

Phone Password Manager

Configuration Guide

Software revision: 12.2.4
Document revision: 30072
Last changed: 2022-03-01

Contents

I	INTRODUCTION	2
1	About this document	3
1.1	This document	3
1.2	Conventions	3
1.3	Feedback and help	4
2	Overview of <i>Phone Password Manager</i>	5
2.1	Architecture	5
2.2	Example process	6
II	PRE-INSTALLATION SETUP	8
3	Task Checklist	9
4	Configuring the IVR Server	12
4.1	Configuring Dialogic® voice boards	12
4.1.1	IVR server requirements	12
4.1.2	Calculating the number and size of voice boards	13
4.1.3	Selecting a voice board	15
4.1.4	Assigning phone lines	15
4.1.5	Installing the hardware and IVR software	16
4.2	Configuring Dialogic® PowerMedia Host Media Processing Software	17
4.2.1	<i>Phone Password Manager</i> Server requirements	17
4.2.2	Installing the Host Media Processing software	17
4.3	Configuring Asterisk® Software	18
4.3.1	AsteriskNow	18

4.3.2	Asterisk package on a Linux environment	18
4.3.3	Asterisk package on a Windows environment	19
4.3.4	Configuring the Asterisk® server	19
4.4	Configuring a software phone	21
4.4.1	Placing a call with softphone	21
5	Setting up the <i>Bravura Pass</i> Remote API	22
5.1	Enabling the API Service	23
5.2	Configuring an IDAPI caller	23
III	INSTALLATION AND INTEGRATION	25
6	Installing the <i>Phone Password Manager</i> Software	26
6.1	Before you begin	26
6.2	Using setup to manage installations	27
6.3	Using setup to install the software	28
6.4	Managing alternate instances	30
6.5	Uninstalling the software	30
7	Integrating <i>Bravura Pass</i> and <i>Phone Password Manager</i>	31
7.1	Integration Mechanisms	31
7.1.1	Web service	31
7.1.2	Example function call sequence	32
7.1.3	Event actions	32
7.2	Editing the configuration file	33
7.2.1	Editing the Asterisk® configuration file	34
7.2.2	Using 3CX PBX systems	34
7.2.3	Defining disconnect tones in idtel.cfg	34
7.2.4	Defining disconnect tones with a TSF file	35
7.3	Writing call logic scripts	36
7.4	Managing audio files	37
7.4.1	Adding custom authentication questions	37
7.4.2	Defining custom target systems	38

7.4.3	Defining custom target system groups	38
7.4.4	Supporting custom call logic	39
7.5	Adding additional languages	39
7.5.1	HDD Encryption Audio files	40
7.6	Mapping user IDs to telephone keypads	40
7.7	Using multiple <i>Bravura Pass</i> instances with <i>Phone Password Manager</i>	41
IV	CONFIGURATION	42
8	User Authentication	43
8.1	Setting up IVR question sets	43
8.1.1	Using the default question set	44
8.1.2	Adding question sets	44
8.1.3	Question set options	44
8.1.4	Adding questions	46
8.1.5	Recording question vocals	47
9	Voice Print Enrollment	48
9.1	Setting up voice print enrollment	48
9.2	Configuring voice print options	49
9.3	Testing voice print enrollment	50
9.4	Testing voice print enrollment (command line)	50
9.5	Removing voice print enrollment data	51
10	Speech Recognition and Text-to-Speech	53
10.1	Configuring the speech service	54
10.2	Building .wav files using SAPI	55
11	Monitoring Line Status	57
12	Call Modes	58
12.0.1	Auto-answer mode	58
12.0.2	Inbound mode	58
12.0.3	Outbound mode	59

13	Call Transfer	60
13.1	Pre-configuration	60
13.2	Configuring TransferCall for SIP protocol	61
13.3	Configuring TransferCall for H323 protocol	62
14	Bridge Transfer	63
14.1	Pre-configuration	64
14.2	Configuring bridge transfers	64
14.3	Recording bridge transfers	65
14.4	Configuration notes	65
15	HTTPS Encryption	66
16	Viewing logs	67
16.1	Viewing the unlock logs	67
16.2	Viewing the reset logs	68
17	Troubleshooting	69
17.1	Problems with the voice board	69
17.2	Problems with hangup events	70
17.3	Welcome message does not play	70
17.4	Some or all audio files are not playing	70
17.4.1	Asterisk® audio files	71
17.5	The <i>Phone Password Manager</i> service fails to start	71
17.6	<i>Phone Password Manager</i> cannot return requests	72
17.7	<i>Phone Password Manager</i> cannot connect to the softphone system	72
17.8	<i>Phone Password Manager</i> fails unexpectedly	72
17.9	SoX version mismatched	73
	Index	74

Hitachi ID Systems, Inc.

DISCLAIMER

Although every effort has been made to ensure that the information in this manual is accurate, some information may be inconsistent with the most current software release.

For assistance with installation and configuration, please contact support@Hitachi-ID.com.

Part I

INTRODUCTION

About this document

1

1.1 This document

This document, the *Phone Password Manager* Configuration Guide, contains important information about how to install *Phone Password Manager* and integrate it with an existing *Hitachi ID Bravura Pass* installation.

This guide is to be used in conjunction with the entire set of *Bravura Pass* documents; all of the other guides are complimentary to it. Because there are many components to *Phone Password Manager* - *Bravura Pass* integration, including third party hardware and software, installation and configuration tasks can be challenging.

It is *strongly* recommended that you review this entire document before proceeding with an installation.

1.2 Conventions

This document uses the following conventions:

This information . . .	displayed in . . .
Variable text (substituted for your own text)	<angle brackets>
Non-text keystrokes – for example, [Enter] key on a keyboard.	[brackets]
Terms unique to <i>Hitachi ID Bravura Security Fabric</i>	<i>italics</i>
Button names, text fields, and menu items	boldface
Web pages (names)	<i>italics and boldface</i>
Literal text, as typed into configuration files, batch files, command prompts, and data entry fields	monospace font
Wrapped lines of literal text (indicated by the → character)	Write this string as a →single line of text.
Hypertext links – click the link to jump to a section in this document or a web site	Purple text
External document – click the link to jump to a section in another document. The links only work if the documents are kept in the relative directory path.	Magenta text

1.3 Feedback and help

If you have feedback about this document or wish to report an omission or error, please contact doc-feedback@Hitachi-ID.com.

If you require technical assistance with *Hitachi ID Bravura Pass*, contact support@Hitachi-ID.com.

Overview of *Phone Password Manager*

2

Phone Password Manager™ is a turn-key telephony-enabled password reset solution. It allows users to:

- Reset their own forgotten or expired passwords on one or more systems
- Clear intruder lockouts on one or more of their own accounts
- Manage their own RSA SecurID tokens

Phone Password Manager is useful for mobile and remote users who cannot readily access a web user interface for self service or who may be accustomed to getting service from the IT organization via telephone.

2.1 Architecture

At minimum, a typical *Phone Password Manager* deployment consists of the following servers:

- A *Hitachi ID Bravura Pass* server running the API SOAP Service (idapisoap)
This service allows applications to access *Bravura Pass* functionality using the PASSWORD MANAGER REMOTE API.
- A separate Interactive Voice Response (IVR) server with *Phone Password Manager* installed, and running either the *Phone Password Manager module service*, and/or the *VoIP Telephony Service*
The *Phone Password Manager* service is an *IVR application* that interacts with either hardware *voice boards*, the Asterisk® framework, or *host media processing software* installed on the system to receive incoming calls and provide responses to the user. It uses the `pspushpass.dll` library to communicate with the API SOAP Service (idapisoap) on the *Bravura Pass* server.

Note: It is recommended that your *Bravura Pass* and *Phone Password Manager* servers are hosted on separate hardware, as this configuration has several advantages over a single server setup. Installing both of these products on the same hardware means that both servers need to be shut down in the event maintenance is required.

Figure 2.1 shows a typical *Phone Password Manager* installation. The physical layout of the *Phone Password Manager* solution varies depending on the needs of your organization.

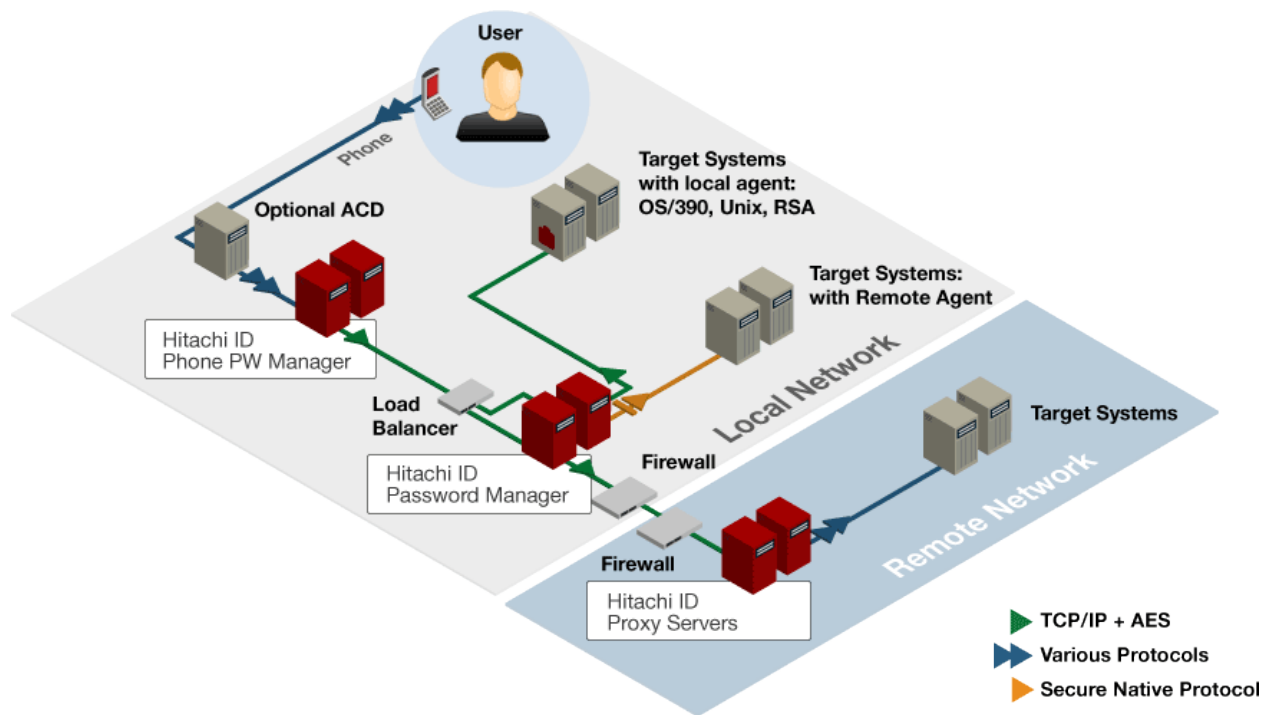


Figure 2.1: Architecture

2.2 Example process

The following example illustrates how end-users can use their telephones to authenticate to *Phone Password Manager* using *Hitachi ID Bravura Pass* challenge-response, and reset their password to a new, random value:

1. **User:** forgets password or triggers intruder lockout.
2. **User:** dials the support number, and is connected to the *Phone Password Manager* server.
3. **Phone Password Manager server:** prompts the user to key in numeric ID.
4. **User:** keys in the ID.
5. **Phone Password Manager server:** sends a request to the *Bravura Pass* server.
6. **Bravura Pass server:** looks up the user's profile.
7. **Bravura Pass server:** returns a random subset of the user's fixed-length numeric pre-defined questions to *Phone Password Manager*.
8. **Phone Password Manager server:** prompts the user to answer the selected questions.
9. **User:** keys in (numeric) answers to the selected questions, or answers questions verbally, depending on configuration.
10. **Phone Password Manager server:** forwards answers to the *Bravura Pass* server.

11. **Bravura Pass server:** evaluates the validity of the user's credentials, and either allows them access, repeats the login process, or potentially triggers a lockout.
12. **Phone Password Manager server:** prompts the user to select reset password or unlock account.
13. **User:** navigates the audio menu, and requests a password reset.
14. **Phone Password Manager server:** prompts the user to select on target systems.
15. **User:** selects an individual target system or all target systems.
16. **Phone Password Manager server:** invokes secure API/RPC to request a random password for this user, on the target or target systems the user selected.
17. **Bravura Pass server:** generates a random, policy-compliant password value.
18. **Bravura Pass server:** implements the password reset on the target system or target systems the user selected.
19. **Phone Password Manager server:** enunciates the password, and prompts the user if to reset another password.
20. **Bravura Pass server:** if configured, writes a ticket to a call tracking system.
21. **Bravura Pass server:** if configured, sends the user a confirmation email.

