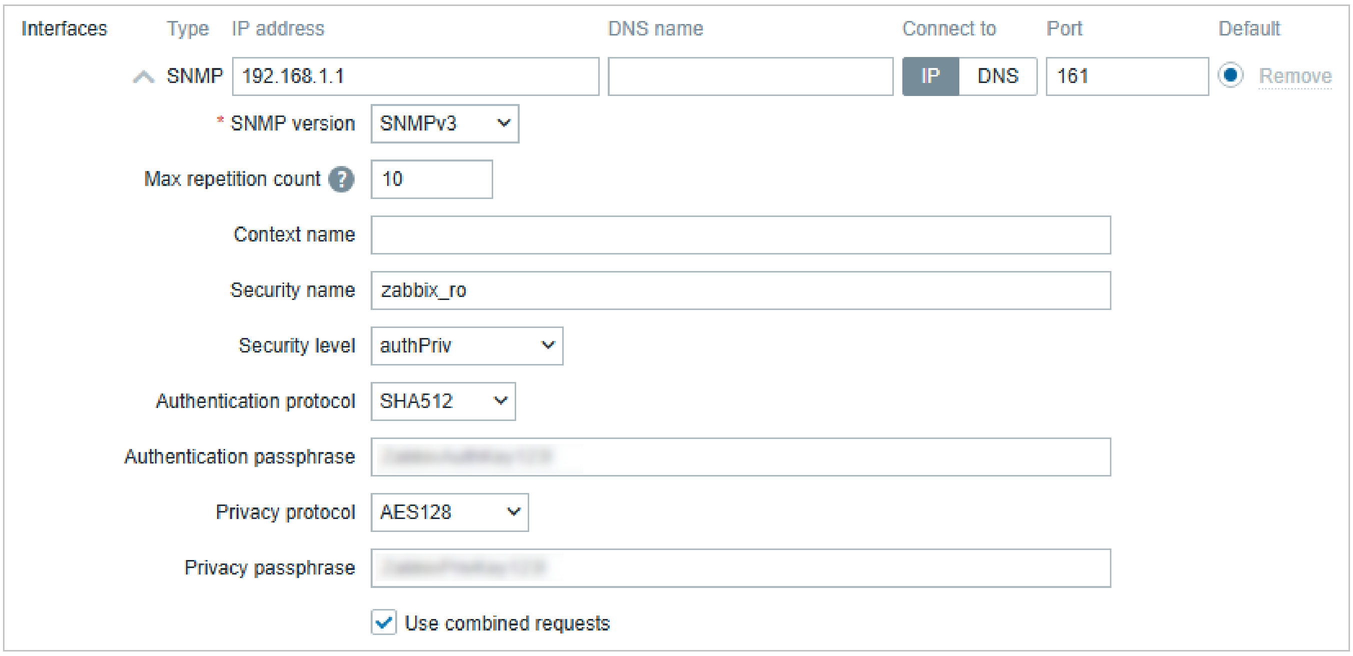
| | Read | Write |
|---|---|---|
| Community | public | private |

☑ Enable SNMPv3 access

| | Read | Write |
|---|---|---|
| Username | zabbix_ro | |
| Authentication | SHA-512 | none |
| Authentication Password | ••••••••••••••••• | |
| Privacy | AES | none |
| Privacy Password | ••••••••••••••••• | |

Figure 5: SNMP encryption on the router

Next, in the Zabbix web interface, configure the corresponding encryption settings for the host's SNMP interface so that they exactly match the router's configuration, as shown in Figure 6.



| Interfaces | Type | IP address | DNS name | Connect to | Port | Default | |
|---|---|---|---|---|---|---|---|
| | SNMP | 192.168.1.1 | | IP / DNS | 161 | ● | Remove |

| | |
|---|---|
| * SNMP version | SNMPv3 |
| Max repetition count | 10 |
| Context name | |
| Security name | zabbix_ro |
| Security level | authPriv |
| Authentication protocol | SHA512 |
| Authentication passphrase | [redacted] |
| Privacy protocol | AES128 |
| Privacy passphrase | [redacted] |

☑ Use combined requests

Figure 6: SNMP encryption on the Zabbix server

# 5. Zabbix Agent Router App

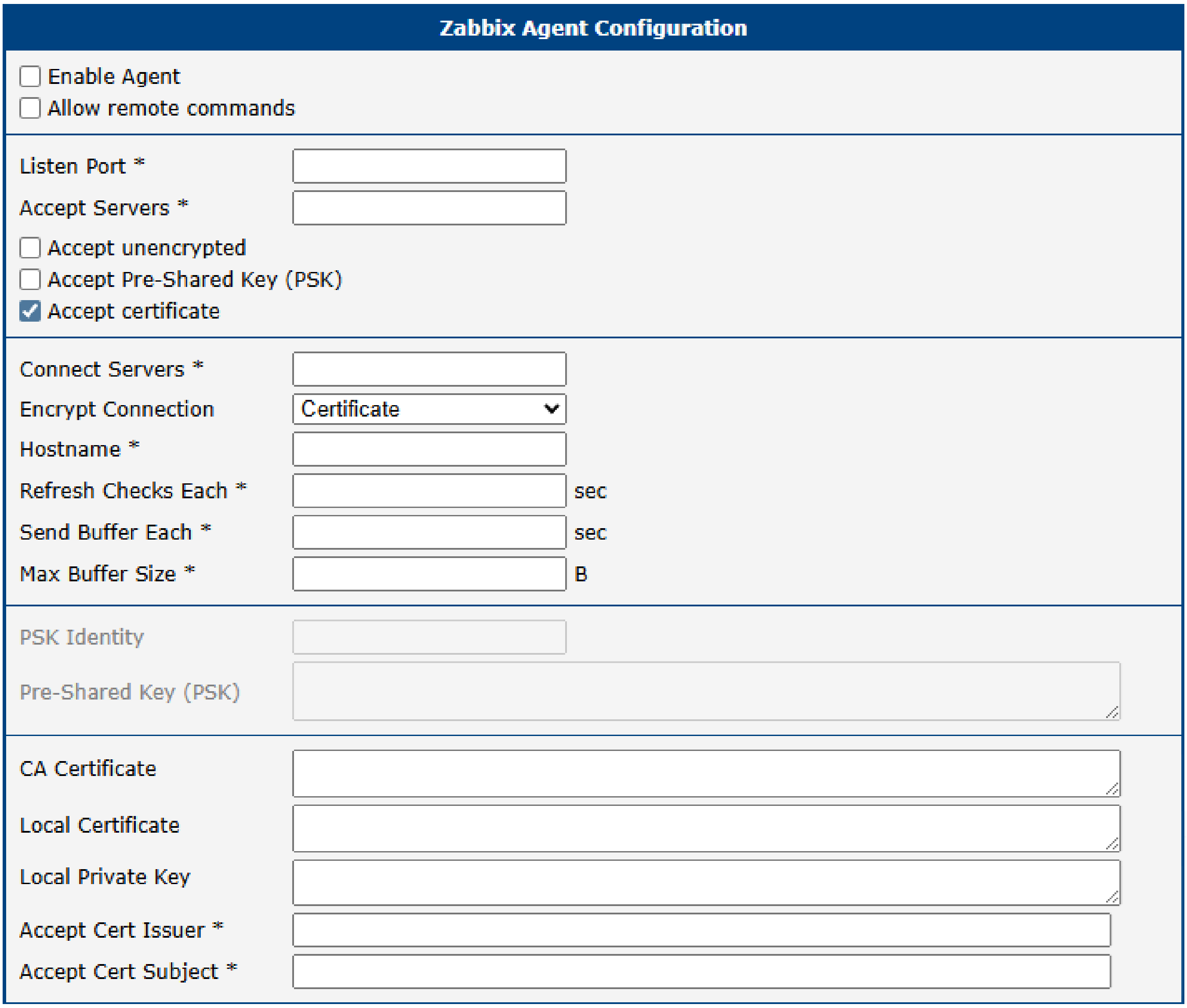
The *Zabbix Agent* Router App provides monitoring of values which are not available through the SNMP protocol, supporting both active and passive checks. Compared to SNMP, the Zabbix Agent provides active checks, custom keys, and TLS-based communication. This application can be downloaded from the *Zabbix Agent* Router App page.

