Realtime

New in Tiki25. Use built-in websockets server within Tiki to enable realtime communication features of the app.

This is optional. If your server doesn't support this, it's OK. If you have it, supported features will be faster and smoother. For now, it is used for Tiki Manager Package, and will later be added to other Tiki Features.

Powered by https://packagist.org/packages/cboden/Ratchet and https://reactphp.org

This will require a special server configuration (possibly root access). tiki-check.php will indicate if server is correctly set up.

In this documentation, we are going to configure Virtualmin (but any other server can be used), and then we will check if it worked.

Configuration

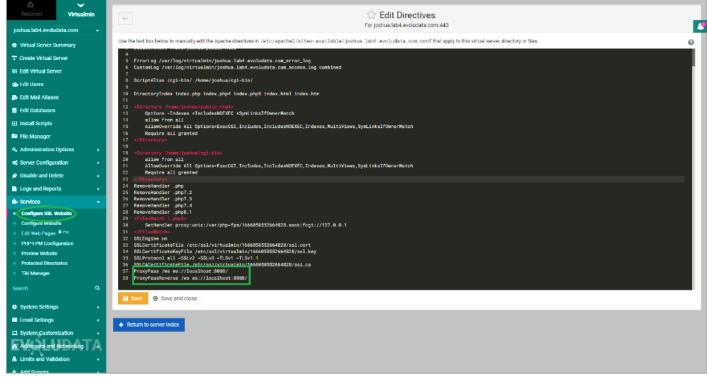
Requirements

We already have a Tiki instance up and running, on Virtualmin, now we need to take into account a few more things: **Proxy Settings**, **Realtime Service**, **proxy_wstunnel**, **Realtime** feature, and **Realtime port** in Tiki in order to get Realtime working.

1. Proxy Settings

First, let us edit Apache directives and add the ProxyPass and the ProxyPassReverse settings in the SSI virtual host. We make both point to */ws ws://localhost:8080/*

To do this, we go to *Virtualmin -> Services -> Configure SSL Website -> Edit Directives*. In the text box that opens, we add ProxyPass /ws ws://localhost:8080/ and ProxyPassReverse /ws ws://localhost:8080/ at the end, like shown in the picture below:





2. Realtime Service

At this step, we need to create a Systemd service to start the WS server.

N.B: Start the WS server with the same user that Tiki web requests run as (to avoid permission issues) - e.g. sudo - u www-data php tiki-realtime.php

The process is simple. Just go to *Webmin -> System -> Bootup and Shutdown -> create a new systemd service*.

Webmin Virtualmin	0		Bootup and Shutdown			7
arch Q						
Webmin 4	S :	Select all 🛛 Invert selection 🖉	Service description	Service status	Start at boot?	Running now?
System 🗸		apache-htcacheclean	Start the htcacheclean helper	Unknown	Start at boot? No	Running now?
Bootup and Shutdown		apache-htcacheclean.service	Disk Cache Cleaning Daemon for Apache HTTP Server	Inactive (dead)	No	No
Change Passwords		apache2	Start the web server	Unknown	Yes	Yes
		apache2.service	The Apache HTTP Server	Active (running)	Yes	Yes
Disk and Network Filesystems		apparmor	AppArmor init script. This script loads all AppArmor profiles.	Unknown	No	Unknown
Disk Quotas		apparmor.service	Load AppArmor profiles	Active (exited)	Yes	Yes
Filesystem Backup		apt-daily-upgrade.service	Daily apt upgrade and clean activities	Inactive (dead)	Always	No
Log File Rotation		apt-daily-upgrade.timer	Daily apt upgrade and clean activities	Active (waiting)	Yes	Yes
MIME Type Programs		apt-daily.service	Daily apt download activities	Inactive (dead)	Always	No
PAM Authentication		basic.target	Basic System	Active (active)	Always	Yes
		boot-complete.target	Boot Completion Check	Inactive (dead)	Always	No
Running Processes		certbot.service	Certbot	Inactive (dead)	Always	No
Scheduled Cron Jobs		certbot.timer	Run certbot twice daily	Active (waiting)	Yes	Yes
Software Package Updates		console-getty.service	Console Getty	Inactive (dead)	No	No
Software Packages		console-setup.service	Set console font and keymap	Active (exited)	Yes	Yes
System Documentation		console-setup.sh	Set console font and keymap	Unknown	Yes	No
		cron	cron is a standard UNIX program that runs user-specified	Unknown	Yes	Yes
System Logs		cron.service	Regular background program processing daemon	Active (running)	Yes	Yes
System Logs Viewer		cryptdisks-early.service	cryptdisks-early.service	Inactive (dead)	No	No
Users and Groups		cryptdisks.service	cryptdisks.service	Inactive (dead)	No	No
Servers 4		cryptsetup-pre.target	Local Encrypted Volumes (Pre)	Inactive (dead)	Always	No
		cryptsetup.target	Local Encrypted Volumes	Active (active)	Always	Yes
Tools 4		dbus	D-Bus is a simple interprocess messaging system, used	Unknown	Yes	Yes
Networking 4		dbus.socket	D-Bus System Message Bus Socket	Active (running)	Always	Yes
Networking 4		dovecot	Init script for dovecot services	Unknown	Yes	Yes
		e2scrub_all.service	Online ext4 Metadata Check for All Filesystems	Inactive (dead)	Always	No
LODAIA		e2scrub_all.timer	Periodic ext4 Online Metadata Check for All Filesystems	Active (waiting)	Yes	Yes
Cluster 🖢 🔹 🖣		e2scrub_reap.service	Remove Stale Online ext4 Metadata Check Snapshots	Inactive (dead)	Yes	No
Un-used Modules		emergency.service	Emergency Shell	Inactive (dead)	Always	No

