

XOOPS User Pooling - Modules

NEWS_PDF_AUTHOR: wishcraft

NEWS_PDF_DATE: 2008/5/12 3:40:00

Users pooling is an older concept, OpenID is one example where it works well. It is not often done due to the complexity, but now it is written into 2.3.1 as you can see at the demo site <http://www.policybackup.com>.

Just register at <http://www.policybackup.com>, and you'll receive a confirmation email from another server: <http://www.xoops.asia>. Confirm it there, and now you can login with the same username/login at both <http://www.policybackup.com> and <http://www.xoops.asia>.

Following this example, you'll be able to subscribe to XOOPS, and use the same login/password for all XOOPS Websites, e.g. <http://dev.xoops.org>.

With the copy of [X-Soap 3.0.1](#) which comes with the necessary plugins, you can connect your userbase to a central service, have all the other either direct accessing the central service, or series of tiers in a p2p chain sequence. All you need to do is have access to the plugin on that system and you can authorize the users.

Completely secure, SOAP serializes itself between the services sending its data encrypted, so there is no chance of somebody in the middle attacking/hacking it. And it operates on port 80 so it works in just about any environment. This library uses the XoopsUser object primarily, but you could theoretically have your XOOPS system authoring and syndicating services with several hybrid systems like .NET, an IBM Mainframe or other systems that support SOAP.

Users pooling is an older concept, OpenID is one example where it works well. It is not often done due to the complexity, but now it is written into 2.3.1 as you can see at the demo site <http://www.policybackup.com>.

Just register at <http://www.policybackup.com>, and you'll receive a confirmation email from another server: <http://www.xoops.asia>. Confirm it there, and now you can login with the same username/login at both <http://www.policybackup.com> and <http://www.xoops.asia>.

Following this example, you'll be able to subscribe to XOOPS, and use the same login/password for all XOOPS Websites, e.g. <http://dev.xoops.org>.

With the copy of [X-Soap 3.0.1](#) which comes with the necessary plugins, you can connect your userbase to a central service, have all the other either direct accessing the central service, or series of tiers in a p2p chain sequence. All you need to do is have access to the plugin on that system and you can authorize the users.

Completely secure, SOAP serializes itself between the services sending its data encrypted, so there is no chance of somebody in the middle attacking/hacking it. And it operates on port 80 so it works in just about any environment. This library uses the XoopsUser object primarily, but you could theoretically have your XOOPS system authoring and syndicating services with several hybrid systems like .NET, an IBM Mainframe or other systems that support SOAP.