Part II

PRE-INSTALLATION SETUP

Task Checklist

3

You can set up *Phone Password Manager* using either voice boards or VoIP software depending on the structure of your phone system.

Depending on your chosen setup, you must configure either:

- The Dialogic® card, Dialogic® driver, *Bravura Pass* software, and *Phone Password Manager* software to all work together.

Or

- The *Phone Password Manager* software, *Hitachi ID Bravura Pass* software, and Dialogic® PowerMedia Host Media Processing software to all work together.

Or

- The *Phone Password Manager* software, *Bravura Pass* software, and an Asterisk® server to work together.

This is a simple process, but it can become complicated if steps are skipped or performed incorrectly.

WARNING!: The success of your *Phone Password Manager* project depends largely on the proper set up and configuration of your PBX system. Hitachi ID Systems *strongly* recommends that you involve an in house expert, or hire an outside consultant, to set up your system.

Use this checklist as a guide to ensure that you complete each step:

1. Configure the interactive voice response (IVR) backend:

- Using voice boards:
 - (a) Review the IVR server requirements and designate a machine (p12).
 - (b) Calculate the number and size of voice boards that you will require (p13).
 - (c) Select and purchase the voice board that best suits your needs (p15).
It is important to select a board that has voice capabilities and is compatible with your existing telephony infrastructure.
 - (d) Have a phone technician assign additional phone lines if required (p15).
It is important that the phone lines match the type of voice board that you will be using (analog vs. digital).
 - (e) Install the voice boards, drivers, and System Release software (p16).

- (f) Test your system to ensure that it is functioning correctly and that the boards are responding to external calls.
 - Using VoIP Software:
 - (a) Review the IVR server requirements and designate a machine (p17).
 - (b) Install the Dialogic® PowerMedia Host Media Processing (HMP) Software (p17).
 - Using Asterisk® :
 - (a) Determine which Asterisk® distribution is right for your organization, and install (p18).
 - (b) Integrate Asterisk® and *Phone Password Manager* (p19).
 - (c) Configure any VoIP softphones you may require (p21).
 - If you will be deploying biometric voice print verification on the *Phone Password Manager* server, install the VoiceVantage software (p48).
2. Configure the *Bravura Pass* server.
- (a) Use setup to install *Bravura Pass*.

During installation and configuration, ensure that you keep a record of the following information in a safe location, as it will be required when installing and configuring *Phone Password Manager*:

 - Host name or IP address of the *Bravura Pass* server.
 - Name and password of the API caller.
 - Web address and Port that the API SOAP Service (idapisoap) is listening on.
 - Communication encryption key.
 - (b) Configure the API caller (p23).
 - (c) *Optional: Configure custom authentication questions* (p43).
 - (d) If you have not already done so, configure target systems, and run auto discovery to list users.
 - (e) Ensure that the password policy on the *Bravura Pass* server matches the password policies of any target system where *Phone Password Manager* users have accounts. If they do not match, then password changes will eventually fail on the target systems.
3. Install *Phone Password Manager*, and configure *Bravura Pass* integration:
- (a) Install *Phone Password Manager* (p26).
 - (b) *Optional: Edit the configuration files* (p33).

Do this if your board configuration differs from what the *Phone Password Manager* installer auto-detected, or if you want to modify the default settings.
 - (c) Determine how user IDs will map to telephone keypads (p40).
 - (d) Supply additional vocals (p37) as required.
 - (e) *Optional: Configure Phone Password Manager to support multiple languages* (p39). Do this if you intend the *Phone Password Manager* system to offer support for languages other than English.
 - (f) *Optional: Write new scripts, or modify the existing call logic script* (p36).

The default script allows users to log in, authenticate using challenge-response questions stored in *Bravura Pass*, and perform the following operations: reset passwords, unlock accounts, or manage tokens (If configured).
 - (g) If required, modify \<Instance>\service\pspushpass.cfg to configure:
 - *Bravura Pass* server address.

- API Service (idapi) address, port, and authentication settings.
- SSL Settings. See [HTTPS Encryption](#) for further SSL configuration details.

4. Configure *Phone Password Manager* security and call handling:

- If you want *Phone Password Manager* to authenticate users by prompting them with questions from their *Bravura Pass* security question profiles, [configure *Phone Password Manager* question sets](#) (p43).

Question sets and questions must be enabled specifically for *Phone Password Manager* integration. To do this, navigate to the **Question set information** page and enable the “Ask telephone users to answer questions from this set” option.

- To set up voiceprint authentication, [configure the *Generate voice print enrollment PIN* \(PSI\) module](#) (p48).
- If you want *Phone Password Manager* to be able to recognize spoken words and playback text as audible speech, [configure speech recognition and text-to-speech](#) (p53).
- If you want *Phone Password Manager* to selectively answer calls, or to make outbound calls, [configure call modes](#) (p58).
- If you are using VoIP, and you want to be able to transfer IVR calls, [configure the TransferCall function](#) (p60).
- If you want *Phone Password Manager* to use HTTPS encryption, [configure the *Bravura Pass* server](#) (p66).

5. Test your system to ensure that users can log in to the IVR system and perform the supported operations using a telephone.

Configuring the IVR Server

4

This chapter describes how to set up your interactive voice response (IVR) server. Do not install any software or hardware until you have read and understood all of the information in this chapter.

The configuration requirements differ depending on your phone system:

- Dialogic® voice boards (p12)
- Dialogic® PowerMedia Host Media Processing Software (p17)
- Asterisk® Software (p18)

4.1 Configuring Dialogic® voice boards

4.1.1 IVR server requirements

In order to work with *Phone Password Manager*, your interactive voice response (IVR) server must be configured as follows:

Operating system

Phone Password Manager requires Microsoft Windows Server 2012 at current service packs.

System Release software

You must install both the hardware and the Dialogic® System Release 6.0+ PCI for Windows, including the latest service update. This System Release software is available from your vendor, though an evaluation version is also available.

For more information about the System Release contact your vendor or visit:

<http://www.dialogic.com/en/products/media/system-release-software/>

Additional software

If you will be deploying biometric voice print verification on the IVR server, you must also purchase and install VoiceVantage software.

For more information about VoiceVantage contact sales@Hitachi-ID.com.

4.1.2 Calculating the number and size of voice boards

The number of boards to install and the size of each board depends on the following factors:

- On average, how long does each call take to complete?
- How many calls must be processed at one time?
- How many users are in your organization?
- What features will be provided?
- What is the expiry interval for passwords?
- How many calls per day does the help desk currently receive for forgotten passwords?
- What percentage of those calls are expected to be offloaded to *Phone Password Manager*?

We can calculate the number of phone lines needed by taking the average call duration, which we determined to be two and a half minutes, and determining how many calls a single line can handle in an hour:

$$\text{calls per line per hour (CALLSPERLINE)} = 3600 \text{ sec/hr} \times 1 \text{ call}/150 \text{ sec} = 24 \text{ calls/hr}$$

We can then calculate the minimum number of required lines:

$$\text{number of lines needed (NLINES)} = \text{number of calls received per hour} / \text{CALLSPERLINE}$$

This formula is illustrated in the following example.

Example

Global Enterprises has 80,000 users. Users are required to change their passwords every 30 days. We can estimate the number of calls received per hour based on the following assumptions:

- If 10,000 users a month forget their passwords, and if the *Phone Password Manager* service handles all password change requests, a total of 90,000 calls will go through the *Phone Password Manager* service every month.
- If there are 20 work days in the average month, there will be 4,500 password change requests every day.
- If password change requests are distributed evenly throughout the 24-hour period, the *Phone Password Manager* service will receive approximately 188 password change requests every hour.

Using this example:

$$\text{NLINES} = 188 / 24 = 7.8 \text{ (round up to 8)}$$

An 8-port board (or two 4-port boards) is the absolute minimum required to support 188 calls / hr. Keep in mind that this is the *minimum* requirement. To allow for growth within your organization, we recommend in this case that a 12-port board be used.

To make the example more applicable to the real world, the following facts must be taken into consideration:

- Calls will not be evenly distributed throughout the day.
- Users will also be performing other tasks (like unlocking accounts) at the same time.
- Your organization will grow over time.

Your calculations can be made more realistic by changing the numbers in the formula to account for peak call times and adding in other requests that might have to be processed. For example, you might expect 50% of the calls in a single day to occur between 8:00 am and 10:00 am. That means you could receive up to 1,125 calls per hour. You might also receive 200 account unlock requests within an hour, thus bringing the number of potential calls in an hour to 1,325. In this case, you need to decide whether to buy the necessary hardware to support that volume of calls, or to allow users to be put on hold for a short time until a line becomes available.

4.1.3 Selecting a voice board

The *Phone Password Manager* service is able to interact with a number of Dialogic® boards. When selecting a voice board, you must consider the following factors:

- The amount of load expected on the IVR server.

See [Calculating the number and size of voice boards](#) for more help with this.

- Existing phone lines.

If you have a sufficient number of existing lines, you *must* purchase a board that matches your type of line (digital or analog) to avoid having to install new lines.

WARNING!: Using a digital line with an analog board, or an analog line with a digital board can permanently damage your telephony hardware.

- The configuration of the *PBX* (private branch exchange) at your organization.

A PBX integration board provides all of the basic voice and call processing capabilities of standard voice boards, and adds hardware and firmware that eases integration with supported PBXs.

The PBX integration board connects directly to your PBX using a digital line. If you require this type of board, make sure to select one that is compatible with your PBX.

The PBX integration board must be configured to assign a phone number to the Dialogic® card.

Consult your PBX documentation for details.

Note: Hitachi ID Systems highly recommends that you install Dialogic® PowerMedia Host Media Processing (HMP) server on a physical machine rather than a virtual machine to avoid performance and inconsistency issues.

For more information about voice boards, see:

<https://www.sangoma.com/telephony-cards/dialogic-boards/jct/jct-combined/>

For more information about PBX integration boards, see:

<https://www.sangoma.com/telephony-cards/dialogic-boards/jct/pbx-integration-boards/>

Note: *Phone Password Manager* does not support Dialogic® Diva® products.

4.1.4 Assigning phone lines

The type and number of line that you must install depends on your voice board. Check your vendor documentation to ensure that you are installing the correct type of phone line.

A phone technician should perform the following tasks:

1. Ensure that the necessary number of phone lines are installed and tested for functionality.
2. If required, configure the telephone lines as a sequential *hunt group*.
3. If required, assign the lead line on the sequential hunt group a dedicated telephone number.

WARNING!: Plugging an analog line into a digital board and vice versa can damage the board.

4.1.5 Installing the hardware and IVR software

Once the physical lines are available, you can install the voice boards, drivers, and System Release software. Since some boards require the software to be installed first and other boards need to be installed before the software, it is recommended that you refer to the documentation included with the System Release before you install the voice board.

After you have completed the installation, test your system to ensure that it is functioning correctly and that the board is responding to external calls. Typically, the Dialogic® card drivers include software for testing that the card is picking up calls.

Note: If you require drivers or troubleshooting information, contact your vendor.

If you will be deploying biometric voice print verification on the IVR server, you can install the VoiceVantage software *after* you have verified that your system is functioning correctly.

Next:

Set up the [Bravura Pass Remote API](#) (p22).

4.2 Configuring Dialogic® PowerMedia Host Media Processing Software

4.2.1 Phone Password Manager Server requirements

In order to use VoIP or softphone technology with *Phone Password Manager*, your interactive voice response (IVR) server must be configured as follows:

Operating system

Phone Password Manager requires Microsoft Windows Server 2012 at current service packs.

Your IVR system should be installed on a designated server. This machine should be separate from the *Hitachi ID Bravura Pass* server.

4.2.2 Installing the Host Media Processing software

If you want to configure *Phone Password Manager* to use VoIP or softphone systems instead of voice boards, then you must install Dialogic® PowerMedia Host Media Processing (HMP) 3.0+ for Windows.

The trial version of HMP and a demo license can be downloaded from the Dialogic® web site.

For more information about Dialogic® PowerMedia Host Media Processing, contact your vendor or visit:
<http://www.dialogic.com/en/products/media-server-software/hmp-software.aspx>

CAUTION: Dialogic® PowerMedia Host Media Processing is incompatible with several virtualization programs, and it is highly recommended that you install it on a physical machine.

Once the Dialogic® PowerMedia Host Media Processing software is installed, the *Phone Password Manager* will automatically perform any required configurations upon installation.

Next:

Set up the *Bravura Pass Remote API* (p22).