Click to expand

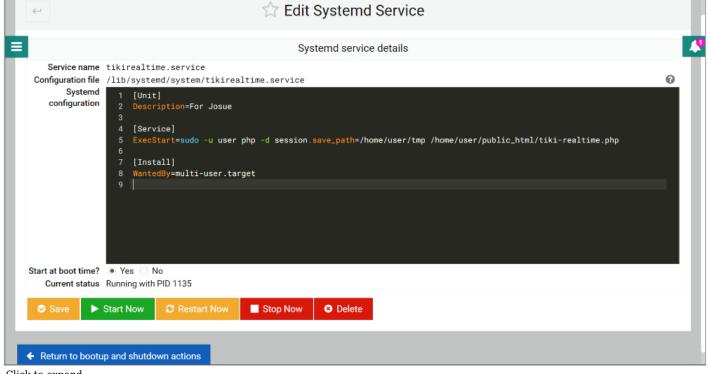
In the opening form, we fill in the **Service name**, the **service description**, and the **commands to run on startup**. As we don't need to shut this down, we are going to leave empty the "Commands to run on shutdown" field, check **Yes** for Start at boot time option, and click **create**.

The command to run on startup is: sudo -u user PHP -d session.save_path=/home/user/tmp
/home/user/public_html/tiki-realtime.php
Replace user with your user

- Create Systemd Service				
		System	d service details	
Service name	tikirealtime	.service		
Service description	Systemd service to start the WS serve	Systemd service to start the WS server		
Commands to run on startup	sudo -u user PHP -d session.save_path=/home/user/tmp /home/user/public_html/tiki-realtime.php			
			©	
Commands to run on shutdown				
			e	
Start at boot time?	Yes No			
Create				
Return to bootup ar	nd shutdown actions			

Click to expand

After this, you can check the content of your system service, and it should look something similar to the image below:



```
Click to expand
```

3. The proxy_wstunnel

The last step here in Virtualmin is to make sure the proxy_wstunnel module is enabled.

To do this, go to **Webmin -> Servers -> Apache Webserver -> Global configuration -> Configure Apache Modules**, and check the **proxy_wstunnel** module checkbox, like in the figure below:

	- country ca	Urbeureu	proxy	Linguieu
	authriz_ldap	Disabled	proxy_ajp	Disabled
Webmin Virtualmin	 authz_core 	Enabled	proxy_balancer	Enabled
	authz_dbd	Disabled	proxy_connect	Enabled
Search Q	authz_dbm	Disabled	proxy_express	Disabled
	authz_groupfile	Disabled	proxy_fcgi	Enabled
🗘 Webmin 🔹 🔹	authz_host	Enabled	proxy_fdpass	Disabled
⊐ System •	authz_owner	Disabled	proxy_ftp	Disabled
a system .	 authz_user 	Enabled	proxy_hcheck	Disabled
Servers -	 autoindex 	Enabled	proxy_html	Disabled
Apache Webserver	Drotii	Disabled	proxy_http	Enabled
	 buffer 	Disabled	proxy_http2	Disabled
AWStats Reporting	 cache 	Disabled	proxy_sogi	Disabled
BIND DNS Server	cache_disk	Disabled	proving a world	Disabled
 Dovecot IMAP/POP3 Server 	cache_socache	Disabled	proxy_wstunnel	Enabled
MySQL Database Server	cem_meta	Disabled	1808000	Disabled
Postfix Mail Server	🗹 cgi	Enabled	reflector	Disabled
Procmail Mail Filter	cgid	Disabled	remoteip	Disabled
	charset_lite	Disabled	reqtimeout	Enabled
ProFTPD Server	data	Disabled	request	Disabled
 Read User Mail 	🗹 dav	Enabled	rewrite	Enabled
SSH Server	🗹 dav_fa	Enabled	sed sed	Disabled
Virtualmin Virtual Servers	dav_lock	Disabled	session	Disabled
	dbd	Disabled	session_cookie	Disabled
X Tools •	deflate	Enabled	session_crypto	Disabled
🗛 Networking 🔹 🔹	dialup	Disabled	session_dbd	Disabled
-	🗹 dir	Enabled	setervif	Enabled
B Hardware 4	dump_io	Disabled	slotmem_plain	Disabled
🛢 Cluster 🔹 🔹	echo	Disabled	Slotmern_shm	Enabled
	env	Enabled	socache_dbm	Disabled
In-used Modules	 expires 	Disabled	socache_memcache	Disabled
C Refresh Modules	ext_filter	Disabled	socache_redis	Disabled
	fogid	Enabled	socache_shmcb	Enabled
Dashboard	file_cache	Disabled	speling	Disabled
	filter	Enabled	🗹 ssl	Enabled
	headers	Disabled	status	Enabled
4] 🌭 >_ 🜟 🍰 root 😐 😂	 heartbeat 	Disabled	substitute	Disabled
	heartmonitor	Disabled	STRANDC	Enabled
	http2	Disabled	unique_ld	Disabled
	Ident	Disabled	userdir	Disabled
	 Imagemap 	Disabled	usertrack	Disabled
E Widbuile menaged hosting	Include	Disabled	vhost_allas	Disabled
	🗌 info	Disabled	xml2enc	Disabled
	Ibmethod_bybusyness	Disabled		
	Ibmethod_byrequests	Disabled		

