

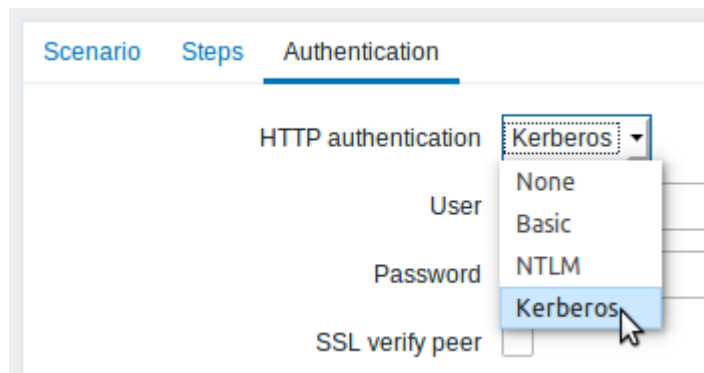
## 5 What's new in Zabbix 4.4.0

Zabbix 4.4.0 is not released yet.

### Kerberos authentication

Kerberos authentication is now supported in:

- [Web monitoring](#)
- [HTTP items](#)



### Low-level discovery

#### Low-level discovery of block devices

Low-level discovery of block devices and their type is supported using a new built-in discovery key:

```
vfs.dev.discovery
```

The discovery will return a JSON with the values of two macros - `{#DEVNAME}` and `{#DEVTYPE}`, identifying the block device name and type respectively.

These macros can be used to create item prototypes using the `vfs.dev.read[]` and `vfs.dev.write[]` agent items, i.e. `vfs.dev.read[{#DEVNAME}, sps]`. See also: [Discovery of block devices](#).

#### Low-level discovery of systemd services

Low-level discovery of systemd units (services, by default) is supported using a new built-in discovery key:

```
systemd.unit.discovery
```

This item key is supported by the new Zabbix agent version written in Go language.

The discovery will return a JSON with the values of several macros, identifying various systemd unit properties. These macros can be used to create item prototypes using the new

`systemd.unit.info[]` item key, for example  
`systemd.unit.info["{#UNIT.NAME}", LoadState]`.

See also:

- [Discovery of systemd services](#)
- [Zabbix agent items](#) (`systemd.unit.info[]` item key)

## Extended item value preprocessing options

### Custom error handling

Custom error handling is now also available for the following preprocessing steps:

- Check for error in JSON
- Check for error in XML
- Check for error using regular expression

The *Custom on fail* checkbox is available in these preprocessing steps for regular items and item prototypes. For low-level discovery rules the *Custom on fail* checkbox is made available for 'Check for error in JSON' and 'Check for error in XML' preprocessing steps.

In a typical use case of custom error handling, data can be skipped if an error message is found.

### XML-related preprocessing options added to LLD

*XML Xpath* and *Check for error in XML* preprocessing options have been added to [low-level discovery rules](#).

## Secure autoregistration

Previously all communications during agent autoregistration were performed unencrypted. In the new version a secure way of autoregistration is possible by configuring PSK-based authentication with encrypted connections.

The level of encryption is configured globally in *Administration* → [General](#), in the new Autoregistration section accessible through the dropdown to the right. It is possible to select no encryption, TLS encryption with PSK authentication or both (so that some hosts may register without encryption while others through encryption):



Authentication by PSK is verified by Zabbix server before adding a host. If successful, the host is added and [Connections from/to host](#) are set to 'PSK' only with identity/pre-shared key the same as in the global autoregistration setting.

## Autoregistration with DNS name

It is now possible to specify that the host should be autoregistered with the DNS name as the default agent interface. To do that, the DNS name should be specified/returned as the value of either 'HostInterface' or 'HostInterfaceItem' configuration parameters. Note that if the value of one of the two parameters changes, the autoregistered host interface is updated. So it is possible to update the default interface to another DNS name or an IP address. For the changes to take effect though, the agent has to be restarted. 'HostInterface' or 'HostInterfaceItem' configuration parameters are supported since Zabbix 4.4.

## Longer host names allowed in discovery

Maximum allowed length of a host name has been lifted from 64 characters to 128 characters in host discovery and active agent auto-registration.