## 5.1 Router Configuration (Zabbix Agent)

First, install the *Zabbix Agent* Router App on the router. For more information on how to upload a Router App, refer to the router's *Configuration Manual*, chapter *Customization* → *Router Apps*.

In the Router App configuration, you need to configure connectivity to the Zabbix server. There are two basic modes, passive and active, as described in Chapter *2.2 Possible Interfaces*.

Figure 7 shows the Router App configuration GUI, with all items explained in Table 5.



**Zabbix Agent Configuration**

☐ Enable Agent
☐ Allow remote commands

Listen Port *
Accept Servers *
☐ Accept unencrypted
☐ Accept Pre-Shared Key (PSK)
☑ Accept certificate

Connect Servers *
Encrypt Connection: Certificate
Hostname *
Refresh Checks Each * sec
Send Buffer Each * sec
Max Buffer Size * B

PSK Identity
Pre-Shared Key (PSK)

CA Certificate
Local Certificate
Local Private Key
Accept Cert Issuer *
Accept Cert Subject *

Figure 7: Zabbix agent configuration – The main part

| Item | Description |
|---|---|
| *Enable Agent* | Enables/disables the Zabbix agent service on the router. |
| *Allow Remote Commands* | Specifies whether remote commands from the Zabbix server are permitted. When disabled, `system.run` checks will be rejected. |
| *Listen Port* | The port on which the agent (in **passive mode**) listens for connections from the server. The default value is 10050. |
| *Accept Servers* | Incoming **passive connections** will only be accepted from the IP addresses listed here. Enter the IP address of your Zabbix server. Multiple comma-separated addresses can be provided. If left empty, passive mode is disabled. |
| *Accept unencrypted* | Allows **passive connections** without encryption (not recommended). Encryption of the host on Zabbix server should be set to *No encryption*, which is the default option. |
| *Accept Pre-Shared Key (PSK)* | Allows **passive connections** secured with TLS and a Pre-Shared Key (PSK). When enabled, the *PSK Identity* and *Pre-Shared Key (PSK)* fields must be configured. Encryption of the host on Zabbix server should be set to *PSK* and properly configured. |
| *Accept certificate* | Allows **passive connections** secured with TLS and a certificate. When enabled, the *CA Certificate*, *Local Certificate*, and *Local Private Key* must be configured. Encryption of the host on Zabbix server should be set to *Certificate* and properly configured. |
| *Connect Servers* | Specifies the connection to the Zabbix server (or servers) for **active checks**. Supported formats are `IP:port`, `IP`, `hostname:port`, or `hostname`. Multiple comma-separated addresses can be provided to use several independent Zabbix servers in parallel. If left empty, active checks are disabled. |
| *Encrypt Connection* | Specifies the encryption method the agent uses to connect to the Zabbix server. This must match the *Connections from host* setting in the Zabbix web interface. |
| *Hostname* | A unique hostname for the router. This **must exactly match** the *Host name* field configured in the Zabbix web interface of a host. If not specified, router's hostname will be used (defined in *Configuration → System → Identification*). |
| *Refresh Checks Each* | Defines how often (in seconds) the agent retrieves the list of active checks from the Zabbix server. The default value is 10 seconds. |
| *Send Buffer Each* | Defines the time interval (in seconds) for syncing buffered check results to the Zabbix server. The default value is 5 seconds. |
| *Max Buffer Size* | Defines the maximum size of the buffer. When this capacity is reached, the agent syncs the buffered values to the server immediately. The default value is 100. |
| *PSK Identity* | The pre-shared key identity string. This must match the *PSK identity* field in the Zabbix web interface. The same PSK identity is used for both passive and active checks. |
| *Pre-Shared Key (PSK)* | The pre-shared key string. This must match the *PSK* field in the Zabbix web interface. |
| *CA Certificate* | The CA certificate chain of the authority that issued the Zabbix server certificates. |
| *Local Certificate* | The certificate of the router, corresponding to its private key. The certificate purpose must include "client authentication". If generated via OpenSSL, `extendedKeyUsage = clientAuth` must be set. The CA certificate of the issuing authority must be included in the `TLSCAFile` on the Zabbix server. |
| *Local Private Key* | The private key of the router. The same private key and certificates are used for both passive and active checks. |
| *Accept Cert Issuer* | The allowed server certificate issuer. If specified, it must match the issuer of the server's certificate. |
| *Accept Cert Subject* | The allowed server certificate subject. If specified, it must match the subject of the server's certificate. |

Table 5: Zabbix agent configuration

**Info**

Log messages of this Router App can be viewed in the system log, see *Status* → *System Log*.