4.3 Configuring Asterisk® Software

Asterisk® is an open-source telephony solution which provides a wide variety of call management functionality. As a primarily Unix-based product, Asterisk® is best installed on a server separate from your *Phone Password Manager* server.

When using Asterisk® software, *Phone Password Manager* can be installed on a hardware machine or a virtual machine, and does *not* require a Dialogic® card or HMP software.

The Asterisk® module is included by default, and installs support for receiving calls from an Asterisk® server.

Phone Password Manager supports:

- [AsteriskNow](#) (p18) complete Linux distribution
- [Asterisk package](#) (p18) on a Linux environment
- [Asterisk package](#) (p19) on a Windows environment

4.3.1 AsteriskNow

AsteriskNow is a complete Linux distribution with Asterisk, the DAHDI driver framework, and optionally, the FreePBX administrative GUI.

Download the AsteriskNow ISO from www.asterisk.org/downloads/asterisknow and install the operating system on either a new virtual machine or a hardware box.

Next:

Configure the Asterisk® server (p19).

4.3.2 Asterisk package on a Linux environment

You can install the Asterisk® server in a Linux environment either on a hardware machine or a virtual machine.

To install, use the system's package manager to select the following packages, and apply the change:

- asterisk
- asterisk-core-sounds-en-gsm
- asterisk-config
- sox

Note: The sox package is required by *Phone Password Manager* in order to convert audio files.

Note: When using a source package to install Asterisk®, ensure that the *make* commands are run after all the libraries listed here have been installed and configured.

Additional information on installing Asterisk® can be found at: <http://blogs.digium.com/2012/11/05/how-to-install-asterisk-11-on-centos-6/>

You may also need to allow access to the Asterisk audio file path, so that the "asterisk" user can create a new directory there; for example:

```
sudo chown asterisk:asterisk /usr/share/asterisk
```

The destination might vary between Asterisk versions and Linux environments; it can be found in the *Phone Password Manager* log, where it shows the creation of the directory failed due to the permission.

Next:

Configure the Asterisk® server (p19).

4.3.3 Asterisk package on a Windows environment

It is also possible to install the Asterisk® server in a Windows environment. Please note that Asterisk® for Windows is a 32-bit binary which is only supported on Windows 2000, XP, or 2003. You will need to install Cygwin on this server, in order to run the SoX binary required by *Phone Password Manager*.

Download the executable from www.asteriskwin32.com and install Asterisk in the default location.

Note: The site requires the Adobe® Flash plugin.

Install the sox package from Cygwin. Cygwin is a Linux-like environment for Windows. The SoX package will allow *Phone Password Manager* to convert the ".wav" audio files to ".ulaw" audio files, which can be played back to end users. You can download and install Cygwin tools from:

<http://www.cygwin.com/setup.exe>

Using the Cygwin setup binary, search for the sox package and install it on the Asterisk installed directory (default c:\cygroot), or ensure the installation directory is in the system path.

Next:

Configure the Asterisk® server (p19).

4.3.4 Configuring the Asterisk® server

Once installation is complete, configure the Asterisk® server:

1. Start the Asterisk service by running the following command:


```
sudo asterisk -vc
```

or, on Windows:

```
asterisk -vc
```

2. If you are using a Linux system, connect to the Asterisk console:

```
sudo asterisk -r
```

3. Navigate to the `/etc/asterisk` directory on Linux, or `C:\cygroot\asterisk\etc` on a Windows system.

4. Open `sip.conf` for editing.

5. Modify the configuration file by adding one or more of the following tags to the end of the file:

```
[777]
type=friend
username=777
secret=777
host=dynamic
disallow=all
allow=ulaw
```

Write your changes and close the file.

6. Run the `sip reload` command from the Asterisk console to reload the new SIP configuration.

7. Open `extensions.conf` for editing.

8. Add one or more of the following tags at the end of the configuration file:

```
[HiTPM]
exten => 777,1,Ringing
exten => 777,n,AGI(agi://10.0.42.103)
exten => 777,n,Hangup
```

Where:

777 is the newly-added extension

Ringing will ring first when calling this extension

10.0.42.103 is the IP address where the *Phone Password Manager* instance is installed

Hangup will hang up after connection to the instance is finished

9. To include the newly-added tag (HiTPM), add `include => HiTPM` to the `[default]` tag section below `include => demo`. You may also need to add this tag to the `[From-Local]` section as well.

10. Write your changes and close the file.

11. If the Asterisk server has not already started, run `asterisk start`.

12. Run the `dialplan reload` command from the Asterisk console to load the new configuration.

Next:

Configuring a software phone (p21).

4.4 Configuring a software phone

Installing a software phone (softphone) will allow you to place calls to the IVR software, and can be used to test that your configuration is running properly. Once the interactive voice response (IVR) server is configured, create a softphone account and activate it.

4.4.1 Placing a call with softphone

Depending on your IVR configuration, you may need to alter how you address the IVR server when placing a call.

4.4.1.1 Placing calls to Asterisk®

To place a call to Asterisk®, use the softphone profile you configured in [the previous section \(p21\)](#). Once you have enabled this profile, you only need to dial the extension configured in your `extensions.conf` file, then connect the call.

4.4.1.2 Placing calls to Dialogic®

When placing a call to a Dialogic® server, the call is directed to the IVR server using the dialling address, instead of being included in the user profile [you created earlier \(p21\)](#). In order to place a call to a Dialogic® system, the address should be formatted as:

```
<Extension>@<Server IP>
```

Unlike Asterisk®, dialogic will accept a call on any extension by default. Due to this behaviour, call filtering and line-specific logic is largely done in the *Phone Password Manager* `psynch.ps1` script.

Setting up the *Bravura* Pass Remote API

5

The *Hitachi ID Bravura Pass* remote API enables *Phone Password Manager* and other external applications to access *Bravura Pass* features such as user authentication and password change.

The *Phone Password Manager* server uses the following components to communicate with the API:

- The *Phone Password Manager* service, or the Hitachi-ID VoIP Telephony service.
- The Password Manager API Services (IDAPI and IDAPISOAP) on the *Bravura Pass* server.
- The *Bravura Pass* API DLL (`pspushpass.dll`) installed on the *Phone Password Manager* server during setup.

In order to set up the Remote API on *Hitachi ID Bravura Pass* server, you will need to:

- [Enable the API Service \(p23\)](#)
- [Configure an IDAPI caller \(p23\)](#)



See also:

See [API Service \(idapi\)](#) in the *Bravura Security Fabric Reference Manual* for more information about the API Service (idapi).

5.1 Enabling the API Service

The API Service (idapi) and API SOAP Service (idapisoap) must be running on *Hitachi ID Bravura Pass* server before you can call the API. Both of these services are enabled and started by default, but unexpected shutdowns or maintenance may necessitate a manual restart.

To enable and start the API Service:

1. Click **Manage the system** → **Maintenance** → **Services**.
2. Select  **Hitachi ID (idapi) API Service**.
3. Click **Enable the service**.
4. Click **Start the service**.
5. Select  **Hitachi ID (idapisoap) API SOAP Service**.
6. Click **Enable the service**.
7. Click **Start the service**.

Next:

Configure an IDAPI caller (p23).

5.2 Configuring an IDAPI caller

The API caller is an administrator account on the *Bravura Pass* instance which *Phone Password Manager* uses to implement API calls. The account `_API_USER_TPM` is included in a *Bravura Pass* installation, but is disabled by default.

You can designate any product administrator as the API caller by giving them the **API caller** privilege. You can also create a new product administrator.

To configure the API caller:

1. Click **Manage the system** → **Security** → **Access to product features**.
2. Configure `_API_USER_TPM`, or create a new product administrator, and grant it the **API Caller** privilege.
3. Configure the **IP address with CIDR bitmask** to include the IP address of the *Phone Password Manager* server.
4. Configure a password for the API caller, and click **Update**.
5. Click **Enable** to activate the API caller account.
6. If necessary, see **Password Manager Service (idpm)** in the *Reference Manual* for help configuring the Password Manager service (idpm).

Next:

Install the *Phone Password Manager* software (p26).

See also:

A reference manual for the API, *Hitachi ID Bravura Pass Remote API* guide, is available with your *Bravura Pass* software package or from support@Hitachi-ID.com.

Part III

INSTALLATION AND INTEGRATION

Installing the *Phone Password Manager* Software

6

This chapter shows you how to install the *Phone Password Manager* software. You must install *Phone Password Manager* on the same Windows server where the Dialogic® board or Dialogic® PowerMedia Host Media Processing Software is installed (interactive voice response (IVR) server). If you will be using an Asterisk® server, it is recommended that *Phone Password Manager* and Asterisk® are installed on separate servers.

If you have a hardware phone system and have *not* yet installed a Dialogic® board, or if you require additional information, please refer to [Configuring the IVR Server](#).

6.1 Before you begin

Before you install *Phone Password Manager* software on the interactive voice response (IVR) server, ensure that your:

- Hardware phone system is fully functioning on the IVR server, including voice board, drivers, and all connections.
The Dialogic® product System Service, and Dialogic® product Boardserver service, must be running.
Or
- Dialogic® PowerMedia Host Media Processing Software is installed and fully functioning on the IVR server.
Or
- The Asterisk® server is running, and has been configured to connect to the IVR server.

6.2 Using setup to manage installations

If you do not already have the *Phone Password Manager* software, please contact your Hitachi ID Systems account representative to find out how to obtain the zip file.

Once the zip file is downloaded and unzipped to a temporary folder on the interactive voice response (IVR) server, run the **setup** program that is located at the root of the distribution folder. It gathers initial configuration information and launches the installer (**idtel.msi**).

The **setup** program detects your configuration. If you have:

- Dialogic® boards, then *Phone Password Manager* installs support for hardware phone systems.
- Dialogic® PowerMedia Host Media Processing Software, then *Phone Password Manager* installs support for softphone systems.
- By default, **setup** will install the Asterisk® module. This module includes the *<Instance>\service\Asterisk.cfg* file, as well as the **tpm** service, and **tpm.cfg** configuration file. These files are used to configure Asterisk® IVRs only.

Note: For an Asterisk installation, *Phone Password Manager* will **not** install the *idtel.cfg* file, nor will it start the VoIP Telephony service. These files are only installed when a Dialogic® configuration is detected on the system.

Note: You must run **setup** as a member of the Administrators group.

Currently, the **setup** program is used to:

- Install *Phone Password Manager* (p28)
- Manage *Phone Password Manager* instances (p30)
- Uninstall *Phone Password Manager* (p30)

6.3 Using setup to install the software

To install the *Phone Password Manager* service:

1. Run the **setup** program (p27).
2. If you already have *Bravura Pass* instance installed on the Windows server, **setup** displays the **Select an instance to configure** page. Click **Install New** to proceed.
3. Select **Hitachi ID Bravura IVR**, then click **Next**.
4. Enter a unique instance name, and optionally a description.
Click **Next**.
The **setup** program performs a pre-installation check and verifies all of the requirements for installation.
5. If all of the checks are successful, click **Next** to proceed with the installation.

Note: If any of the pre-install checks produce warnings or errors, click **Report** for details.

The **setup** program launches the **Hitachi ID Telephone Password Manager <instance> Setup Wizard**.

6. Click **Next**.
7. Read and accept the license agreement.
Click **Next**.
8. Type the location of the *Phone Password Manager* license file. Alternatively, you can use the **Browse** button to select the location of the file.
Click **Next**.
9. Choose the setup type that best suits your needs:
 - Click **Typical** to install *Phone Password Manager*, and the Asterisk® module without any additional features.
Or,
 - Click **Complete** to install *Phone Password Manager* with all additional features: Asterisk® module, Speech Service, RSA Audio files, and HDD Encryption Audio files.
Or,
 - Click **Custom** to select which additional *Phone Password Manager* features to install. Select only the items you want to install:
 - Asterisk Module – See [Configuring Asterisk® Software](#).
 - RSA Audio files – Audio files for RSA Authentication Manager targets.
 - Speech Service – See [Speech Recognition and Text-to-Speech](#) for details.
 - HDD Encryption Audio files – Audio files for Hard Drive Encryption targets.

Note: It is recommended that you do *not* change the install **Location**.

Click **Next**.

10. If you chose a custom or complete installation, choose the locations for the:

Directory to store log files This directory should be *unique* for each instance. The default is *<Program Files path>\Hitachi ID\Telephone Password Manager\Logs\<instance>*
Click **Next**.

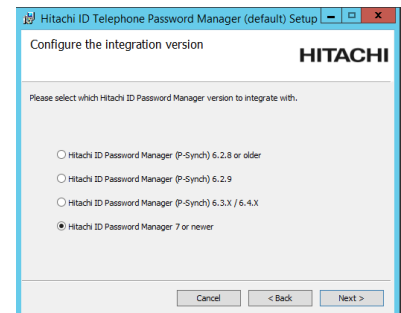
11. Type and confirm a password for the **Service user ID**.

