

# Configuring Balabit's Privileged Session Management for Blindspotter

June 19, 2018

## Abstract

Configure Balabit's Privileged Session Management (PSM, formerly called SCB) to work with Privileged Account Analytics (PAA, formerly called Blindspotter (BSP))



# Table of Contents

1. Configuring Balabit's Privileged Session Management for Blindspotter .....	3
2. The PSM RPC API .....	4
2.1. Requirements for using the RPC API .....	4
2.2. RPC client requirements .....	4
2.3. Locking PSM configuration from the RPC API .....	5
2.4. Documentation of the RPC API .....	6
2.5. Enabling RPC API access to PSM .....	6
3. Usernames in RDP connections .....	7

## 1. Configuring Balabit's Privileged Session Management for Blindspotter

To be able to add Balabit's Privileged Session Management as a data source to Blindspotter, you will have to configure Balabit's Privileged Session Management in a way so that it contains data that can be used by Blindspotter later.

Shell Control Box has the following requirements:

Type	Requirement
SCB version	Any supported version from version 4.0.4 onward, ideally the latest one
API	Enabled RPC API access on SCB
Access rights	A user account with search access rights
Data	Data that contains real, unique usernames linked to users other than root/administrator or a shared account.  Ensure that there is least 2-3 months worth of data available in SCB, with at least 50 sessions per user.

Table 1. Shell Control Box prerequisites

To enable Blindspotter to fetch data from PSM, you will need PSM's IP address and the credentials:

- PSM's IP address
- PSM username
- PSM password

To be able to fetch activities, make sure that the connections include user names. To check this, navigate to **Search > Search** and check whether the **Username** column contains data.

This is important, because users will be fetched directly from SCB, and the activities performed will be linked to these users.

In addition, enable indexing for your SSH connections. This is required if you want Blindspotter to analyze command data at some point. For detailed instructions on configuring full-text indexing of audit trails, see *Chapter 15, Indexing audit trails* in *The Balabit's Privileged Session Management 5 LTS Administrator Guide*.

---

All questions, comments or inquiries should be directed to <info@balabit.com> or by post to the following address: One Identity LLC 1117 Budapest, Alíz Str. 2 Phone: +36 1 398 6700 Fax: +36 1 208 0875 Web: <https://www.balabit.com/>  
 Copyright © 2018 One Identity LLC All rights reserved. This document is protected by copyright and is distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this document may be reproduced in any form by any means without prior written authorization of One Identity.

All trademarks and product names mentioned herein are the trademarks of their respective owners.

## 2. The PSM RPC API

Version 3 F3 and later of Balabit's Privileged Session Management can be accessed using a Remote-Procedure Call Application Programming Interface (RPC API).

The PSM RPC API allows you to access, query, and manage PSM from remote applications. You can access the API using the Simple Object Access Protocol (SOAP) protocol over HTTPS, meaning that you can use any programming language that has access to a SOAP client to integrate PSM to your environment. You can download simple, proof-of-concept clients for Python and other languages from the PSM web interface.

Accessing PSM with the RPC API offers several advantages:

- Integration into custom applications and environments
- Flexible, dynamic search queries and management

### 2.1. Requirements for using the RPC API

To access PSM using the RPC API, the following requirements must be met:

- Accessing the appliance via the RPC API must be enabled on the web interface. For details, see *Procedure 2.5, Enabling RPC API access to PSM (p. 6)*.
- The appliance can be accessed using the SOAP protocol over authenticated HTTPS connections. The WSDL describing the available services is available at <https://<ip-address-of-PSM>/rpc.php/<techversion>?wsdl>. For details on the client libraries tested with PSM see *Section 2.2, RPC client requirements (p. 4)*.
- The user account used to access PSM via RPC must have **read and write/perform** rights for the **Access RPC API** privilege. This is required for every type of RPC access, even for read-only operations. Members of the *api* group automatically have this privilege. For details on managing user privileges, see *Procedure 5.7.2, Modifying group privileges in The Balabit's Privileged Session Management 5 LTS Administrator Guide*.