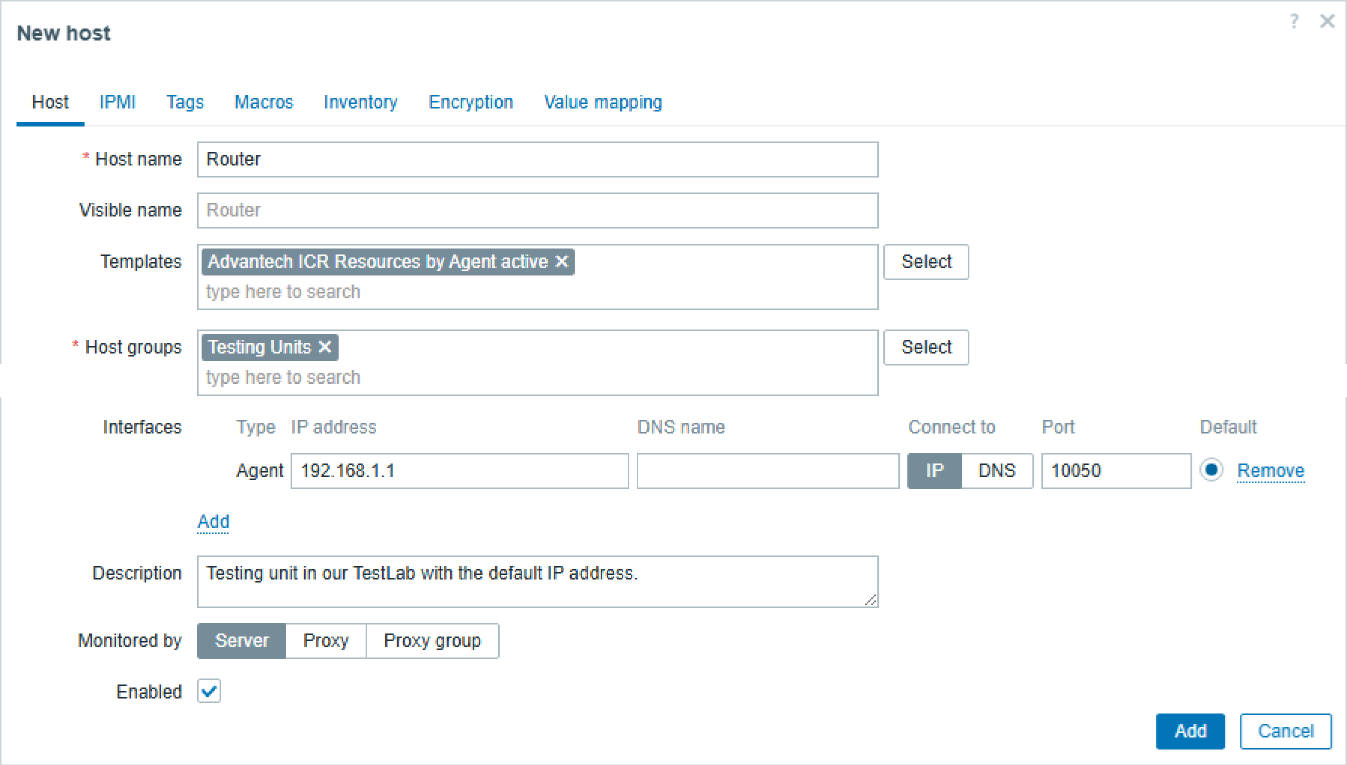
## 5.2 Configuring a Host in Zabbix

**Info**

This chapter describes a quick way to get Advantech router monitoring using Agent interface. There are many options for configuring the Zabbix server and frontend; please refer to the official manuals on the *Zabbix Documentation* pages for more detailed information.

The core logical configuration structure was described in Chapter *3.3 Zabbix Web Configuration Structure*. Here is the simplest way to add a router to the monitored hosts in the Zabbix web interface:

- We assume that you have already imported the templates from the Advantech YAML file into the Zabbix web interface under *Data collection* → *Templates*, as described in Chapter *3.2 Advantech Templates*.
- In the Zabbix web interface, navigate to *Monitoring* → *Hosts* and click the *Create host* button. A *New host* configuration form will appear. Figure 8 illustrates the configuration of a new host for agent-based monitoring, using the active agent template.



New host

| | |
|---|---|
| * Host name | Router |
| Visible name | Router |
| Templates | Advantech ICR Resources by Agent active ✕<br>type here to search [Select] |
| * Host groups | Testing Units ✕<br>type here to search [Select] |
| Description | Testing unit in our TestLab with the default IP address. |
| Monitored by | [Server] Proxy / Proxy group |
| Enabled | ☑ |

Interfaces

| Type | IP address | DNS name | Connect to | Port | Default | |
|---|---|---|---|---|---|---|
| Agent | 192.168.1.1 | | [IP] DNS | 10050 | ● | Remove |

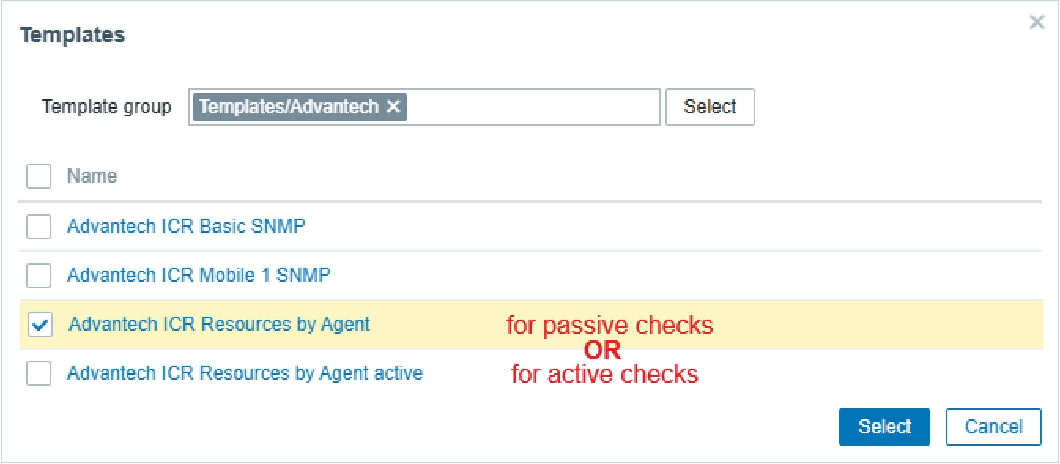
Add

[Add] [Cancel]

Figure 8: New host form with active agent interface

- Fill in the following fields in the form:
  - *Host name* – The host name, which **must exactly match the router's actual hostname**. The router's default hostname is `Router`, but it can be configured on the router under *Configuration* → *System* → *Identification*. You can optionally use the *Description* field below for a detailed host description.

- ○ *Templates* – Choose the imported **Advantech ICR Resources by Agent** template for passive monitoring, or the **Advantech ICR Resources by Agent active** template for active monitoring (you can filter for Advantech templates as shown in Figure 9):



Templates

Template group: Templates/Advantech ✕

□ Name

□ Advantech ICR Basic SNMP

□ Advantech ICR Mobile 1 SNMP

☑ Advantech ICR Resources by Agent — for passive checks

OR

□ Advantech ICR Resources by Agent active — for active checks

Figure 9: Choose template for active checks

- ○ *Host groups* – Select from existing host groups. You can create your own group in *Data collection* → *Host groups*.
- ○ *Interfaces* – Click on *Add* and choose the *Agent*.