If the account does not already exist, the installer creates it with the specified password. The default user ID is `.\psadmin`.

This is the account the *Phone Password Manager* service will run as. It is automatically added to the server's *Administrators* group. You can change this group membership as long as the account has the right to logon as a service.

Click **Next**.

12. If you chose a custom installation, select which version of *Hitachi ID Bravura Pass* to integrate with.



Click **Next**.

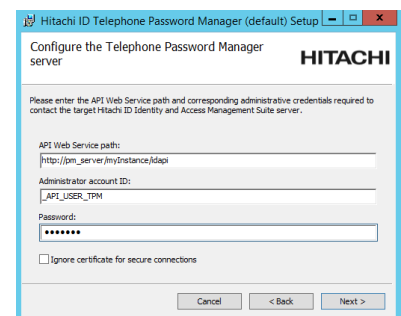
13. Type the communication key. Enter the same key here as you did on the main *Bravura Pass* server (communication key (or Master Key)). If you did not record the key in a secure location, copy the `idmsetup.inf` file from `\<instance>\psconfig\` on the *Bravura Pass* server to the same location as the installer. The installer will extract the communication key (or Master Key) value from the file.
14. Configure the *Phone Password Manager* server:

API Web Service Path This is the URL for the *Bravura Pass* server to the idapi's web interface. The default path should be:
`http://server/instance/idapi.`

Administrator account ID This is the API caller's user ID. For example, this could be set to `_API_USER_TPM` for the default API caller. See [User types and access rules](#) for details on API callers.

Password This is the API caller's password.

Ignore certificate for secure connections Enable this box to ignore the SSL certification check.



Click **Next** to verify that the information provided for the API Web Service Path is correct. The installer will check that it is able to contact the API Service (idapi), and will display an error if it cannot.

15. Click **Install** to start the installation.

The installer begins copying files to your computer.

16. Click **Finish** to exit after the installer has **Completed the Phone Password Manager (<instance> Setup Wizard)**.

Phone Password Manager is now installed on your interactive voice response (IVR) server.

Next:

- [Integrate Bravura Pass and Phone Password Manager.](#) (p31)

6.4 Managing alternate instances

You can use **setup** to install one or more *Phone Password Manager* instances. For example, if you are running multiple instances of *Hitachi ID Bravura Pass*.

You can also use the **setup** program to manage instances. The **setup** program detects that you already have an instance of *Bravura Pass* installed, and presents you with a list of choices.

Click:

Modify	to add or remove features for an instance
Repair	to reinstall any files that may be missing from the instance
Uninstall	to remove an instance
Install new	to install another instance

6.5 Uninstalling the software

To uninstall *Phone Password Manager*:

1. Run the **setup program** (p27).
2. Click **Uninstall** next to the instance that you want to remove.
3. If prompted to remove the service user, select **Yes** or **No** to continue.

Note that you are not prompted to remove the service user account if:

- It existed before the installation of the *Phone Password Manager* software
- It is used by other instances of *Phone Password Manager*.

4. Click **Close** to exit the installer.

You can also use **setup** to remove additional components.

Integrating *Bravura Pass* and *Phone Password Manager*

7

Read this chapter to learn about the steps that you must perform in order to configure the *Phone Password Manager* service, and to customize the interaction between *Bravura Pass* and *Phone Password Manager*.

Integrating *Bravura Pass* and *Phone Password Manager* provides:

- Self-service password reset and password synchronization from a telephone
- Self-service token management from a telephone
- Active enrollment of biometric voice print sample registration

7.1 Integration Mechanisms

Hitachi ID Bravura Pass exposes APIs suitable for use by an interactive voice response (IVR) system using HTTPS web access. The API implements strong encryption policies for all connections, allowing you to securely locate the IVR system at a different site from the *Bravura Pass* server.

7.1.1 Web service

A web service allows IVR systems and other applications to remotely invoke methods on the *Bravura Pass* server to perform functions such as user and account lookup, security question authentication, random password generation, and to initiate password resets or to clear intruder lockouts. Remote applications normally access the web service over HTTPS for security. Security is also accomplished by use of a secure transport layer, an API user and password, and a one-time-use session ID. Organizations wanting an extra level of security can limit the range of IP addresses that are permitted to access the API to just legitimate IVR systems or other applications.

IVR systems that support integration using web services include those from Intervoice and Nortel/Periphonics.

7.1.2 Example function call sequence

The touch-tone-authenticated password reset process, described in [Password resets from a telephone](#) in the *Self-Service Anywhere Implementation Guide (self-service-anywhere.pdf)* is implemented by calling the following library functions using any of the API variants described above:

- **Login** – initialize the API session and connect with a valid username and password. Required to start any API session.
- **UserIVRList** – Return a list of users matching a numerical ID. This function returns all users which match the numerical identifier. For more information on mapping users to a numerical ID, see: [Mapping user IDs to telephone keypads](#)
- **UserQuestionsGet** – get a random selection of authentication questions that the user might be required to answer. The IVR system must be pre-programmed with speech recordings for every available question, or a text-to-speech engine.
- **UserAnswersValidate** – validate that the answers keyed in by the user are correct.
- **PasswordRandomGet** – called at least once, and possibly several times, to generate a random valid password, and read it out to the user as a possible new password.
- **UserPasswordSync** Reset one or more passwords associated to a user account. Depending on the parameters passed, this call can also allow the user to reset individual passwords, rather than every one.
- **Logout** – close the current API session.

See [API Service](#) in the Bravura Security Fabric *Remote API* for more information on writing customized API calls.

7.1.3 Event actions

The following event actions are supported for *Phone Password Manager*, and can be configured on the *Hitachi ID Bravura Pass* server:

- **ET_ADMIN_RESET_SUCCESS** Triggered when *Phone Password Manager* successfully attempts to reset a user's password using the **UserPasswordSync** API Service (idapi) call.
- **ET_ADMIN_RESET_FAILURE** Triggered when *Phone Password Manager* fails an attempt at resetting a user's password using the **UserPasswordSync** API Service call.
- **ET_ADMIN_UNLOCK_SUCCESS** Triggered when *Phone Password Manager* successfully attempts to unlock a user's account using the **UserAccountsUnlock** API Service call.
- **ET_ADMIN_UNLOCK_FAILURE** Triggered when *Phone Password Manager* fails an attempt to unlock a user's account using the **UserAccountsUnlock** API Service call.

For more information on using these API Service calls, see: [SOAP API Reference](#) in the Bravura Security Fabric *Remote API*.

7.2 Editing the configuration file

The *Phone Password Manager* service uses one of two configuration files, named `tpm.cfg`, or `idtel.cfg` to determine:

- The names of script files that define call flow and logic
- Supported languages
- Enrollment types
- Audio file types
- The play back volume adjustment
- Dialogic® voice board setup:
 - The number of lines supported on the boards
 - The type of boards
 - The number of boards installed
 - Whether or not the boards use SCBus routing
 - Any custom tones the boards support (if loop-current detection is not supported)
- Dialogic® Host Media Processing Software setup:
 - Playback volume levels
 - The number of lines supported
 - Audio file type
 - Audio codec

When you install *Phone Password Manager*, the installer program automatically detects your interactive voice response (IVR) system configuration and creates this file in the `\<instance>\service\` directory on the IVR server. The `idtel.cfg` file is only created when a Dialogic® Voice board, or Dialogic® PowerMedia Host Media Processing is detected. The `idtel.cfg` file is used in place of `tpm.cfg` to handle calls received from Dialogic® equipment.

You can modify `idtel.cfg` if your system configuration differs from what was auto-detected, or if you want to modify the default settings. The file includes instructions for modifying each setting in-line as comments.

There are settings within `idtel.cfg` that only apply to VoIP and softphone systems, which can be found under “VoIP Proxy Server Registration”.

Note: You must restart the *Phone Password Manager* service in order for your configuration changes to take effect.

If you are using a Dialogic® voice board and loop disconnect supervision is provided as a tone or cadence, you can configure *Phone Password Manager* to detect the condition of the calling party prematurely hanging up. This can be done either by defining the disconnect tone or cadence in `idtel.cfg`, or by creating a TSF file containing call progress tone information.

If loop disconnect supervision is provided with a loop current drop, it is not necessary to define this information, and the change in line status will be detected automatically if your voice board supports loop current supervision.

7.2.1 Editing the Asterisk® configuration file

Phone Password Manager uses the file `asterisk.cfg` to determine the following settings when interacting with an Asterisk® server:

- The port used to listen for Asterisk® communication.
- The IP range to listen for.
- The number of channels upon which communication can be accepted.
- The list of IP addresses with which communication is permitted.
- The name of the Asterisk® server's audio file directory
- If *Phone Password Manager* should automatically upload files to the Asterisk® server.
- Which folders to exclude if auto-uploading is enabled.
- Whether or not to keep the temporary files created during auto-upload.
- (Optional) Which call logic scripts to run when a call is received from an Asterisk® server.

Phone Password Manager automatically configures this file to the default settings when the Asterisk® module is installed, however you may use these options to restrict which communications your *Phone Password Manager* server accepts, for the sake of security.

7.2.2 Using 3CX PBX systems

Phone Password Manager is also capable of registering to a 3CX PBX system, in a manner similar to the Dialogic® configuration. To enable a 3CX configuration, alter the `idtel.cfg` file to include the following settings:

```
Registration "" = {
    Server = 10.0.1.1
    Realm = "3CXPhoneSystem" // SIP only
    PhoneNumber = 333
    Password = "333" // SIP only
}
```

Setting the `Realm = "3CXPhoneSystem"` value will enable *Phone Password Manager* to register a 3CX system when the next call is placed.

7.2.3 Defining disconnect tones in idtel.cfg

Disconnect tones can be explicitly defined within `idtel.cfg`. To do this:

1. Create a KVGroup in `idtel.cfg` with one of the following types, according to the example provided in `idtel.cfg`:

- SingleTone
- SingleCadenceTone
- DualTone
- DualCadenceTone

Populate the KVGroup with the tones, timings, and tolerances provided by the local loop on disconnect. The name of the KVGroup can be arbitrary.

2. Define the **ToneType** parameter and set it to the name of the KVGroup defining the tone or cadence.
3. Restart all the *Phone Password Manager* services.

Note: This method, called **Global Tone Detection**, is unreliable on many Dialogic® cards. It is strongly recommended that a TSF file be specified if possible.

7.2.4 Defining disconnect tones with a TSF file

Dialogic® cards can be configured to use a TSF file to recognize call progress tones, including loop disconnect. The contents of this file must be defined according to the environment providing the call progress tones to the voice board. To implement this:

1. Use the PBX Expert application provided as part of the Dialogic® drivers to discover the call progress tones. It is only necessary to discover the disconnect tones.
2. Before saving the TSF file, mark the discovered tone set for consolidation, consolidate it, compile it, and enable it. Save it under the Dialogic® data directory.
3. Use the Dialogic® Configuration Manager to enable TSF file support globally and set the TSF file name to the file you created.
4. Restart the Dialogic® card.
5. Restart all the *Phone Password Manager* services.

If this process was successful, when the *Phone Password Manager* service starts it will print an informational log message containing the text `TSF has been loaded successfully`. It will now treat the specified disconnect tones as a loop current off event.

7.3 Writing call logic scripts

The *Phone Password Manager* service uses scripts to determine the logic and flow of each call to the interactive voice response (IVR) system. These scripts are used to define the workflow of any call received, including which sound files to play to the user, what kind of user input to expect following a prompt, and what operations to perform based on that user input. Descriptions for each section of the script and instructions for customizing the file are included in-line as comments.

The call logic scripts are written in PSLANG – a scripting language with a syntax much like C, but with a large set of built-in functions, some of which are specific to *Phone Password Manager*. The *Phone Password Manager*-specific functions can:

- Interact with the Dialogic® voice boards and Dialogic® PowerMedia Host Media Processing Software.
- Interact with the *Bravura Pass* remote API.
- Perform voice print related operations.

Phone Password Manager is shipped with a default script, **psynch.ps1**, that is configured to guide users through log in and authentication using their challenge-response questions stored in *Bravura Pass*. After login, the script offers users the option to perform password resets, account unlocks, and SecurID token management (If available).

Script files must be located in the `\<instance>\script\` directory on the IVR server. If required, you can change the name of the script file or enable multiple script files by modifying **idtel.cfg** (p33).