#### Warning

Each PSM release provides a separate API with a new API version number. You are recommended to use the PSM version 5 LTS with the corresponding API version. Earlier versions are not supported

### 2.2. RPC client requirements

The client application used to access PSM must meet the following criteria:

- Support SOAP version 1.1 or later.
- Support WSDL version 1.1.
- Properly handle complex object types.
- Include a JSON decoder for interpreting the results of search operations.

The following client libraries have been tested with PSM.

Client name	Programming language	Status	Comments
Apache Axis 1	Java	Working	
Built-in .NET library	.NET	Working	PSM does not support the Expect HTTP Header feature, and must be disabled, for example, using <code>\$client-&gt;ServiceProxyMap.ExpectCookie = false;</code>
Scio	Python	Partially working	Does not handle complex object types, so it cannot perform search queries.
SOAP::Lite	Perl	Working	<ul style="list-style-type: none"> <li>Simple types can be used with the following format: <code>\$service-&gt;\$meth(@params)</code></li> <li>Complex types work only with the following format: <code>\$service-&gt;call(\$meth, @params)</code></li> <li>Calls using the <code>\$service-&gt;call()</code> format seem to work after doing at least one <code>\$service-&gt;\$meth(@params)</code> call, for example, a login.</li> </ul>
SOAP::WSDL	Perl	Not working	
Suds	Python	Working	

Table 2. SOAP libraries tested with PSM

### 2.3. Locking PSM configuration from the RPC API

Accessing PSM using the RPC API locks certain components of PSM from other users, just like accessing PSM using the web interface or the console. Locking PSM via RPC can be performed either explicitly by calling the `lockAcquire` function, or implicitly when an operation requires the lock. In either case, ensure that your

application verifies that the lock is received and properly handles if the component is locked by someone else (for example, because a user is accessing the component from the web interface).

For details on how locking works in PSM, see *Section 4.2.2, Multiple users and locking* in *The Balabit's Privileged Session Management 5 LTS Administrator Guide*.

## 2.4. Documentation of the RPC API

The documentation of the PSM RPC API is available online from the PSM web interface: select **Basic Settings** > **Management** > **RPC API settings** > **Open documentation**, or directly from the following URL: <https://<ip-address-of-PSM>/rpc-api-doc/>. This documentation contains the detailed description of the available services and classes.

## 2.5. Procedure – Enabling RPC API access to PSM

### Purpose:

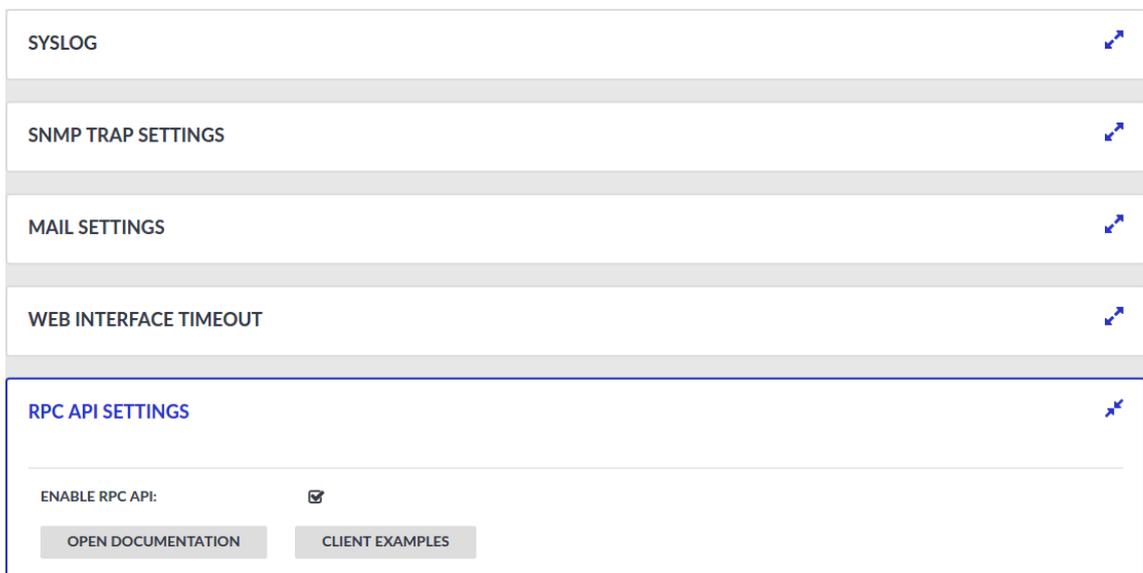
To configure PSM to accept RPC API connections, complete the following steps.