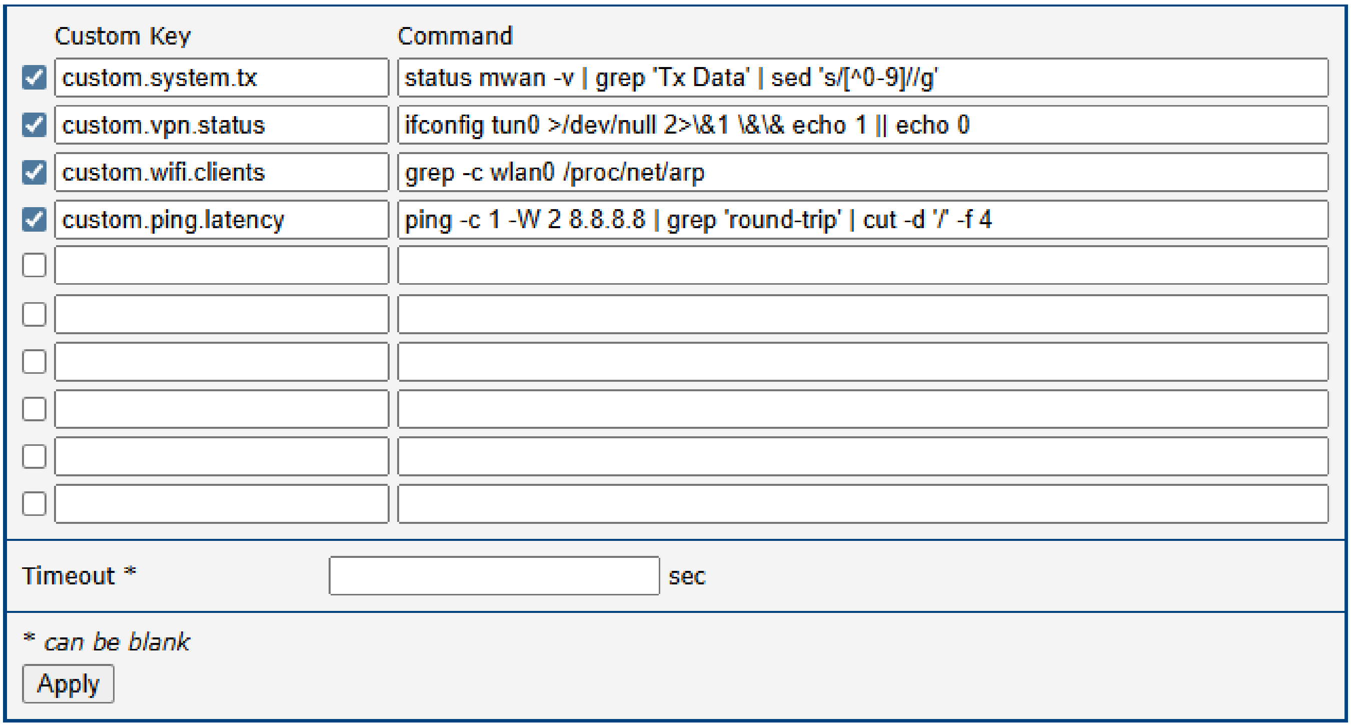
- Finally, click the *Add* button at the bottom of the form to save the host.

If everything is configured correctly, after a few minutes you should see:

- In *Monitoring* → *Hosts*, the `ZBX` (Agent) label in the *Availability* column for your host should turn green.
- Retrieved status information under *Monitoring* → *Latest data*. You can filter the results by your specific host name.

## 5.3 Custom Keys Configuration

In addition to the standard items, you can define custom items to be monitored by your agent in either active or passive mode. The configuration for these custom items is located in the bottom part of the configuration screen.



| | Custom Key | Command |
|---|---|---|
| ☑ | custom.system.tx | status mwan -v \| grep 'Tx Data' \| sed 's/[^0-9]//g' |
| ☑ | custom.vpn.status | ifconfig tun0 >/dev/null 2>\&1 \&\& echo 1 \|\| echo 0 |
| ☑ | custom.wifi.clients | grep -c wlan0 /proc/net/arp |
| ☑ | custom.ping.latency | ping -c 1 -W 2 8.8.8.8 \| grep 'round-trip' \| cut -d '/' -f 4 |
| ☐ | | |
| ☐ | | |
| ☐ | | |
| ☐ | | |
| ☐ | | |
| ☐ | | |

Timeout * sec

* can be blank

Figure 10: Zabbix agent configuration, custom keys

| Item | Description |
|---|---|
| *Custom Key* | The key for the custom Zabbix item. |
| *Command* | The command to execute, including any optional arguments. This must be a single command on a single line. The command is executed, and the first line of the standard output (stdout) is used as the resulting value. The *Command* field supports only a limited set of characters: double quotes `"` are not allowed, and dollar signs `$` must be escaped using `\$` . For more complex checks, create a shell script and use the *Command* field to execute it. |
| *Timeout* | Limits the execution time of a single check. The default value is 3 s. |

Table 6: Zabbix agent configuration, custom keys

On Figure 10, you can see examples of custom keys. The table below describes the meaning of the used custom keys and their data types.

| Key | Description | Type |
|---|---|---|
| *custom.system.tx* | Amount of transmitted data on the mobile WAN interface (TX). | Numeric (integer) |
| *custom.vpn.status* | Status of the VPN tunnel (1 = up, 0 = down). | Numeric (integer) |
| *custom.wifi.clients* | Number of connected Wi-Fi clients. | Numeric (integer) |
| *custom.ping.latency* | Network latency to the specified host (in milliseconds). | Numeric (float) |

Table 7: Custom keys examples description

### Adding the Custom Key to Zabbix Server

Once you have defined a *Custom Key* on the router, you must configure the Zabbix server to start collecting its data. This is done by adding a new Item to the corresponding Host or Template in the Zabbix frontend.

1. Navigate to *Data collection* → *Hosts* (or *Templates*).
2. Click on the **Items** link in the row of your Advantech router.
3. Click the **Create item** button in the top right corner.
4. Fill in the required item parameters:
   - **Name:** Enter a descriptive name for the item (e.g., *WiFi Clients Count*).
   - **Type:** Select **Zabbix agent** (for passive checks) or **Zabbix agent (active)** depending on your router's agent configuration.
   - **Key:** Enter the exact *Custom Key* you defined on the router (e.g., `custom.wifi.clients` ).
   - **Type of information:** Select the data type that matches the command's output. For example, choose *Numeric (unsigned)* for counts/statuses, *Numeric (float)* for latency, or *Character* for text strings.
   - **Update interval:** Define how often the server should poll this item (e.g., `1m` or `5m` ).
5. Click the **Add** button to save the configuration.