To use the VoiceVantage script instead of the DTMF script, rename the **psynch.voiceprint.ps1** sample script to **psynch.ps1**, and put it in the `\<instance>\script\` directory.

There are several global variables which can be called from within a call logic script:

Table 7.1: call logic global variables

Variable	Description
\$trunk	An integer representing the current line number for this call.
\$lineMode	An integer representing the call mode. Values include: <ul style="list-style-type: none"> • 0 - Auto-answer mode. (Default) • 1 - Inbound mode. • 2 - Outbound mode. See Call Modes for details.
\$supportedLanguages	A space-delimited String containing the languages defined in the configuration file.
\$enrollmentTypes	A space-delimited String containing the enrollment types defined in the configuration file.
\$callerId	A string containing the caller's phone number, or URI.
\$callerName	A string containing the caller's name, if available.

... continued on next page

Table 7.1: call logic global variables (Continued)

Variable	Description
\$calledId	A string containing the number or URI dialled by the caller.

See also:

- The [Introduction](#) in the *PSLang Reference Manual* for more information about the PSLANG and its *Phone Password Manager* specific functions.
- The *Phone Password Manager* samples* directory for additional call logic scripts, including a script that contains voice-print related operations (`psynch.voiceprint.ps1`).

7.4 Managing audio files

Phone Password Manager is shipped with most of the audio files necessary for complete operation in English, however you must provide and configure the audio files for:

- Any additional questions that you will add for [touch-tone authentication](#) (p37).
- [The names of any custom target systems available to IVR users.](#) (p38).
- Any modifications that you will make to the [call logic scripts](#) (p39).

7.4.1 Adding custom authentication questions

In order to create custom questions with which to authenticate users of *Phone Password Manager*, you must configure the question definition on your *Bravura Pass* instance, and provide an audio file which *Phone Password Manager* will associate to that question.

Firstly, define your new authentication questions. See [IVR with touch-tone authentication](#) in the *Self-Service Anywhere Implementation Guide* for more information on configuring question sets.

When you define the new question, the following conditions must be met:

- The `description` field *must* be in the format: `!!!DEFAULT_PREDEFQSET_<QID>_DESC`
- A KVG file has been configured in `<Instance>\design\custom` to translate the machine-readable question definition for each language you wish to support. See: the [Bravura Security Fabric Documentation](#) for more information.
- An audio file exists on your interactive voice response (IVR) server, in each `<Instance>\audio\<Language>` directory, titled `QD-PREDEFINED_<QID>`, that corresponds to the newly defined question.

To properly ID additional questions:

1. Modify **en-us-errmsg.kvg**, located in the `<instance>\design\src\common` directory, to include a new tag for the additional question.
2. Use the tag ID as the question description.
3. Generate and install the new skin files.

The vocal should prompt the user to type the answer the question followed by pound; for example, *"Enter the year you graduated high school, followed by the pound key."*

See [User Authentication](#) for details.

7.4.2 Defining custom target systems

Each target system that users can reset their passwords or unlock their accounts for.

These files must be named `reset_<target ID>.wav` and `unlock_<target ID>.wav` respectively.

The vocal should prompt the user to perform the action for the specific target system; for example, *"To unlock your account on Windows" or "To reset your password on Unix"*.

Phone Password Manager is shipped with a set of files for these common target systems:

- Microsoft Active Directory – **reset_AD.wav, unlock_AD.wav**
- Novell Directory Services (NDS) – **reset_NDS.wav, unlock_NDS.wav**
- Microsoft Windows server – **reset_NT.wav, unlock_NT.wav**

If you want to use these shipped files, simply rename each file so that the `<target ID>` portion matches your actual target IDs.

7.4.3 Defining custom target system groups

When a user has accounts in more than one target system group, *Phone Password Manager* offers them the ability to select those groups when initiating a password reset.

In order to provide the user with a menu from which to select target groups, *Phone Password Manager* will individually spell out each letter of the custom group's ID value.

In order to configure a custom audio file to present these target groups to users, create a new audio recording of the target group's name, and save it to the appropriate `<Instance>\Audio\<Language code>` directory, depending on the language the audio file will be used for. The name of this file should exactly mirror the target group's ID value: **<Group ID>.wav**

Note: You will need to restart your *Phone Password Manager* Windows services in order for the changes to take effect. Additional steps for Asterisk® backends may be required. For more information, see [Asterisk® audio files](#).

7.4.4 Supporting custom call logic

You will need to configure the audio files for any modifications that you will make to the [call logic scripts](#) (p36).

If you are not using the default call script, or if you have modified it, ensure that you have appropriate audio files for each `PlayFile()` or `PlayFileEx()` function. See: [Dialogic Functions](#) in the *PSLang Reference Manual* for more details on these functions.

All audio files used for play back must be stored in the `<instance>\audio\<lang>-<locale>` directory on the *Phone Password Manager* server. The value of `<lang>-<locale>` refers to the language and locale of the user. For example, use `en-us` for United States (`us`) English (`en`).

Additionally, all audio files must be recorded in the format specified in the `idtel.cfg` file (p33). The default is a MuLaw-encoded, PCM Wave file, with 8 bit sample size, 8kHz sample rate, and 64kbps bit rate. Audio files should always be recorded in monaural format.

7.5 Adding additional languages

To add a foreign language:

1. Add a new `Language` key-value pair to the `idtel.cfg` file (p33) and restart the *Phone Password Manager* service.

For example, add the line: `Language = fr-ca`

2. In `psynch.ps1`, set the value:

```
$selectlang = 1;
```

This value enables the language select menu, which prompts users to select their language preference before entering the main menu.

3. Create a subdirectory in `<instance>\audio\` that matches the language code (`<lang>-<locale>`) for the language you will be adding.

For example, create a directory named `fr-ca`, to handle files for Canadian French. This code must match the key-value pair configured in step 1.

4. Record a complete set of new vocals in the appropriate language, and save the files in the newly created directory. These new files should use the same names as their counterparts in the default `en-us` folder.

5. (Optional) Enable language support by extension. Uncomment the following code block in `psynch.ps1`:

```
// Extension to language mapping table:
var $langmapping[];
//$langmapping["777"] = "en-us";
//$langmapping["888"] = "fr-fr";
```

Enabling this code will configure *Phone Password Manager* to automatically provide service in an alternate language, depending on the phone extension used to dial in to the system.

Note: *Phone Password Manager* can offer users any number of language options by default. However, if you wish to offer ten or more language options in the same menu, additional configuration is required to allow *Phone Password Manager* to prompt for, and accept two-button input in the language selection menu.

To help with translations, the `vocal-script.txt` file in the `<instance>\audio\en-us\` directory contains a complete listing of all shipped English-language files and their transcriptions.

7.5.1 HDD Encryption Audio files

Phone Password Manager includes support for HDD (Hard Disk Drive) Encryption agents used by *Bravura Pass*, such as `agtmcee6` for McAfee McAfee Endpoint Encryption 6.x.

To support these functions, *Phone Password Manager* requires the "HDD Encryption Audio files" package to be installed on the system.

The HDD Encryption package includes several audio files which allow *Phone Password Manager* to properly present to users the HDD Encryption functions.

Note: If the HDD Encryption Audio files package is added to a running installation which uses the Asterisk® backend, a *Phone Password Manager* service restart is required to help propagate the new audio files to the Asterisk server.

No configuration is required for the HDD Encryption Audio files, as the *Phone Password Manager* will automatically recognize the relevant agent and play the corresponding audio when required.

7.6 Mapping user IDs to telephone keypads

Users identify themselves to the interactive voice response (IVR) system by typing their IVR IDs on telephone keypads. An IVR ID is the numeric representation of one of their *Hitachi ID Bravura Pass* profile and request attributes. Therefore, each user that logs into the *Phone Password Manager* system must first exist in *Bravura Pass*.

You can change the profile and request attribute that is used as a source of users' IVR IDs by using the TPM ID ATTR option. By default, profile IDs are used as the source of IVR IDs.

If users' IVR IDs contain punctuation, they should be instructed to skip all punctuation, including # and *,

when entering their IDs for validation. For example, the IVR ID O'Hare should be entered as 64273.

Note: *Phone Password Manager* assumes that telephone keypads are mapped according to the current international standard (ITU E.161).

If this is not the case at your organization, contact support@Hitachi-ID.com for assistance.

If *Phone Password Manager* discovers multiple matching usernames, then all matching users are listed with a maximum of 9 users per page. At the end of any page, you can:

- Press 0 to go back and enter another *Bravura Pass* profile ID.
- Press # to repeat the list of users, starting from the first page.

If no selection is made, the next page of users is played. At any time, you can:

- Press * to skip all previously played users, and play the remaining users, with a maximum of 9 users per page.

7.7 Using multiple *Bravura Pass* instances with Phone Password Manager

You can use multiple *Bravura Pass* instances with a single *Phone Password Manager* server, by adding the LoadInstanceCFG function in the **psynch.ps1** (p36) file; for example:

```
LoadInstanceCfg("C:\\Program Files\\Hitachi ID\\Telephone Password Manager\\My  
Instance\\script\\myspushpass.cfg")
```

Multiple *Bravura Pass* instances require multiple phone lines or extensions.

Part IV

CONFIGURATION

User Authentication

8

Secure methods of authenticating users to *Phone Password Manager* include:

- **Touch-tone authentication (Numeric questions and answers)**

Users key-in their answers to personal questions using a telephone key pad.

You can set up *Phone Password Manager* to authenticate users by prompting them with questions from their *Hitachi ID Bravura Pass* question and answer profiles.

- **Biometric voice print verification**

Users speak one or more phrases so that their voice can be compared to a previously registered sample.

In order to use voice print verification, users must first register their voice samples. You can set up *Bravura Pass* to facilitate and secure the registration process.

Read this chapter to learn how to set up question sets in *Bravura Pass* that *Phone Password Manager* can use to authenticate users. See [IVR with voice print authentication](#) in the *Self-Service Anywhere Implementation Guide* to learn how to setup self-service interactive voice response registration.

8.1 Setting up IVR question sets

In order for *Phone Password Manager* to use questions and answers from *Hitachi ID Bravura Pass* to authenticate users, the following conditions must be met:

- Question sets and questions must be set up correctly for *Phone Password Manager* integration.
 - *Phone Password Manager* question sets must be pre-defined, and all questions must have all-numeric answers so they can be easily entered from a telephone keypad.
 - You can either add a new question set specifically for the *Phone Password Manager*, or use the existing pre-defined question set.
 - The question set must have **Ask telephone users to answer questions from this set** enabled.
- Users must complete their security question profiles.
- Users' completed profiles must include at least <N> questions from each question set that can be used for *Phone Password Manager* integration.

Where <N> is equal to the question set's **Number of questions to ask during authentication** setting.

8.1.1 Using the default question set

Bravura Pass is shipped with a pre-defined question set, `DEFAULT_PREDEFQSET`, that contains three questions that can be used for *Phone Password Manager* authentication:

What is your favorite or lucky number?
What was your first telephone number?
On what year did you purchase your first car?

8.1.2 Adding question sets

If you do not want to use the default question set (`DEFAULT_PREDEFQSET`) for touch-tone authentication, or if you want to strengthen the authentication process, you can add more *Phone Password Manager* specific question sets.

To do this:

1. Click **Manage the system** → **Policies** → **Question sets** → **Pre-defined questions**.
2. If *Bravura Pass* displays a list of existing question sets, click **Add new...** at the bottom of the list.
3. Enable the **Ask telephone users to answer questions from this set** checkbox.
4. Set appropriate options for the new question set.
See [Question sets](#) for details regarding question set options.
5. Click **Add**.

Next:

[Add questions \(p46\)](#).

8.1.3 Question set options

Table 8.1: Question set options

Option	Description
ID	(Required) A unique identifier for the new question set.
Description	(Required) The description that <i>Hitachi ID Bravura Pass</i> displays to users.
Enabled	Select this checkbox to enable the question set for use with <i>Bravura Pass</i> . This is enabled by default.
★ Users allowed to edit answers	Select this checkbox to allow regular users to edit answers in a pre-defined question set. This is enabled by default. If disabled, this question set will not be accessible through the <i>Update security questions (PSQ)</i> module.