## New templates

New official templates are available for:

- **Nginx:**
  - *Template App Nginx by Zabbix agent* - collects metrics by polling [ngx\\_stub\\_status\\_module](#) locally with Zabbix agent (see [description](#));
  - *Template App Nginx by HTTP* - collects metrics by polling [ngx\\_stub\\_status\\_module](#) with HTTP agent remotely (see [description](#)).
- **Apache:**
  - *Template App Apache by Zabbix agent* - collects metrics by polling [mod\\_status](#) locally with Zabbix agent (see [description](#));
  - *Template App Apache by HTTP* - collects metrics by polling [mod\\_status](#) with HTTP agent remotely (see [description](#)).

You can get these templates:

- In *Configuration* → *Templates* in new installations;
- If you are upgrading from previous versions, you can find these templates in the `templates` directory of the downloaded latest Zabbix version. While in *Configuration* → *Templates* you can import them manually from this directory.

## Jabber, Ez Texting media types removed

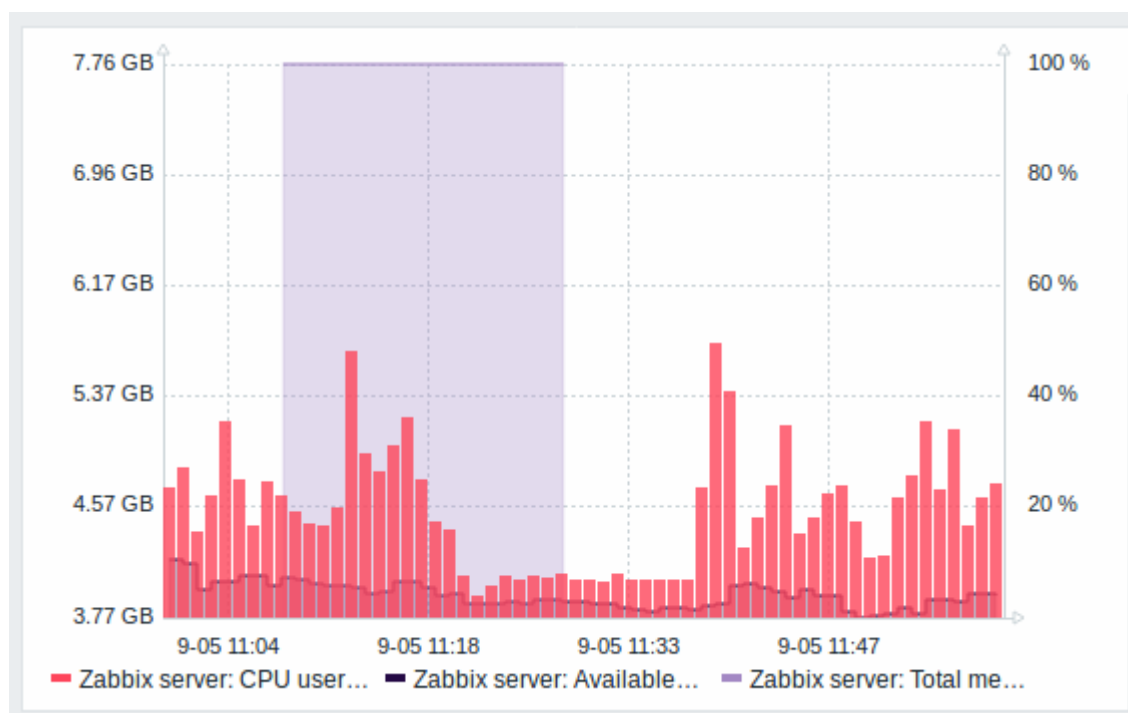
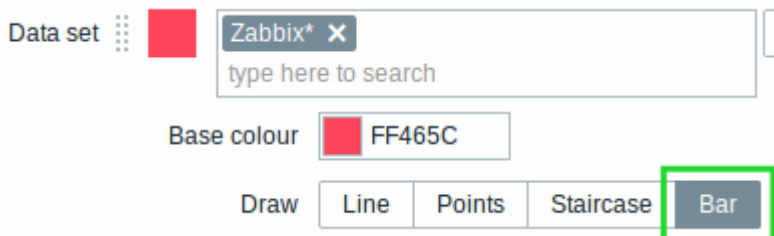
Jabber and Ez Texting [media types](#) for delivering notifications have been removed. If these media types are present in your existing installation, during the upgrade they will be replaced by a script media type with all relevant parameters preserved.