**Info**

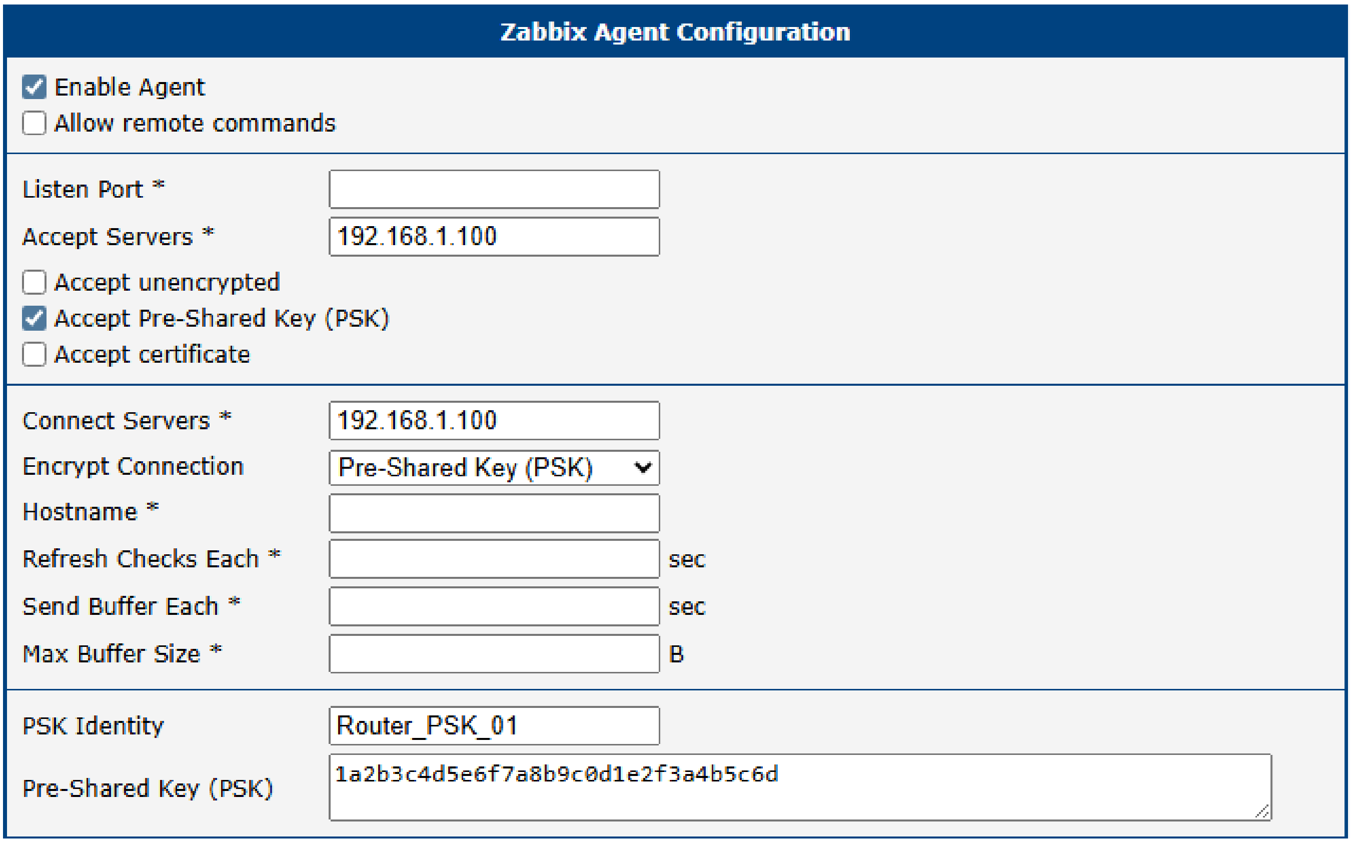
**Important:** The *Key* field in the Zabbix frontend must exactly match the *Custom Key* string defined in the router's agent configuration. Otherwise, the Zabbix server will report the item as "Not supported".

## 5.4 Agent TLS Configuration

In a production environment, it is highly recommended to use encryption for the connection, either by using a Pre-Shared Key (PSK) or a CA certificate.

### PSK Encryption

An example configuration in the *Zabbix Agent* Router App, where only PSK encryption is enabled, is shown in Figure 11. It is configured for both passive and active checks.



**Zabbix Agent Configuration**

☑ Enable Agent
☐ Allow remote commands

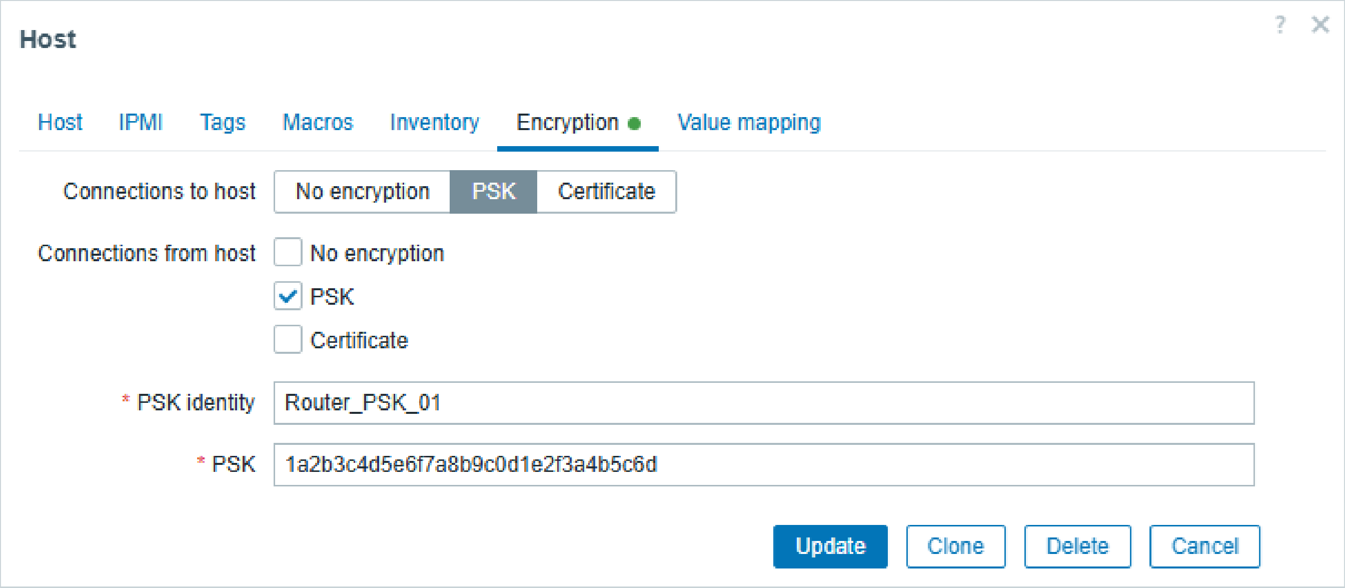
| | | |
|---|---|---|
| Listen Port * | | |
| Accept Servers * | 192.168.1.100 | |

☐ Accept unencrypted
☑ Accept Pre-Shared Key (PSK)
☐ Accept certificate

| | | |
|---|---|---|
| Connect Servers * | 192.168.1.100 | |
| Encrypt Connection | Pre-Shared Key (PSK) | |
| Hostname * | | |
| Refresh Checks Each * | | sec |
| Send Buffer Each * | | sec |
| Max Buffer Size * | | B |
| PSK Identity | Router_PSK_01 | |
| Pre-Shared Key (PSK) | 1a2b3c4d5e6f7a8b9c0d1e2f3a4b5c6d | |

Figure 11: PSK encryption on the router

Next, in the Zabbix web interface, configure the corresponding encryption settings for the host's Agent interface so that they exactly match the router's configuration, as shown in Figure 12.