Click to expand

4. Realtime feature

At this step, we need to enable Realtime feature in Tiki. When not enabled, this feature is visible only upon search. Under **Settings** -> **Control Panels**, in the "Search preferences" search box, type a keyword, "realtime", for example.

🐵 🔻 Preference Filters -	Access ▼ Content ▼ System ▼ Toob ▼	۲
Search preferences Q		
🏫 Admin Dashboard	General Settings •	
🛠 Global Setup	Giobal site configuration, date formats, etc.	
General Settings	😂 Tiki Installer 🚍 Menus 🗇 Server Fitness 🔍 Check for Updates Now 🛛 Ap	ply
() Features		
Wizards Look & Feel	General Preferences General Settings Navigation Date and Time Not	abs
Registration & Log in		
👍 Modules	Server	
🖷 Categories	Temporary directory /home/joshua/public_html/temp	
Q Search	Realtime Tiki	
🚥 i18n 🎓 Profiles	Contact	
Main Features		
Main Features	Contact us O S You can use Trackers to create oustom Contact Us forms.	
	Stats	
Users & Community	Stats	
♣ Advanced	Stats 🗌 💿 🛇	
	Apply	
×		

Click to expand

5. The Realtime port

Let's note that this not required, but it is an interesting preference and usefull in some cases (default port is not available), because it allows you to set the port on which the server will be listening to. 8080 is the port number by default...

Preference Filters •	Access • Content • System • Tools •			۵
Search preferences Q		C		
Admin Dashboard		General Set	-	
🗶 Global Setup		Global site configuration, date fo	ormats, etc.	Apply
General Settings			as 🗸 ourse inness of cheere of through the	
රා Features		General Preferences G	eneral Settings Navigation Date and Time	No Tabs
🖌 Wizards				
Look & Feel		Server		
 Registration & Log in 		Temporary directory	/home/joshua/public_html/temp	
A Modules			3 D advanced	
 Categories Search 		Realtime Tiki	2 9 D advanced	
⊂, search n⊠a i18n		Realtime port	8843	
Profiles			Schwarzen Schwarzen	
			in not specified, port dodd will be used	
🚔 Main Features		Contact		
A Miscellaneous		Contact us		
地 Users & Community		contact us	You can use Trackers to create custom Contact Us forms.	
🍫 Advanced		Stats		
		Stats	000	
			Apply	

Click to expand

Let's note also that Tiki Realtime can also be started from CLI by running, from the project root, the command **php tikirealtime.php**, and, here too, the port can be specified by adding a port parameter to the command: **php tiki-realtime.php -p 8843** which means port number will be set to **8843**.

After this is done, Tiki Realtime should be well set up and working. There is a way to check that. So let's see if everything went well...

Check that Realtime is working

This task is as simple as opening our Tiki and running "Server Check"

To do this you need to go under **Settings** -> **Control Panels** -> **Global Setup** -> **General Settings** and find "**Server Fitness**", click on it, and on the page that opens, scroll down until you find the "**Tiki Realtime**" section. If Tiki Realtime is working, all the requirements status should be "Good" as shown in the image below:

Realtime Tiki		
Requirements	Status	Message
Feature enabled	오 good	Feature is enabled.
Server listening	🕑 good	Server is listening on local system port 8843.
Connectivity	오 good	Connection to WS server established successfully.
Message exchange	🕑 good	Successfully exchanged messages with realtime server.

Click to expand

For developers

- Initial commit: https://gitlab.com/tikiwiki/tiki/-/merge_requests/1638
- If you'd like some guidance to deploy this to various Tiki features, please reach out to Marc Laporte

Related links

• Chatbots