## Frontend

## Bar graphs

Graph values in the graph [widget](#) can now be displayed as bars.

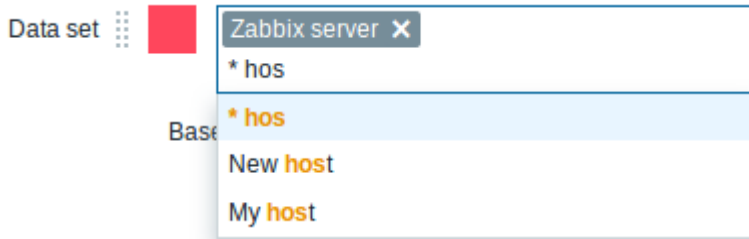
To display values as bars, when configuring the data set of the graph, select *Bar* as the draw option:



## Multi-select field for selecting graph items

Previously multiple item selection in the new graph widget (introduced in Zabbix 4.0) worked by separating the items with a comma. That created a limitation for selecting items already having a comma, a legitimate symbol, in the name.

To lift this limitation, in the new version, both host and item selection is implemented as multi-select fields where a comma is no longer needed as a separator. The functionality to specify host/item names with a wildcard is also retained. To specify a name with wildcard, just enter the string manually and press *Enter*. While you are typing, note how all matching entities are displayed in the dropdown.



### Operational data of problems

Time	Recovery time	Status	Info	Host	Problem • Severity	Operational data
06:50:35		PROBLEM		Zabbix server	Zabbix discoverer processes more than 75% busy	Current value: 100 %

A new *Operational data* column has been added to *Monitoring* → *Problems* and the *Problems* dashboard widget, next to the *Problem* column. It is available if *Show operational data* is selected in the filter or the widget configuration.

This column is capable of displaying a live view of operational data based on the definition (arbitrary strings plus macros) given in the [trigger configuration](#), which now has a new field *Operational data*.

Trigger	Tags	Dependencies
* Name	Zabbix discoverer processes more than 75% busy	
Operational data	Current value: {ITEM.LASTVALUE}	

Importantly this gives the opportunity to see the **latest values** of selected items from the trigger expression that caused the problem, using the {ITEM.LASTVALUE<1-9>} macro. Note that the *Operational data* column replaces the *Latest values* column.

*Operational data* can also be included in notifications using the new {EVENT.OPDATA} macro.

### Instant editing for dashboard widgets

Dashboard widget editing can now be accessed with one mouse click.