Host

Connections to host: No encryption | **PSK** | Certificate

Connections from host:
☐ No encryption
☑ PSK
☐ Certificate

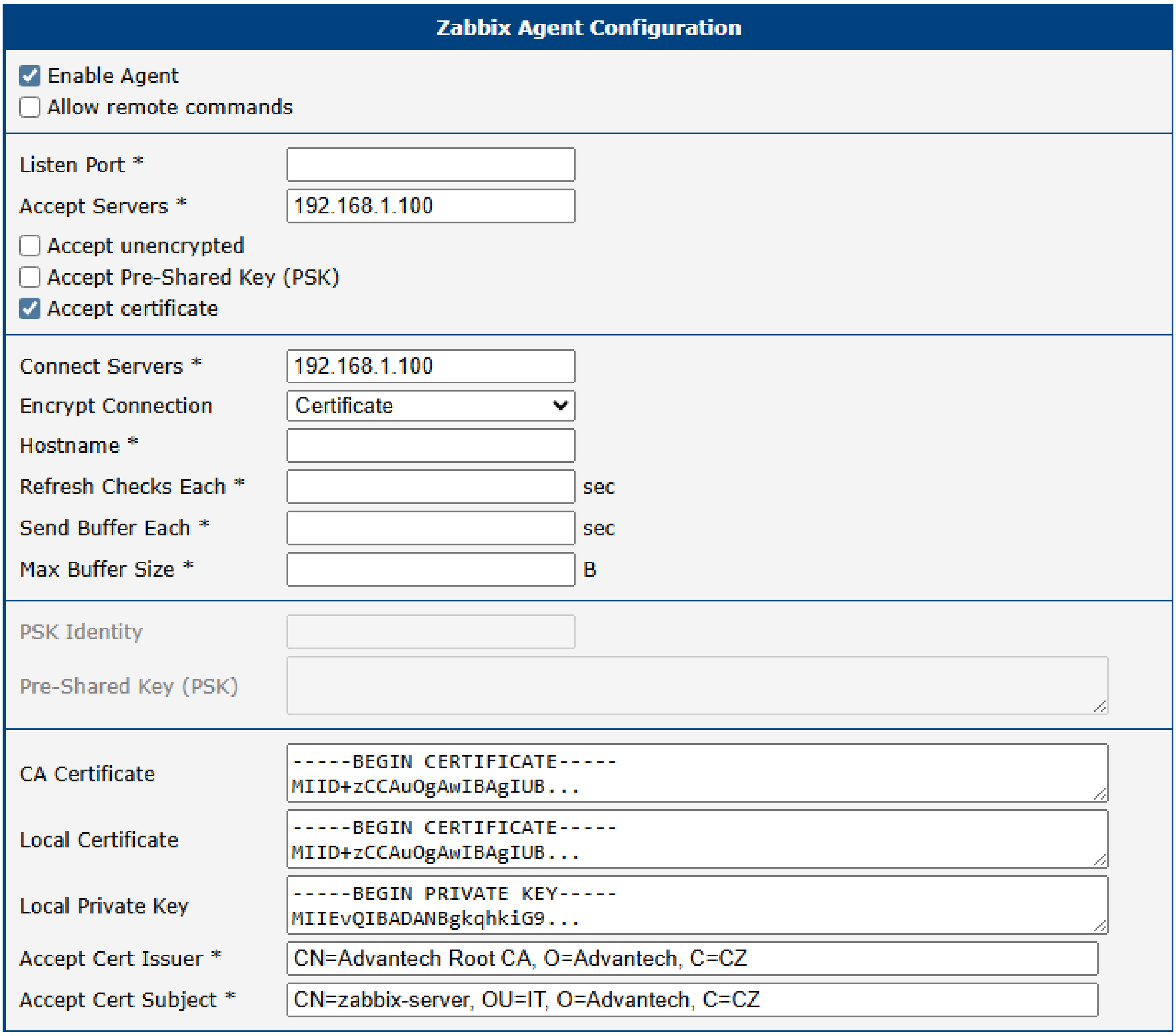
* PSK identity: Router_PSK_01

* PSK: 1a2b3c4d5e6f7a8b9c0d1e2f3a4b5c6d

Figure 12: PSK encryption on the Zabbix server

### CA Certificate Encryption

The highest level of security is achieved by using TLS certificate authentication (mTLS). An example configuration in the *Zabbix Agent* Router App, where only CA Certificate encryption is enabled, is shown in Figure 13. It is configured for both passive and active checks.



**Zabbix Agent Configuration**

☑ Enable Agent
□ Allow remote commands

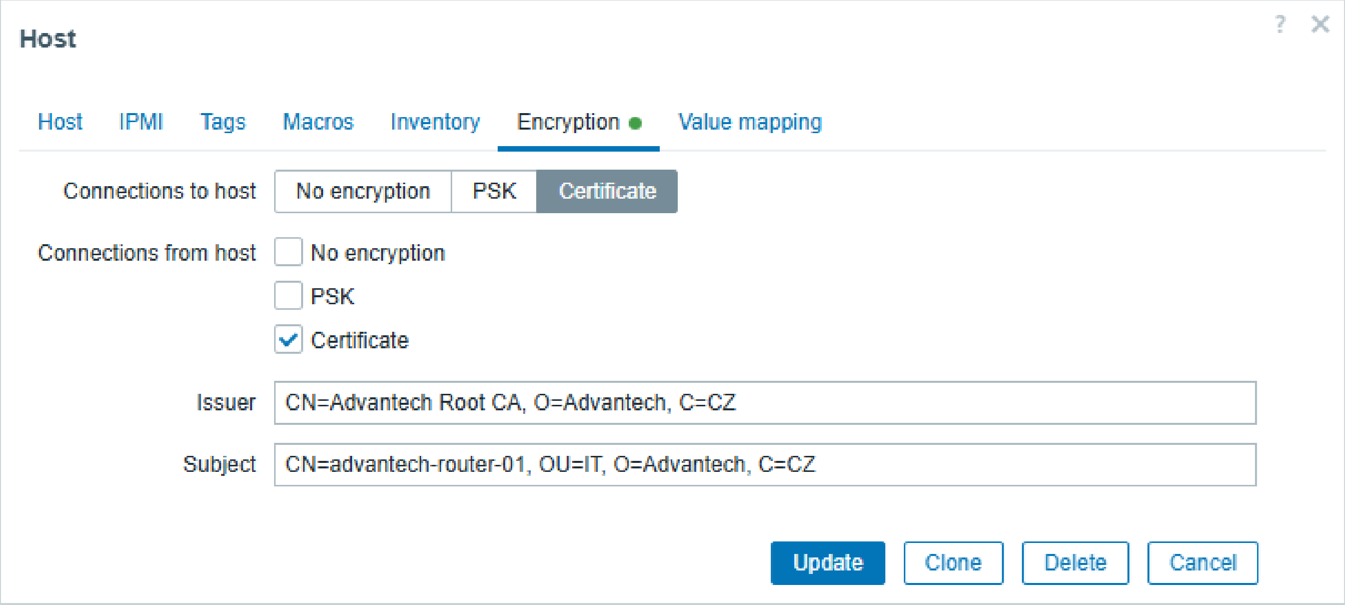
| | |
|---|---|
| Listen Port * | |
| Accept Servers * | 192.168.1.100 |