... continued on next page

Table 8.1: Question set options (Continued)

Option	Description
† Users allowed to edit questions/answers	Select this checkbox to allow regular users to edit questions and answers in an external question set. If disabled, this question set will not be accessible through the <i>Update security questions</i> (PSQ) module.
Minimum number of questions user profiles should contain	<p>(Required) The number of questions to which a user must provide answers. Set the value to 0 to make a question set optional.</p> <p>When you change this setting, <i>Bravura Pass</i> automatically schedules the psdonechk program to run once to check compliance. To modify the scheduled job, click Manage the system → Maintenance → Scheduled jobs, then select PSDONECHK.</p>
Answers in the set must be unique	Select this checkbox to prevent users from giving the same answer to two different questions.
Help-desk permissions	<p>Select a value to control how help desk users can interact with questions and answers in this set:</p> <ul style="list-style-type: none"> • Not allowed to view security questions • Requires authentication with security questions • Allowed to view security questions
Ask users to answer questions from this set	Select this checkbox to prompt users to answer questions from this set during authentication.
Ask telephone users to answer questions from this set	Select this checkbox to prompt <i>Phone Password Manager</i> users to answer questions from this set during interactive voice response (IVR) authentication.
Number of questions to ask during authentication	<p>(Required) Set the number of questions to randomly draw from this set to ask a user during authentication.</p> <p>The number of questions to ask cannot exceed the Minimum number of questions user profiles should contain. For an external question set, set this number to -1 if you want to get all questions from the external source.</p>
Page number for question set to be displayed in	<p>(Required) Users may be prompted to answer questions in a sequence of authentication pages. Type an integer in this field to make questions in this set appear before or after questions in other sets.</p> <p>The page number must be unique for external question sets.</p>
Algorithm to match answers during authentication	<p>Select the algorithm to use when comparing the answers typed during authentication to the answers stored in the user's profile:</p> <ul style="list-style-type: none"> • Exact string match • Case-insensitive • Only alpha characters • Soundex algorithm • One extra character • N extra characters

... continued on next page

Table 8.1: Question set options (Continued)

Option	Description
† External program	<p>(Required) The name of the external program, or <i>authentication plugin</i>, to run.</p> <p>See External question set plugin tasks for details on the authentication plugin and how to correctly configure the question set according to your needs and the capabilities of the program.</p>
† Check that user has an account on target system	<p>Determines how <i>Bravura Pass</i> identifies users on an external system.</p> <p>See Identifying users on external systems for details.</p>
† External program provides answers along with questions	<p>Select this checkbox if an external program will provide both questions and answers for <i>Bravura Pass</i> to display and validate.</p> <p>If this checkbox is not selected, then the program will accept answers from <i>Bravura Pass</i> and validate them.</p> <p>See External question set plugin tasks for details.</p>

8.1.4 Adding questions

In order for a question to be suitable for touch-tone authentication, it must have the following characteristics:

- Answers are private – relatively hard for anyone other than the user to come by.
- Answers are easy – users should be able to quickly and reliably answer the question without having to remember anything new, and with a low likelihood of making mistakes.
- Answers are all-numeric and have a fixed length.

If you are defining a new question for which a sound file does not exist by default, the question's **Description** field *must* be formatted as: `DEFAULT_PREDEFQSET_<QID>`. This formatting is mandatory in order to associate a new question to its respective sound file.

The QID defined in the description field is used to uniquely address the sound files on the IVR, and should be unique to every custom question you wish to define.

See: [Adding custom authentication questions](#) for more information on defining custom *Phone Password Manager* authentication questions.

See [Question sets](#) to learn how to add a question to a pre-defined question set.

Next:

[Record question vocals for new questions \(p47\)](#)

8.1.5 Recording question vocals

When users call into the *Phone Password Manager*, the system plays vocals (sound files) that prompt the users to prove their identities by keying in numerical answers to the questions that they have configured in *Bravura Pass*.

Phone Password Manager is shipped with vocals for each numeric question in the default pre-defined question set (DEFAULT_QSET). If you have added additional questions for touch-tone authentication you must record a vocal for each new question, in each supported language.

Vocal files must be named QD-PREDEFINED_<QID>.wav, and must be located in the <instance>\audio\<lang>-<locale> directory on the *Phone Password Manager* server. Note that the value of:

- <QID> must match the number defined in the DESCRIPTION field for the question as it was defined in *Bravura Pass*. The description field should always be formatted as: DEFAULT_PREDEFQSET_<QID>.
- <lang>-<locale> corresponds to language that the vocal was recorded for.

See [Managing audio files](#) for more information on *Phone Password Manager* audio files.

Voice Print Enrollment

9

A voice print is a form of biometric authentication where the characteristic being measured is the timbre, tone, speed and volume of the user's voice. Typically, the user speaks a phrase during enrollment, then later repeats that phrase as part of the authentication process.

9.1 Setting up voice print enrollment

To set up voice print enrollment:

1. Install VoiceVantage VoiceCheck SDK on the *Phone Password Manager* server.
2. Copy `psynch.voiceprint.psl` from the `samples*` directory to the `\<instance>\script\` directory.
3. Modify `C:\Program Files (x86)\Hitachi ID\Telephone Password Manager\<instance>\service\idtel.cfg` by changing `ScriptName` as follows:

```
ScriptName = "psynch.voiceprint.psl"
```

4. Restart the *Phone Password Manager* service.
5. Enable and configure the *Generate voice print enrollment PIN* (PSI) module on *Bravura Pass* server. Configure the following settings:
 - **PSI ENABLED:** On
 - **PSI RANDOM DIGITS:** 4
 - **PSI RANDOM EXPIRY:** 600
6. Restart the Password Manager Service on *Bravura Pass* server.

Phone Password Manager is now configured and ready for users to enroll via a PIN.

9.2 Configuring voice print options

The tone of your voice varies each time you authenticate against your voice print. If your voice sounds too different from your voice print, *Bravura Pass* might not be able to properly authenticate you.

The following options, located in `idtel.cfg` or `tpm.cfg`, help you configure the sensitivity thresholds, and the recording time for voice print authentication.

Table 9.1: Voice print authentication options

Option	Description
DefaultVoicePrintConsistencyThreshold	This is used during voice print enrollment to ensure that the multiple voice samples are consistent enough. Set a value between 1 and 99 to control the consistency threshold. The higher the number, the more closely-matched the samples must be. Default value is 50.
DefaultVoicePrintVerificationThreshold	This is used during voice print verification to ensure that the spoken voice matches the recorded voice. Set a value between 1 and 99 to control the verification threshold. The higher the number, the more closely-matched the samples must be. Default value is 55.
VoicePrintMaxRecordTime	The maximum recording time for a single recording. If the recording is longer than this time, the voice will be cut off after the maximum time reached. Default value is 6 seconds.
VoicePrintSilenceToEnd	The seconds that the voice print system will wait until it stops recording; for example, if set to 2, the system will wait for 2 seconds after the voice stopped to finish recording. Default value is 2 seconds.
ReplayVoicePrintBadSample	If set to 1, the voice print system will replay a bad sample, where the signal can be "too loud", "too quiet", "too short" or anything where VoiceVantage is "Unable to extract recording sample". If set to 0, nothing will be played back. Default value is 1.

Note: If you change the values of these options in `idtel.cfg`, then you must restart the *Phone Password Manager* service.

9.3 Testing voice print enrollment

To test voice print enrollment:

1. Log into the *Generate voice print enrollment PIN* (PSI) module as a regular user.
2. Click **Generate voice print enrollment PIN**.
3. Place a phone call to the interactive voice response (IVR) server.
4. Select **1** to enroll.
You are prompted for a PIN to authenticate.
5. Type in the user's PIN.
You are now ready to do voice print enrollment.

9.4 Testing voice print enrollment (command line)

Description

Use the **vpcmd** program, installed with *Phone Password Manager*, to test the generation or verification of voice print audio files.

Usage

```
vpcmd.exe -c (generate|verify) -f <file> -u <userID> -vi <n> (-ei <n>|-en <n>) [ -p "<payload>" ] [ -s 1|0 ]
```

Table 9.2: vpcmd options

Option / Argument	Description
-c <generate verify>	The voiceprint command to perform. Valid commands: generate, verify (required)
-ei <1/2/3>	The enrollment index to generate (users attempt to record speech 3 times).
-en <1/2/3>	The enrollments to verify.
-f <file>	The speech audio file (required)
-l	Prevent voiceprints being running concurrently.
-p <CELL/LAND>	The payload to attach to the generated voiceprint.
-s <1/0>	Whether to save the speech voice: 1 to save, 0 to ignore.
-u <userID>	The user ID (required)
-vi <n>	The verification phrase index (required)

Examples

To generate voice print audio file "a.wav" with user "user1" using cellphone as the recording type:

```
vpcmd.exe -c generate -f a.wav -u user1 -vi 2 -ei 1 -p "CELL"
```

To verify the user voice print with "a.wav" on "user2" using "land-line".

```
vpcmd.exe -c verify -f a.wav -u user2 -vi 2 -en 2 -p "LAND"
```

9.5 Removing voice print enrollment data

Description

The **vputil** program is installed with *Phone Password Manager* and helps to clean the voice print database by removing enrollment data for users who do *not* have a valid *Bravura Pass* profile.

Usage

```
vputil.exe [query|list|clean][<options>]
```

Table 9.3: vputil options

Option / Argument	Description
query	Reports specified user's verification threshold and enrollment consistency threshold. It also shows all enrollments, verification phrases and renditions.
list	Lists users who have completed voice print enrollment, but do <i>not</i> have a valid <i>Bravura Pass</i> profile. Optionally, this can be used in conjunction with the <code>-a</code> option to list all enrolled users, instead of only listed those without valid profiles.
clean	Removes enrollment data for users who have completed voice print enrollment, but do <i>not</i> have a valid <i>Bravura Pass</i> profile. For example, this option removes data for users who have previously enrolled, but their profiles have since been deleted.
-a	Lists <i>all</i> enrolled users instead of only listing those without valid profiles. Used in conjunction with the <code>list</code> option.
-n	Searches for and automatically removes enrollment data for users who do <i>not</i> have a matching <i>Bravura Pass</i> profile.
-t <target system ID>	Specifies which target system to search. If a target system is <i>not</i> specified, then the default target system is chosen.
-u <list file>	Specifies the name of the list file that specifies which users to clean. Used with the <code>clean</code> option.
-w	Specifies the voice print interface DLL. By default, vputil.dll is used.

Examples

To list all users who have completed voice print enrollment:

```
vputil.exe list -a
```

To list users from Active Directory target “AD” who have completed voice print enrollment:

```
vputil.exe list -t AD
```

To remove enrollment data for users who have completed voice print enrollment but do *not* have a valid *Bravura Pass* profile:

```
vputil.exe clean -n
```

Speech Recognition and Text-to-Speech

10

Phone Password Manager supports both speech-to-text (speech recognition) as well as text-to-speech (TTS). Speech recognition converts spoken words to text, and TTS can playback text information as spoken words.

To support these functions, *Phone Password Manager* requires a speech engine to be installed on the system. *Phone Password Manager* uses Microsoft Speech API as the programming interface, and supports SAPI versions 5.1+. However, all SAPI-compliant speech engines can be utilized by *Phone Password Manager*.

Speech recognition is provided by the Speech Service, which is installed during a complete installation or if selected during a custom installation.

Note: The Speech Service can only be installed once on a *Phone Password Manager* server. If you install a second instance of *Phone Password Manager* on the same server, then the Speech Service will be unable to run on the new instance.

When speech recognition is enabled, users can enunciate their profile IDs, new password values, and perform key recovery strings without having to use the numeric keypad.

To set up speech recognition:

1. On the *Phone Password Manager* server, copy **psynch.speech.ps1** and **speech.ps1** from the samples* directory to the <instance>\script\ directory.
2. Modify the **idtel.cfg** file, located in the <instance>\service\ directory, by changing ScriptName as follows:

```
ScriptName = "psynch.speech.ps1"
```

3. Configure the Speech Service.

Modify **idtel.cfg** by changing the SpeechService Dll line as follows:

- For local Speech Service, specify **speechapi.dll**:

```
SpeechService "" = {  
    Dll = "speechapi.dll"  
    //Server = <server>  
    //Port = <port>  
    //Timeout = <timeout>  
}
```

- For remote Speech Service, specify **speechapix.dll**:

```
SpeechService "" = {
  Dll = "speechapix.dll"
  Server = <speech service server name or IP address>
  Port = <speech service port>
}
```

4. Restart the *Phone Password Manager* service.

Speech recognition is now configured and ready to use. To test speech recognition, place a phone call to the interactive voice response (IVR) server and try to use speech instead of the numeric keypad to enter your details.