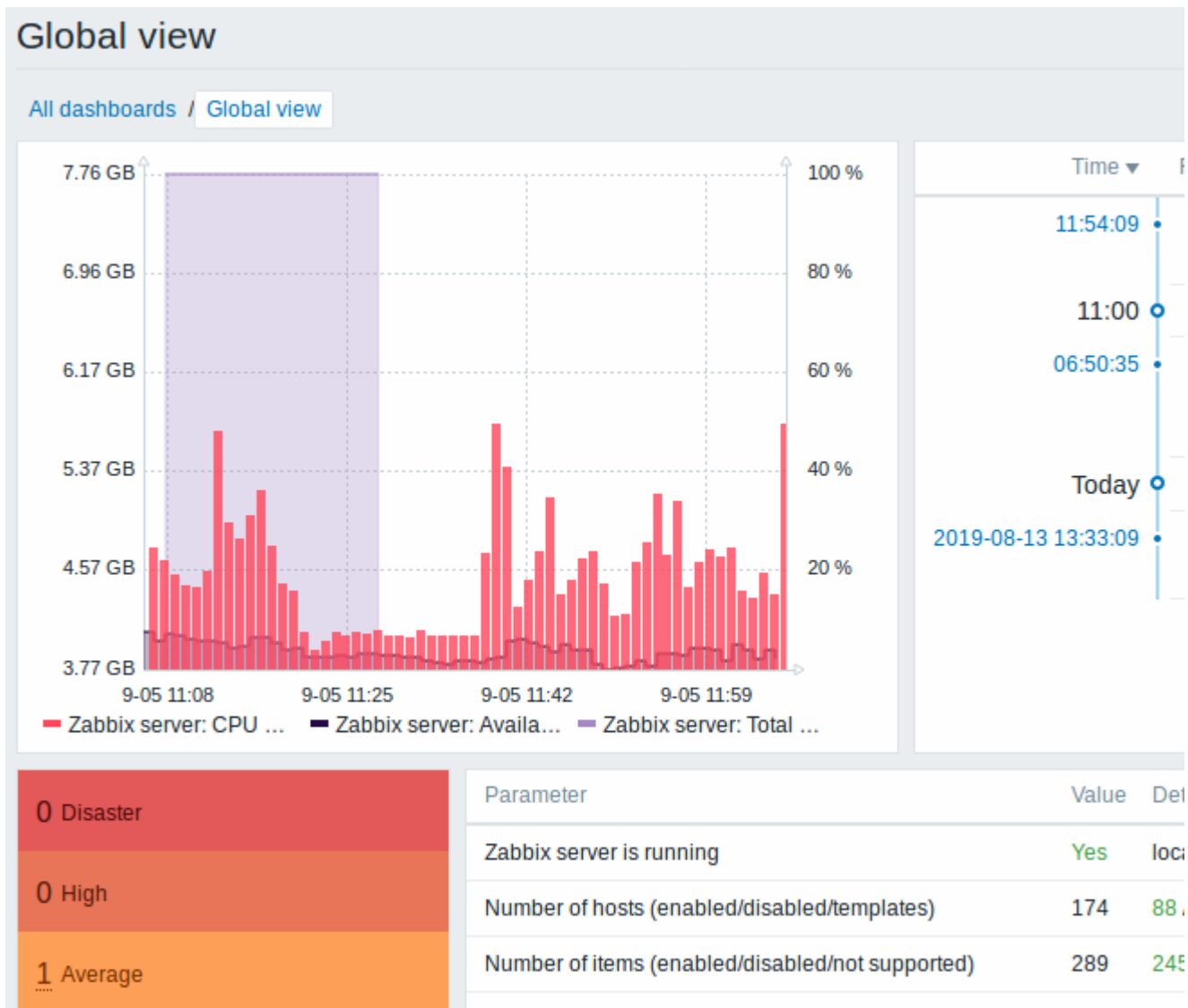
Once the editing button is clicked, the widget editing form is opened and the whole dashboard goes into editing mode.

## 24 column grid in dashboard

The horizontal limit of allowed dashboard widgets has been raised from 12 to 24 columns.

## Hidden widget headers

Widget headers can now be hidden allowing to display the widget content only and save space:



The widget header can be hidden in the [widget configuration](#), by unchecking the new *Show header* option.

When a widget header is hidden it still can be viewed by positioning the mouse over the widget. On mouseover the widget header slides up and becomes visible again.