□ Accept unencrypted
□ Accept Pre-Shared Key (PSK)
☑ Accept certificate

| | | |
|---|---|---|
| Connect Servers * | 192.168.1.100 | |
| Encrypt Connection | Certificate | |
| Hostname * | | |
| Refresh Checks Each * | | sec |
| Send Buffer Each * | | sec |
| Max Buffer Size * | | B |
| PSK Identity | | |
| Pre-Shared Key (PSK) | | |
| CA Certificate | -----BEGIN CERTIFICATE-----<br>MIID+zCCAuOgAwIBAgIUB... | |
| Local Certificate | -----BEGIN CERTIFICATE-----<br>MIID+zCCAuOgAwIBAgIUB... | |
| Local Private Key | -----BEGIN PRIVATE KEY-----<br>MIIEvQIBADANBgkqhkiG9... | |
| Accept Cert Issuer * | CN=Advantech Root CA, O=Advantech, C=CZ | |
| Accept Cert Subject * | CN=zabbix-server, OU=IT, O=Advantech, C=CZ | |

Figure 13: Certificate encryption on the router

Next, in the Zabbix web interface for the host configuration, add an Agent Interface, define the encryption settings to align exactly with the agent's configuration, and link the host to one or more Agent templates. The encryption settings on the Zabbix server are shown in Figure 14.



Connections to host: No encryption | PSK | [✓ Certificate]

Connections from host: □ No encryption
□ PSK
☑ Certificate

Issuer: CN=Advantech Root CA, O=Advantech, C=CZ

Subject: CN=advantech-router-01, OU=IT, O=Advantech, C=CZ

Figure 14: Certificate encryption on the Zabbix server

To use TLS certificates, the Zabbix server needs its own certificates ( `TLSCAFile` , `TLSCertFile` , and `TLSKeyFile` ) properly set up, as described in the Zabbix Manual. See https://www.zabbix.com/documentation/current/manual/encryption/using_certificates.

**Warning**

- Configuring certificates in the Zabbix web interface only defines the verification rules. For TLS encryption to work, the Zabbix server must have its own certificate, private key, and the CA certificate physically stored on its filesystem. These files must be referenced in the `zabbix_server.conf` file using the `TLSCAFile` , `TLSCertFile` , and `TLSKeyFile` parameters. Refer to the official Zabbix documentation for detailed instructions on securing the server.
- The purpose of the certificate must include "server authentication". When generated by OpenSSL, the `extendedKeyUsage = serverAuth` attribute must be set.

## 5.5 Items Supported by Zabbix Agent

Standard Zabbix items (checks) are described in detail `https://www.zabbix.com/documentation/current/manual/config/items/itemtypes/zabbix_agent`

Zabbix documentation also indicates which of the items are supported on various platforms: `https://www.zabbix.com/documentation/6.0/en/manual/appendix/items/supported_by_platform`

The following table complements that information and explains which of the standard agent items are supported on Advantech cellular routers.

| Item Key | Support |
|---|---|
| agent.hostname | Both |
| agent.ping | Both |
| agent.variant | Both |
| agent.version | Both |
| kernel.maxfiles | Both |
| kernel.maxproc | Both |
| kernel.openfiles | Both |
| log[file,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>]<br>example:<br>*log[/var/log/messages,“authentication failure”„,skip„]* | Active only |
| log.count[file,<regexp>,<encoding>,<maxproclines>,<mode>,<maxdelay>] | Active only |
| logrt[file_regexp,<regexp>,<encoding>,<maxlines>,<mode>,<output>,<maxdelay>,<options>] | Active only |
| logrt.count[file_regexp,<regexp>,<encoding>,<maxproclines>,<mode>,<maxdelay>,<options>] | Active only |
| net.dns[<ip>,name,<type>,<timeout>,<count>,<protocol>] | Both |
| net.dns.perf[<ip>,name,<type>,<timeout>,<count>,<protocol>] | Both |
| net.dns.record[<ip>,name,<type>,<timeout>,<count>,<protocol>] | Both |
| net.if.collisions[if] | Both |
| net.if.discovery | Both |
| net.if.in[if,<mode>] | Both |
| net.if.list | Both |
| net.if.out[if,<mode>] | Both |
| net.if.total[if,<mode>] | Both |
| net.tcp.listen[port] | Both |
| net.tcp.port[<ip>,port] | Both |
| net.tcp.service[service,<ip>,<port>] | Both |
| net.tcp.service.perf[service,<ip>,<port>] | Both |
| net.tcp.socket.count[<laddr>,<lport>,<raddr>,<rport>,<state>] | Both |
| net.udp.listen[port] | Both |
| net.udp.service[service,<ip>,<port>] | Both |
| net.udp.service.perf[service,<ip>,<port>] | Both |
| net.udp.socket.count[<laddr>,<lport>,<raddr>,<rport>,<state>] | Both |
| proc.cpu.util[<name>,<user>,<type>,<cmdline>,<mode>,<zone>] | Both |
| proc.get[<name>,<user>,<cmdline>,<mode>] | Both |
| proc.mem[<name>,<user>,<mode>,<cmdline><memtype>] | Both |
| proc.num[<name>,<user>,<state>,<cmdline><zone>] | Both |
| proc.info[process,<attribute>,<type>] | Both |
| system.boottime | Both |
| system.cpu.discovery | Both |
| system.cpu.intr | Both |
| system.cpu.load[<cpu>,<mode>] | Both |