### Steps:

Step 1. Login to the PSM web interface.

Step 2. Select **Basic Settings** > **Management** > **RPC API settings** > **Enable RPC API**.

Figure 1. Basic Settings > Management > RPC API settings — Enabling RPC API access to PSM



Step 3.

Click .

### Expected result:

Users accounts belonging to a usergroup that have *read* and *write/perform* rights to the *Access RPC API* privilege can access PSM via the RPC API.

### 3. Usernames in RDP connections

When processing RDP connections, PSM attempts to extract the username from the connection. For example, you need the username to:

- Use gateway authentication for the connection. For details on gateway authentication, see [Section 17.2, Configuring gateway authentication](#) in *The Balabit's Privileged Session Management 5 LTS Administrator Guide*.
- Use usermapping policies. In this case, PSM compares the username on the server with the username on the gateway. For details on usermapping policies and gateway authentication, see [Procedure 17.1, Configuring usermapping policies](#) in *The Balabit's Privileged Session Management 5 LTS Administrator Guide* and [Section 17.2, Configuring gateway authentication](#) in *The Balabit's Privileged Session Management 5 LTS Administrator Guide*, respectively.



**Note**

In certain cases, PSM receives an empty username from the server, and the connection will be denied by the usermapping policy unless a policy is set for the connection that allows every user for the given group. To add such a policy, specify \* in the **Username on the server** field of the usermapping policy. For a list of cases when PSM receives empty username, see [Section Windows settings that interfere with username extraction](#) (p. 7).

- Search or filter connections by the username on the PSM search interface, or create automatic statistics based on the username.
- Find the connection of the user on the **Four Eyes** and **Active Connections** pages.
- Usernames are also essential if you want to use Privileged Account Analytics. If you are interested in Privileged Account Analytics, [contact our Sales Team](#), or your Balabit representative.

PSM can record the username automatically if the RDP connection is using Network Level Authentication (CredSSP), and usually in other scenarios as well. If PSM cannot automatically extract the username, it displays the following login screen to the user (note that PSM can display this login screen only in [TLS-encrypted connections](#)).

The known scenarios that interfere with RDP usernames are listed in [Section Windows settings that interfere with username extraction](#) (p. 7).

To ensure that your users can access the target servers only when their username is recorded, you can configure PSM to terminate RDP connections if it cannot reliably extract the username. To terminate such connections, clear the **RDP Control > Settings > Permit unreliable usernames** option.

#### Windows settings that interfere with username extraction

The following settings on the Windows client or server can prevent PSM from correctly extracting the username from the RDP connection. As a result, the username is not visible on the **Search**, **Four Eyes** and **Active Connections** pages.

- The **DontDisplayLastUserName** option is enabled on the server. The **DontDisplayLastUserName** security setting of Windows servers specifies whether the username from the last successful login

is displayed on the login screen as a default for the next login. To disable the **DontDisplayLastUserName** security setting, do one of the following.

- D i s a b l e t h e HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System\DontDisplayLastUsername registry setting. For more details, see the [\*DontDisplayLastUserName TechNet article\*](#).



**Note**

Registry settings can be overridden by Group Policy settings.

- Disable this option in the Computer Configuration\Windows Settings\Security Settings\Local Policies\Security Options policy. For details, see [\*Do not display last user name in logon screen TechNet article\*](#).
- There is no server-side authentication. To avoid this problem, ensure that your server requires authentication from the users.
- If the server is Windows 2003 Server or Windows XP and the **Allow to save credentials** or **Remember my credentials** options are enabled in the Remote Desktop client application. In this case, disable these options on the client, and delete any credentials that have already been saved on the client.