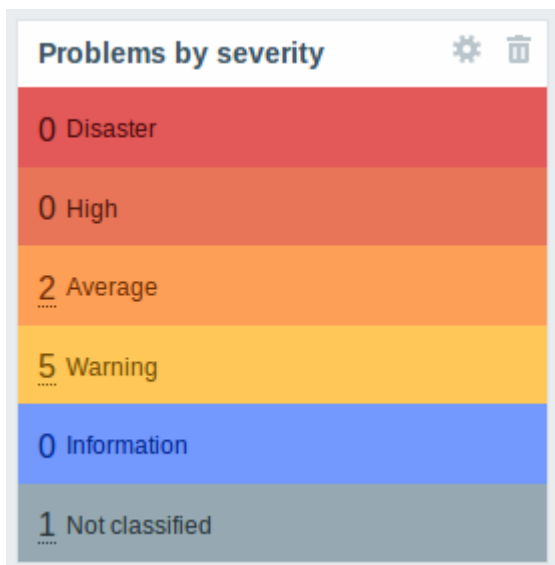
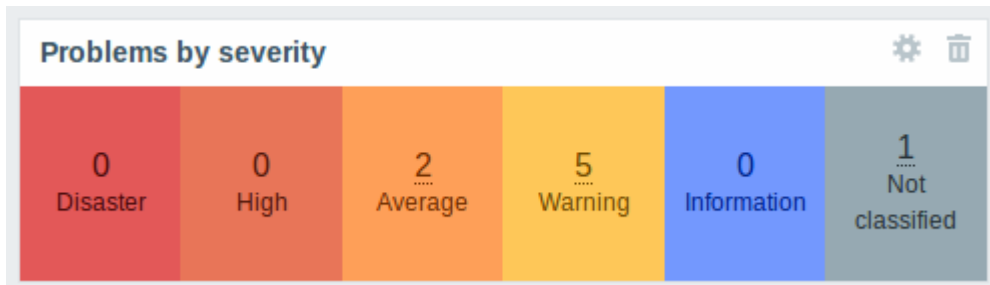
## Problems by severity widget extended

The *Problems by severity* widget has been extended with the option to show problem totals for multiple selected host groups and hosts:

Show

Layout

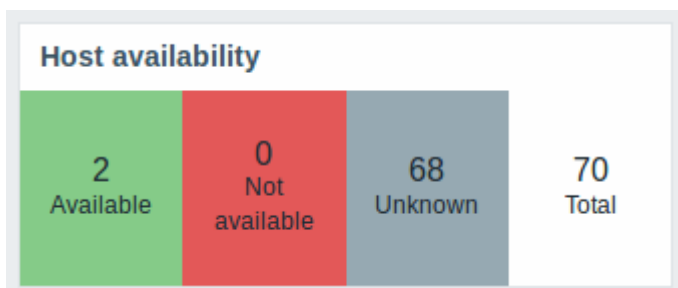
The totals can be displayed in horizontally or vertically stacked blocks:

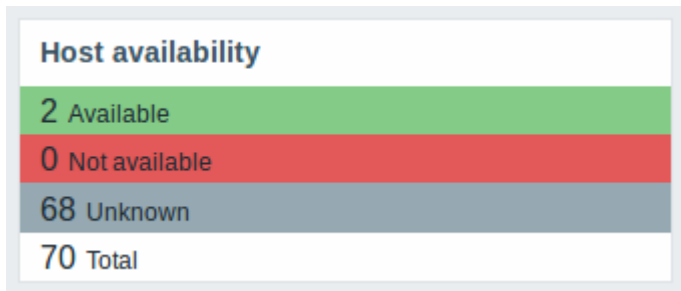


### New host availability widget

A new *Host availability* widget has been added to available dashboard widgets. This widget is similar as the *Host info* screen element and displays high-level statistics of host availability based on the selected host groups.