Table 8: Agent items support

| Item Key | Support |
|---|---|
| system.cpu.num[<type>] | Both |
| system.cpu.switches | Both |
| system.cpu.util[<cpu>,<type>,<mode>] | Both |
| system.hostname | Both |
| system.hw.cpu[<cpu>,<info>] | Both |
| system.hw.macaddr[<interface>,<format>] | Both |
| system.localtime[<type>] | Passive only |
| system.run[command,<mode>]<br>example:<br>*system.run*[ls /] | If enabled |
| system.sw.arch | Both |
| system.sw.os[<info>] | Both |
| system.sw.os.get | Both |
| system.uname | Both |
| system.uptime | Both |
| vfs.dir.count[dir,<regex_incl>,<regex_excl>,<types_incl>,<types_excl>,<max_depth>,<min_size>, <max_size>,<min_age>,<max_age>]<br>example:<br>*vfs.dir.count[/dev]* | Both |
| vfs.dir.get[dir,<regex_incl>,<regex_excl>,<types_incl>,<types_excl>,<max_depth>,<min_size>, <max_size>,<min_age>,<max_age>,<regex_excl_dir>] | Both |
| vfs.dir.size[dir,<regex_incl>,<regex_excl>,<mode>,<max_depth>] | Both |
| vfs.file.cksum[file] | Both |
| vfs.file.contents[file,<encoding>] | Both |
| vfs.file.exists[file,<types_incl>,<types_excl>] | Both |
| vfs.file.get[file] | Both |
| vfs.file.md5sum[file] | Both |
| vfs.file.owner[file,<ownertype>,<resulttype>] | Both |
| vfs.file.permissions[file] | Both |
| vfs.file.regexp[file,regexp,<encoding>,<output>] | Both |
| vfs.file.regmatch[file,regexp,<encoding>] | Both |
| vfs.file.size[file] | Both |
| vfs.file.time[file,<mode>] | Both |
| vfs.fs.discovery | Both |
| vfs.fs.size[fs,<mode>] | Both |
| vm.memory.size[<mode>] | Both |
| web.page.get[host,<path>,<port>] | Both |
| web.page.perf[host,<path>,<port>] | Both |
| web.page.regexp[host,<path>,<port>,regexp,<length>,<output>] | Both |

Table 8: Agent items support (continued)

In addition to the above, the following Advantech specific items are supported:

| Item Key | Description |
|---|---|
| vfs.settings.discovery | List of /etc/settings.* and /opt/*/etc/settings files for autodiscovery |
| vfs.settings.value[name,parameter]<br>example:<br>*vfs.settings.value[wifi_ap, WIFI_AP_SSID]* | Retrieves a single value from the router config /etc/settings.[name] |
| vfs.settings.umod[name,parameter]<br>example:<br>*vfs.settings.umod[gps, MOD_GPS_ENABLED]* | Retrieves a single value from a router app config /opt/[name]/etc/settings |

Table 9: Specific items support

# 6. Troubleshooting

This chapter provides guidance for diagnosing and resolving common issues when integrating Advantech routers with Zabbix. Problems are organized by observed symptoms to help quickly identify the root cause and apply the appropriate solution.

| Symptom | Likely cause |
|---|---|
| No data | Incorrect interface configuration or missing template |
| Host unavailable | Network connectivity issue or firewall blocking communication |
| Items not supported | Incorrect item key or missing feature on the router |

## 6.1 No Data in Zabbix

If no data is displayed in *Monitoring* → *Latest data*, check the following:

- Verify that the host is correctly configured in the Zabbix web interface.
- Ensure that the correct template is assigned to the host.
- Check network connectivity between the router and the Zabbix server (for example, using `ping`).
- Confirm that the selected interface type (SNMP or Agent) matches the router configuration.
- Verify that item update intervals are not too long.

## 6.2 Host Not Available

If the host shows as unavailable (red indicator):

- Verify that the IP address or hostname is correct.
- Check firewall rules on both the router and the server.
- Ensure that required ports are open:
  - SNMP: UDP 161
  - Zabbix Agent: TCP 10050 (passive) or 10051 (active)
- Confirm that the monitoring service (SNMP or Agent) is enabled on the router.

## 6.3 SNMP Issues

### SNMP Not Responding

- Ensure that the SNMP service is enabled on the router.
- Verify that the SNMP version configured in Zabbix matches the router.

- Check that the community string (SNMPv1/v2) or credentials (SNMPv3) are correct.
- Test SNMP manually using tools such as `snmpwalk` .

### SNMPv3 Authentication Errors

- Verify that authentication and privacy settings match exactly on both sides.
- Ensure that the same authentication protocol (e.g., SHA-512) and encryption method (e.g., AES) are used.
- Check username and passwords for typos.

## 6.4 Zabbix Agent Issues

### Agent Not Connecting

- Verify that the Zabbix Agent Router App is installed and running.
- Check the *Connect Servers* configuration for active checks.
- Ensure that the router can reach the Zabbix server (network routing, NAT).
- Verify that the *Hostname* matches the host name configured in Zabbix exactly.

### Passive Checks Not Working

- Ensure that the Zabbix server IP address is listed in *Accept Servers*.
- Verify that the correct port (default 10050) is open and listening.
- Check firewall settings on the router.

### Items Marked as “Not supported”

- Verify that the item key is correct and matches the router configuration.
- Ensure that required features or Router Apps are installed.
- Check for syntax errors in custom commands.
- Review Zabbix server logs for detailed error messages.

## 6.5 Encryption Issues

### PSK Authentication Fails

- Ensure that the PSK identity and key match exactly on both the router and server.
- Check that the correct encryption mode (PSK) is selected in the Zabbix host configuration.

### Certificate Authentication Fails

- Verify that the CA certificate is trusted on both sides.
- Ensure that certificate subject and issuer fields match the configuration.
- Confirm that the certificate includes the correct purpose (client/server authentication).
- Check that certificate files are correctly configured on the Zabbix server (TLSCAFile, TLSCertFile, TLSKeyFile).

## 6.6 Performance Issues

- Reduce item polling frequency if the server is overloaded.
- Increase the number of SNMP pollers in the Zabbix server configuration.
- Prefer active agent checks for large deployments.

## 6.7 Log Analysis

Logs are essential for troubleshooting:

- Zabbix server logs (e.g., `/var/log/zabbix/zabbix_server.log` )
- Router system log (*Status* → *System Log*)

Review logs for errors related to connectivity, authentication, or unsupported items.

# 7. Related Resources

You can obtain all product-related documents, software updates, and supplementary materials on the Advantech *Engineering Portal* at *icr.advantech.com*.

For easy access to specific resources, please refer to the following sections of the portal:

- **Router Support Materials:** To access your router's supporting documents (such as the *Hardware Manual* and *Configuration Manual*), the latest firmware, or other technical resources, navigate to *Support* → *Router Models*. Locate your specific model and select the appropriate tab under the *Documents to download* section. Available tabs include *Brochures*, *Manuals*, *Certificates*, *Firmware*, *Images/3D Models*, *PCN/SA*, and *Others*.
- **Router Apps:** To extend your router's functionality, installation packages and comprehensive manuals for various extension modules are available by navigating to *Download* → *Router Apps*.
- **Application Notes:** For detailed guides, configuration examples, and step-by-step instructions for implementing specific networking features and use cases, navigate to *Download* → *Application Notes*.
- **Development Documents:** If you are interested in custom scripting, programming your own applications, or compiling custom modules, navigate to *Development* page.