10.1 Configuring the speech service

The Speech Service can be configured using the following options, which are located in the **idtel.cfg** file:

Table 10.1: Speech Service configuration options

Option	Description
VoiceActivityDetectThreshold	Controls the sensitivity of the input threshold for the Speech Service. The range of possible values for this option is between -54 and +3; the default value is -40. Lowering the numeric value lowers the input threshold, which increases the sensitivity of the Speech Service. Raising the numeric value raises the input threshold, which decreases the sensitivity of the Speech Service. For example, a value of -54 recognizes even the quietest sounds, whereas a value of +3 only recognizes louder sounds.
SpeechRecognitionMode	Controls which speech recognition mode is used. Possible values: <ul style="list-style-type: none"> • 0 – enables “File based mode”, which creates a file in the temp directory before processing the audio file for speech recognition. • 1 – enables “Stream mode”, which does not create a file, but simply analyzes the stream of audio for speech recognition. This was the only mode available in releases before <i>Bravura Pass</i> version 8.0.

By default, stream mode is enabled.

... continued on next page

Table 10.1: Speech Service configuration options (Continued)

Option	Description
KeepIntermediateSpeechFiles	<p>Controls whether or not to save the audio files created when SpeechRecognitionMode is set to “File based mode.” Possible values:</p> <ul style="list-style-type: none"> • 0 – files are <i>not</i> saved; they are deleted after speech recognition is complete. • 1 – files are saved in the temp directory: C:\Documents and Settings\psadmin\Local Settings\temp

10.2 Building .wav files using SAPI

Description

Use the **voicebuild** program to create audio .wav files based on a vocal script .txt file using SAPI.

Usage

```
voicebuild.exe [<options>]
```

Table 10.2: voicebuild arguments

Argument	Description
-f <folder>	The output folder to store the newly-created .wav files.
-o	Overwrite existing .wav files.
-r <rate>	The rate of the speech. This value can range from -10 to 10; the default value is 0.
-d <speech API DLL>	Speech API dynamic load library which can load either speechapi.dll or speechapix.dll. The speechapix.dll file loads from a remote server. By default, speechapi.dll is loaded.
-i <speechinit>	Specifies a KVG file to load speech API initialization configuration.
-s <vocal-script>	The vocal script file, read from standard input by default.
-c <voice>	The voice used to speak the vocal script.
-a	Enumerate available voices.
-l <volume>	The volume of the speech. This value can range from 0 to 100; the default value is 100.

Examples

- List all the installed voices on the current system:

```
voicebuild.exe -a
```

- Create .wav audio files in the \tmp directory, using the listing file **vocal-script.txt**, and the voice “Cepstral William-8kHz”:

```
voicebuild.exe -f tmp -s ..\audio\en-us\vocal-script.txt -c "Cepstral William-8kHz"
```

- Re-create a set of .wav audio files, and change the volume and rate:

```
voicebuild -l 30 -r 5 -o -f tmp -s ..\audio\en-us\vocal-script.txt -c "Cepstral William-8kHz"
```

Monitoring Line Status

11

Description

Use the `d42util` program to examine a line when transferring or placing a call.

Usage

```
d42util.exe [<options>]
```

Table 11.1: d42util arguments

Argument	Description
-b	Specify the board name; the default device is <code>dxsxB1</code> .
-c	Specify the channel number; the default channel is 1.
-d	Dial the specified string, then exit. By default, the hook state is “off-hook” before dialing the string. If the string starts with <code>=</code> then the program dials the string without changing the hook state. If the string starts with <code>!</code> then the program sets the hook state to “on-hook” before dialing the string.
-i	Specify the refresh interval in milliseconds; the default is 1000.

Examples

- To monitor the `dxsxB1` line, channel 2 status, and display:

```
d42util -b dxsxB1 -c 2
```

- To make a call to extension 332:

```
d42util -d "<ESC>K1,332,, "
```

- To monitor with a specified interval of 10 milliseconds:

```
d42util -i 10
```

Call modes define how *Phone Password Manager* initiates telephone calls with users.

There are three different call modes in *Phone Password Manager*:

- Auto-answer mode
- Inbound mode
- Outbound mode

In order to specify a call mode, edit the `idtel.cfg` file, and change how the `ScriptName` is defined. The syntax for each `ScriptName` entry is as follows:

```
"<script-name>" = "LineNo|[,LineNo]| [BeginLineNo-EndLineNo]:[mode]"
```

The call mode can be:

- a|0 – auto-answer mode (default mode)
- i|1 – inbound (call can be answered selectively)
- o|2 – outbound

The line mode can also be retrieved and set using global variable 'lineMode', when configuring the `Psynch.ps1` call logic script.

12.0.1 Auto-answer mode

Auto-answer mode is the default mode. interactive voice response (IVR) calls are answered by default, call logic scripts run and audio plays according to the scripts.

12.0.2 Inbound mode

Inbound mode is similar to auto-answer mode, but instead of calls being answered by default, calls are only answered if the PSLANG function "setHookOff" is triggered. This allows calls to be answered selectively.

The "setHookOff" PSLANG function for this call mode is specified in the `psynch.ps1` script (p36).

In this example, the inbound call is answered if the callerID is "123":

```

if ( $callerID == "123" )
{
    setHookOff();
}

```

The callerID is one of several global variables that contain information about the current call. See [call logic global variables](#) for details on call logic global variables.

12.0.3 Outbound mode

Outbound mode allows the IVR system to make outbound calls, and is configured in the `psynch.ps1` script. When this is configured, the IVR system can forward the call to another phone number. Once the call is received, it proceeds according to the call logic scripts.

The phone number to which the calls are forwarded is specified in the `psynch.ps1` script (p36).

For example:

```

for( var $i = 0; $i < 30; $i++ )
{
    sleep( 1000 );
}
$ret = MakeCall( "9,403-2737373", 30, $errbuf );
log( "MakeCall returned: " + $ret + ", error: " + $errbuf );

```

In order for outbound mode to function, `idtel.cfg` must be modified as follows:

- Comment-out the "Registration" part of the script. For example, if your configuration does not include a proxy server, then the part to comment-out appears as follows:

```

Registration "" = {
    Server = 10.0.59.100
    Realm = hitachi-id.com //SIP only
    PhoneNumber = 168
    Password = "168"
}

```

- Set the value of `ipSignalPort`, which differs depending on the protocol you use:
 - H.323 ("ipProtocolName = 0") – set `ipSignalPort` to "1720"
 - SIP ("ipProtocolName = 1") – set `ipSignalPort` to "5060"
- Modify `idtel.cfg` to include the following:

```

ScriptNames "" = {
    "filename.ps1" = "2-2:o"
}

```

This loads "filename.ps1" from the `\<instance>\script\` directory of the VoIP instance, and uses line 2 with Outbound mode. The two numbers specify the range of lines, and o specifies outbound mode. By default, this call is included in `idtel.cfg`, but is commented-out.

Phone Password Manager can be configured to support call transfers on SIP and H323 protocols if it is configured to use Dialogic® PowerMedia Host Media Processing Software.

13.1 Pre-configuration

Before the function "TransferCall" can be added to *Phone Password Manager* to support call transfers, you must complete the following configuration:

1. Change the ipDTMFmode setting in `idtel.cfg` (p33):

```
ipDTMFmode = 6
```

This enables the DTMF key after a connection has been established.

2. Install SIP softphone on the *Phone Password Manager* server.
3. For testing purposes, install SIP softphone on another Windows machine as well. This machine receives the transferred call.

Next:

- [Configure TransferCall for SIP protocol \(p61\)](#).
- Or,
- [Configure TransferCall for H323 protocol \(p62\)](#).

13.2 Configuring TransferCall for SIP protocol

To configure *Phone Password Manager* to use the SIP protocol:

1. On the *Phone Password Manager* server, configure the SIP softphone to use the “SIP” protocol with “RFC2833” type.
2. Modify the `idtel.cfg` file for “SIP”:

```
ipBindAddress = Auto
ipSignalPort = 5060
ipProtocolName = 1 // sip = 1, h323 = 0
ipDTMFmode     = 6
```

3. Add the following script into `psynch.ps1` (p36):

```
if( $digits == "<telephone number/extension number>" )
{
    $ret = TransferCall( "<Telephone Password Manager server address>", $errbuf );
    log( "TransferCall returned: " + $ret + ", error: " + $errbuf );
    return 1;
}
```

To test the configuration:

1. Call the *Phone Password Manager* server using the SIP softphone.
2. When prompted for the user ID, type the telephone number or extension number that you want to transfer to, followed by the # sign. For example, 123#.
3. From the other machine with SIP softphone installed, pick up the line and listen.

13.3 Configuring TransferCall for H323 protocol

To configure *Phone Password Manager* to use the H323 protocol:

1. On the *Phone Password Manager* server, configure SIP softphone to use the “H323” protocol type.
2. Modify the `idtel.cfg` file for “H323”:

```
ipBindAddress = Auto
ipProtocolName = 0 // sip = 1, h323 = 0
ipDTMFmode    = 6
```

3. Add the following script into `psynch.ps1` script (p36):

```
if( $digits == "<telephone number|extension number>" )
{
    $ret = TransferCall( "TA:<Telephone Password Manager server address>", $errbuf
    );
    log( "TransferCall returned: " + $ret + ", error: " + $errbuf );
    return 1;
}
```

To test the configuration:

1. Call the *Phone Password Manager* server using the SIP softphone.
2. When prompted for the user ID, type the telephone number or extension number that you want to transfer to, followed by the # sign. For example, 123#.
3. Check the *Phone Password Manager* log file. The message “TransferCall API called” should be included, indicating that the API was triggered.

Bridge Transfer

14

Phone Password Manager can connect two phone lines to each other, which is known as a bridge transfer, hairpin transfer, or supervised transfer call. You can configure bridge transfers in two ways:

- Both the end-user and the help desk user call the *Phone Password Manager* system, and the system connects their calls together. For details, see the **bridge-demo1.ps1** sample script.
- Only the end-user has to call the *Phone Password Manager* system. The system then makes an outbound call to a help desk user. For details, see the **bridge-demo2.ps1** sample script.

Note: The scripts **bridge-demo1.ps1** and **bridge-demo2.ps1** are located in the `samples*` directory.

Phone Password Manager currently supports bridge transfers using both Dialogic® voice boards and Dialogic® PowerMedia Host Media Processing Software.

Once the system is able to complete bridge transfers, it works as follows:

- **Line A:**
 1. The end-user calls the *Phone Password Manager* system and authenticates.
 2. The end-user presses a key to request the help desk.
 3. The end-user is placed in queue.
 4. The end-user's status in the queue is updated until a help desk user becomes available.
 5. The two lines are bridged together.
- **Line B:**
 1. The help desk user either:
 - Calls the *Phone Password Manager* system and checks the queue for help requests.
 - Or,
 - Receives a call from the *Phone Password Manager* system.
 2. The system reports the end-user's information, and the help desk user either:
 - Accepts the request for help
 - Or,
 - Places the request back in the queue; the call maintains its position in the queue.
 3. If the help desk user accepts the request, then the two calls are bridged together.
 4. The system waits until the line is dropped.

Note: If required, the conversation can be recorded. See [Recording bridge transfers](#) for details.

14.1 Pre-configuration

Before the bridge transfer function can be added to *Phone Password Manager*, you must change the `ipDTMFmode` setting in `idtel.cfg` (p33):

```
ipDTMFmode = 6
```

This enables the DTMF key after a connection has been established.

If you are using a softphone, then you must:

1. Install the softphone on the *Phone Password Manager* server.
2. Install the softphone on another Windows machine as well. This machine receives the transferred call.

Next:

- [Configure bridge transfers \(p64\)](#)

14.2 Configuring bridge transfers

If you are configuring the `bridge-demo2.psl` script, then you must specify the help desk number as follows:

```
var $HelpDeskNumber = "SIP:<Server IP>";
```

Where:

- `<Server IP>` is the IP address for the outbound call to the help desk user. This address should connect to the softphone which can receive a call from *Phone Password Manager*.

Modify the `idtel.cfg` file as follows:

- **bridge-demo1.psl:**

```
ScriptName = "bridge-demo1.psl"
```

- **bridge-demo2.psl:**

```
ScriptName = "bridge-demo2.psl"
```

```
ScriptNames "" = {
    "bridge-demo2.psl" = "4-4:o"
}
```

Where:

4-4:o specifies to only use line four in outbound mode.

The two numbers specify the range of lines, and `o` specifies outbound mode.

14.3 Recording bridge transfers

You can record a call that has been transferred to the help desk by a bridge transfer. This function is controlled by the **recorder-demo.ps1** script, which is located in the `samples*` directory. The recorded call produces a WAV file.

This script is provided to help test the [bridge transfer function](#) (p63).

Both sides of a call (caller and automated voice) are recorded. However, it is possible to write a script to record the callers on two different lines. The “RecordFileEx” function has the ability to record two time slots simultaneously.

To configure call recording, you must modify **idtel.cfg** to assign an outbound mode channel for **recorder-demo.ps1** and to start the recording. It is recommended that you do so under guidance from Hitachi ID Systems staff.