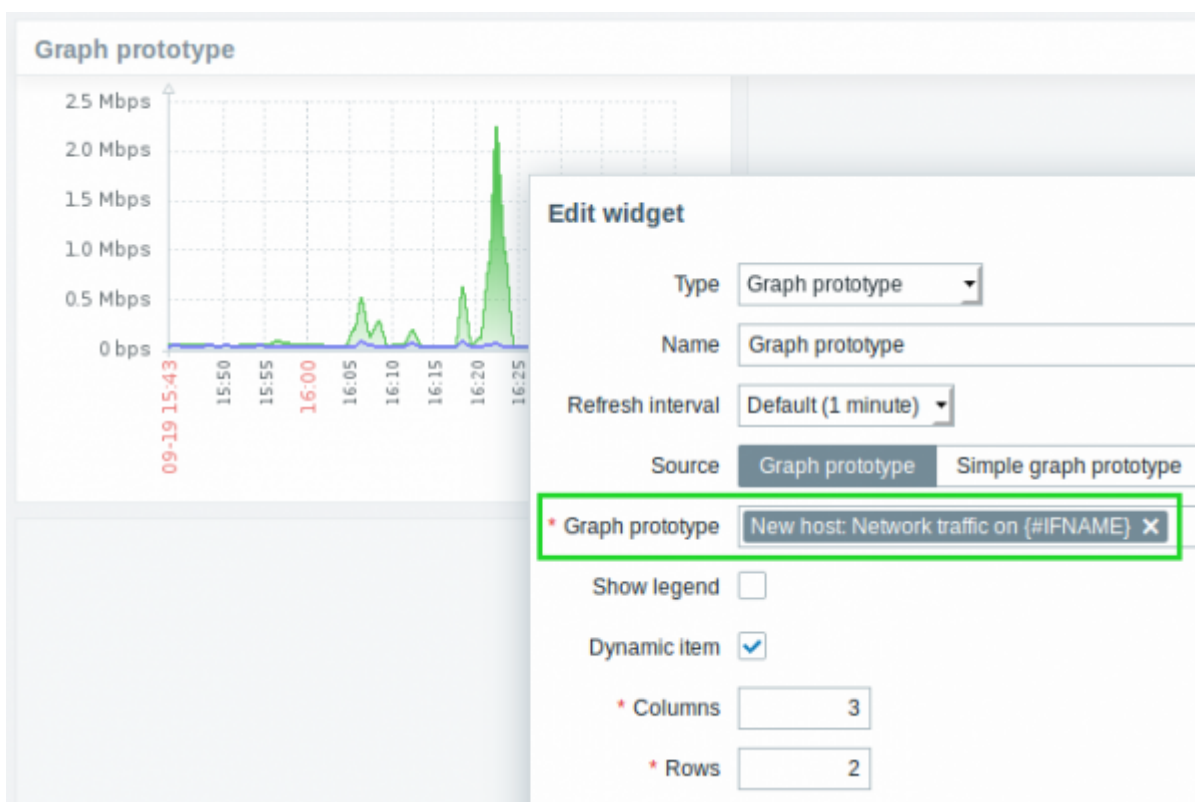
The availability statistics can be displayed in horizontally or vertically stacked blocks:





### New graph prototype widget

A new [graph prototype](#) widget has been added to the dashboard. In it you can display a grid of one or more graphs created from a single graph prototype by low-level discovery.



### Monospaced fonts in trigger expression/command fields

For better readability, text area fonts have been changed to monospace in the following locations:

- Trigger expression and recovery expression field
- Executed script field of SSH/Telnet agent items
- Formula field of calculated items
- SQL query field of database monitor items
- Command field of global scripts
- Remote command field in action configuration

```
* Expression {0017:vfs.file.cksum[/etc/passwd].diff(0)}>0
```





{last()}<0 in the example):

```
<hosts>
  <host>
    <host>Host</host>
    ...
    <items>
      <item>
        <name>Item</name>
        <type>0</type>
        <snmp_community/>
        <snmp_oid/>
        <key>item.key</key>
        ...
        <triggers>
          <trigger>
            <expression>{last()}<0</expression>
            <recovery_mode>0</recovery_mode>
            <recovery_expression/>
            <name>Item value not 0</name>
            <correlation_mode>0</correlation_mode>
            <correlation_tag/>
            <url/>
            <status>0</status>
            <priority>2</priority>
            <description/>
            <type>0</type>
            <manual_close>0</manual_close>
            <dependencies/>
            <tags/>
          </trigger>
        </triggers>
      </item>
    </items>
  </host>
</hosts>
```

The same change affects simple trigger prototypes that are placed under <item\_prototype><trigger\_prototypes>.

However, triggers that are more complex and contain several host items are listed within separate <triggers> tags, as before.

## Miscellaneous

## Performance

## Items table split

The `items` table was previously used by both frontend and the server, resulting in undesirable locking of rows at times when, for example, the server would update fields related to 'log' items. To resolve this situation, realtime fields (*lastlogsize*, *state*, *mtime*, *error*) have been split into a separate table called `item_rtdata`.

From:

<https://www.zabbix.com/documentation/4.4/> - **Zabbix Documentation 4.4**

Permanent link:

<https://www.zabbix.com/documentation/4.4/manual/introduction/whatsnew440>

Last update: **2019/09/19 13:47**