14.4 Configuration notes

- Optionally, you can configure the bridge transfer script to play on-hold music or other programming while a user is waiting in queue. See **bridge-demo1.ps1** or **bridge-demo2.ps1** for details.

HTTPS Encryption

15

Phone Password Manager and *Bravura Pass* support HTTPS connections.

To configure *Phone Password Manager* and *Bravura Pass* to use HTTPS:

1. Install the "Certificate server" role on the *Bravura Pass* server.
2. Issue an SSL certificate and enable HTTPS on the *Bravura Pass* server.
3. Modify `\<instance>\idapiservice\Web-SSL.config` to replace `<instancename>` with your instance name in the `<appSettings>` section. For example:

```
<add key="instanceName" value="<instance>" />
```

Note: Be sure to use `Web-SSL.config`, because it contains additional HTTPS-specific information that is *not* found in `Web.config`.

4. Verify the HTTPS setup is correct by opening the following link in a web browser:
`https://<host server>/<Instance>/idapi`
If the setup is incorrect, you are notified by an exception or error message.
5. Install *Phone Password Manager* on another server.
6. Place a call using the interactive voice response (IVR) system to test *Phone Password Manager* functionality.

Viewing logs

16

You can use the *Hitachi ID Bravura Pass Manage reports* (RPT) module to view log details for the following *Phone Password Manager* operations:

- [unlock](#) (p67)
- [reset](#) (p68)

This allows you to see how many of these operations are being completed by *Phone Password Manager*.

16.1 Viewing the unlock logs

To view the logs for the unlock operation:

1. Navigate to the Event log report by clicking **Manage reports** → **Reports** → **System operation** → **Event log**.
2. Configure the following options:
 - Select the **Operation** code **ULCK Unlock account on target system**.
 - Specify the **Requester** by typing the name of the IDAPI caller that you defined in [Configuring an IDAPI caller](#). By default, this is `_API_USER_TPM`.
 - Enable the checkbox for **Show each detailed event**.
3. Click **Run**.

For full details on the **Event log** report, see [Event log](#) in the *Reports User Guide* ([reports.pdf](#)).

16.2 Viewing the reset logs

To view the logs for the reset operation:

1. Navigate to the Self service password changes report by clicking **Manage reports** → **Reports** → **System operation** → **Self service password changes**.
2. Set **Login method** to **Telephone Password Manager**.
3. Click **Run**.

For full details on the **Self service password changes** report, see [Self service password changes](#) in the *Reports User Guide*.

Read this chapter to learn how to troubleshoot your *Phone Password Manager* installation.

Note: The following sections contain information about how to troubleshoot common problems, usually related to improper installation, encountered during *Phone Password Manager* deployments.

Consult your vendor for additional troubleshooting information regarding your voice board, associated drivers, System Release software, or PBX.

17.1 Problems with the voice board

If you run into problems with your voice board:

- Check for IRQ conflicts.

One of the most common issues encountered when using voice boards is an IRQ conflict. Voice boards work best when configured with their own, high-priority, IRQ.

- Ensure that you are using the right type of phone line.

Only analog lines can be used with analog boards, and only digital lines can be used with digital boards. Plugging in the wrong type of line can damage the voice board.

- Check your PBX documentation to ensure that the board you purchased is compatible with your PBX.

- Ensure that the voice board is able to pick up calls.

You can use the Intel® voice demo to do this.

17.2 Problems with hangup events

If *Phone Password Manager* is having problems with hangup events on a digital network system, and you have correctly completed all configuration steps outlined in this manual, then there is probably a misconfiguration on the Dialogic card. This problem should not happen on digital network systems. If it is happening, contact the support department of your Dialogic hardware supply company, or purchase support services from a third-party company.

If *Phone Password Manager* is having problems with hangup events, and the Dialogic card is connected with analog CO lines or analog PBX lines:

- Enable the “circuit reverse” or “battery reverse” features on those lines. This can offer more reliable disconnect supervision.
- If “circuit reverse” is not available, use the PBXpert to detect and enable tones for the analog Dialogic card. For further details, see your Dialogic manual.

17.3 Welcome message does not play

If you hear dead air instead of the welcome message, then upgrade to the latest Dialogic System Release Software Updates. This issue can also be caused by improper audio file configuration.

17.4 Some or all audio files are not playing

If no sound is played to users who connect, or there are certain menu options for which the sound files are not being played, it is possible that your audio file configuration is incorrect. *Phone Password Manager* organizes audio files in the following directory structure:

- `<Instance>\audio\`
 - en-us
 - * a.wav
 - * b.wav
 - * (etc)
 - `<Language code>`
 - (Other languages)

If this directory structure is disturbed, or any of the audio files themselves are missing, then the system will not be able to locate those files for playback to the user. This error can be easily diagnosed by reviewing the system logs, which will include a message such as:

```
Warning: Cannot open C:\Program Files (x86)\Hitachi ID\Telephone Password Manager\<instance>\audio\en-us\<filename>.wav, errno: 2
```

This error indicates that *Phone Password Manager* was unable to locate the audio files in their usual directory.

17.4.1 Asterisk® audio files

When using an Asterisk® server, *Phone Password Manager* needs to upload the locally-stored audio files onto the Asterisk® server. *Phone Password Manager* will only initiate this file synchronization if it cannot find the following directory on the Asterisk® server:

```
\var\lib\asterisk\sounds\HiTPM\
```

To force *Phone Password Manager* to update the Asterisk® server's audio files: delete the directory listed above, restart the *Phone Password Manager* service, and place a call to the IVR.

17.5 The *Phone Password Manager* service fails to start

In the system services menu, you should see:

- Dialogic® Boardserver
- Dialogic® SS7 Service
- Dialogic® System Service ★
- Hitachi ID logging Service ★
- Hitachi ID Telephone Password Manager Module Service ★, or Hitachi ID VoIP Telephony Service ★ (Dialogic only)

Note: Services marked with a ★ *must* be started in order for *Phone Password Manager* to operate properly. Dialogic® services only appear on a system using a Dialogic® IVR backend. The names of the Dialogic® services may vary.

Note: In *Phone Password Manager* version 9.0+, the VoIP service has been merged into the *Phone Password Manager* module service.

If the *Phone Password Manager* Service fails to start:

- Ensure that the Dialogic® System Service, and Dialogic® Boardserver service, are running and configured to start automatically.
You can do this using the Windows Service Control Manager (SCM), or the Dialogic® product Configuration.
- Ensure that **pspushpass.dll** is installed and can be found in the system PATH.

- Some Dialogic® services are dependant on other Dialogic® services and will not restart automatically after a reboot of the server. After a reboot, make a test call into your interactive voice response (IVR) server, and manually restart Dialogic® services if required.
- The *Phone Password Manager* Service is dependant on Dialogic® system services, and also needs to be manually restarted after a reboot. It may take several seconds before the service is ready to be started, so ensure that you refresh the list of services to confirm that this service is running.

17.6 Phone Password Manager cannot return requests

If *Phone Password Manager* cannot return requests properly due to slow network speeds, then you can modify the “SoapTimeout” registry key:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Hitachi ID\<product>\<instance>\Idapi\
```

Modify the “SoapTimeout” registry key by increasing the value. The default setting for this value is 60000 milliseconds, which is one minute.

17.7 Phone Password Manager cannot connect to the softphone system

If *Phone Password Manager* cannot connect to the softphone system, try switching audio codecs in `idtel.cfg`. The codec in *Phone Password Manager* must match the supported codecs of your softphone system. See [Editing the configuration file](#) for details on `idtel.cfg`.

17.8 Phone Password Manager fails unexpectedly

If *Phone Password Manager* fails unexpectedly, it is possible the Dialogic license is expired or invalid. You may see an error message like the following:

```
2013-01-15 07:54:59.532.2056 - [] idvoip.exe [3292,3180] Warning: gc_Start,
GC ErrorValue: 0x8c - The start procedure of a call control library failed,
CCLibID: 0 - GLOBALCALL, CC ErrorValue: 0x8c - The start procedure of a
call control library failed
```

```
2013-01-15 07:54:59.532.2093 - [] idvoip.exe [3292,3180] Debug: Failed to
initialize GlobalCall Libraries
```

```
2013-01-15 07:54:59.532.3090 - [] idvoip.exe [3292,3180] Error: Failed to
InitGC, service terminated
```

To resolve this, update the Dialogic license, then restart the *Phone Password Manager* service. See the Dialogic documentation for details on updating a license.

17.9 SoX version mismatched

Phone Password Manager is fully functional with newer versions of the SoX utility, however *Phone Password Manager* expects SoX versions equal to or earlier than those shipped with Asterisk® when checking for the existence of a SoX installation. This version mismatch can impede the installation and function of *Phone Password Manager* with Asterisk.

To resolve this issue, log into the Asterisk® server as root, or escalate to root. Execute the following commands in the Unix terminal, in order:

1. `cp 'which sox' 'which sox'2`
2. `echo '#!/bin/bash' > 'which sox'`
3. `echo '["$1" = "--version"] && echo "sox: Version 0.0.0" || sox2 $*' >> 'which
→sox'`

This change will reconfigure how SoX responds when asked to display its version, allowing *Phone Password Manager* to install with newer versions of SoX.

Index

A

- additional languages, [39](#)
- API
 - Password Manager remote API, [22](#)
 - setup for version 8.x, [22](#)
- api call sequence, [32](#)
- architecture, [5](#)
- assigning phone lines, [15](#)
- Asterisk software, [18](#)
- audio files
 - call-logic, [39](#)
 - managing, [37](#)
 - questions, [37](#)
 - target groups, [38](#)
 - targets, [38](#)
- auto-answer mode, [58](#)

B

- Bridge transfer, [63](#)
- bridge transfer
 - testing, [65](#)

C

- call logic
 - writing scripts, [36](#)
- call modes, [58](#)
 - auto-answer mode, [58](#)
 - inbound mode, [58](#)
 - outbound mode, [59](#)
- call recording, [65](#)
- checking logs, [67](#)
- checklist
 - all setup tasks, [9](#)
- configuration
 - IVR server, [31](#)
- Configuring user authentication, [43](#)
- Connecting two calls, [63](#)
- conventions used in this document, [3](#)

- custom authentication questions, [37](#)
- custom call logic, [39](#)
- custom questions
 - formatting, [46](#)
- custom target groups, [38](#)
- custom targets, [38](#)

D

- `d42util`, [57](#)
- Dialogic
 - Host Media Processing software, [17](#)
 - voice boards, [12](#)
- documentation
 - conventions, [3](#)
 - feedback, [4](#)

E

- encryption
 - HTTPS, [66](#)
- event actions (exit traps), [32](#)
- event log
 - checking reset and unlock, [67](#)

G

- Generate voice print enrollment PIN, [11](#), [48](#), [50](#)

H

- HDD Encryption Audio files, [40](#)
- Host Media Processing software, [17](#)
- HTTPS encryption, [66](#)

I

- `idmsetup.inf`, [29](#)
- `idtel.cfg`, [33](#), [53](#)
- inbound mode, [58](#)
- integration mechanisms, [31](#)
- IVR server

configuration, 31
requirements, 12, 17

K

keypads
 mapping user IDs, 40

L

languages
 adding additional, 39
LoadInstanceCFG, 41
log files
 location, 29
logs
 reset and unlock operations, 67
loop disconnect supervision, 33
 TSF files, 35

M

Manage reports, 67, 68
managing alternate instances, 30
managing audio files, 37
mapping user IDs
 telephone keypads, 40
multiple instances, 41

O

outbound mode, 59
overview of required tasks, 9

P

Password Manager Remote API, 5
phone lines
 assigning, 15
playback volume
 adjust level, 33
psadmin, 29
psdonechk, 45
psynch.psl, 36, 58, 59, 61, 62
psynch.voiceprint.psl, 37

Q

question sets
 setup, 43

R

recording calls, 65
removing voice print enrollment data, 51
required setup tasks, 9

S

setup, 30
 managing alternate instances, 30
Speech recognition, 53
Speech to text, 53
SSL connection, 66
styles used in this document, 3
Supervised transfer calls, 63
system structure, 5

T

technical support, 4
tpm.cfg, 33
transferring calls, 60
troubleshooting, 69

U

Uninstalling Telephone Password Manager, 30
user authentication, 43

V

voice boards, 12
 calculating number and size, 13
 installing hardware, 16
 selecting, 15
voicebuild, 55
voice print enrollment
 configuring, 48
 removing data, 51
 testing, 50
volume
 playback level, 33
vpcmd, 50
vputil, 51

W

web services, 31
writing call logic scripts, 36
