Provisioning CounterPath Bria Professional

OEM EditionRetail Edition

CounterPath Corporation.
Suite 300, One Bentall Centre
505 Burrard Street Box 95
Vancouver BC V7X 1M3
Tel: 1.604.320.3344
sales@counterpath.com www.counterpath.com

This manual corresponds to version 2.5 of Bria Professional.

Contents

1 About Provisioning 3
1.1 Provisioning Functions 3
1.2 What Provisioning Does: Writing to Settings 3
1.3 The Mechanism of Remote Provisioning 5
2 Login and Configuring 9
2.1 Credentials Required 9
2.2 Specifying the Login Server 10
2.3 Login Server Fallback Behavior 10
2.4 The Login Process 12
3 Updates and Upgrades 15
3.1 General Setup 15
3.2 Remote Update 17
3.3 Remote Upgrade 18
A Script Samples 21
B Macros 22

About this Manual

This manual describes the *mechanism* of remote login/provisioning. It describes how to set up a server (or servers) for the remote login and optionally the remote provisioning, remote update and remote upgrade features of Bria Professional

- Remote login controls access to the application; the softphone will not start until the user has logged in.
- Remote provisioning lets you configure the softphone remotely.
- Remote update lets you change the configuration of a given deployment of Bria Professionalat runtime (outside of the login process).
- Remote upgrade lets you deploy upgrades of the software remotely.

This manual is intended for:

- System administrators who have purchased Bria Professional from the CounterPath website and are deploying Bria Professional in their enterprise using remote logging and (optionally) remote provisioning. (See Bria Professional Administrator Guide for other ways of deploying.)
- Service providers who have purchased Bria Professional from CounterPath Sales, without further customization or engineering changes.

This manual is intended to be read in conjunction with:

- "Configuring Bria Professional *Retail Edition*", which describes the features that can be configured through remote provisioning.

1 About Provisioning

1.1 Provisioning Functions

Provisioning of Bria Professional includes the following features:

- Controlling access to the VoIP service through a remote login. See page 9.
- The ability to provide a license key remotely. See page 9.
- Updating the Bria Professional configuration (changing the factory defaults). Bria Professional can be configured differently for each user. This feature is optional. See page 15.
- Providing upgrades to the executable by making new versions of Bria Professional available to each Bria Professional installation to download. This feature is optional. See page 15.

1.2 What Provisioning Does: Writing to Settings

Each provisioning function involves writing to settings stored on Bria Professional computer. These settings control the behavior of various features of Bria Professional. For example, a successful login request will result in the creation of new settings representing the account. A remote update may result in changing the value of existing settings.

For detailed information on settings and the features they control, see “Configuring Bria Professional *Retail Edition*.

1.2.1 Provisioned Settings Overwrite GUI Settings

Settings are assigned values in several ways:

- A setting has a default “factory” value.
- Some settings can be changed by the user on the GUI.
- Remote provisioning lets you can change the value of any setting.

At startup, the factory values are loaded, then the user overrides are loaded (overwriting factory values), and finally values that you send through the provisioning response are loaded (overwriting factory or user values). At shutdown, the current user overrides and provisioning overrides are persisted to the user file.

Keep in mind that provisioned settings override user settings. A user may complain that they change a value on the GUI but each time they restart Bria Professional, their changes are lost: you are probably overwriting their value when you provision.

The Bria Professional Settings reference documentation (a Microsoft® Excel® document) includes a column that identifies settings that are represented on the softphone GUI.

1.2.2 Syntax of Settings

Each setting has a fully qualified name: <domain>:<section>:<setting>

For example, proxies:proxy0:register.

The syntax for setting values via provisioning is:

<domain>:<section>:<setting>=<“value”>

For example, proxies:proxy0:domain=“domainA.com”

- The value of the variable must appear in double quotes.
- Always a string. True is represented by “true” or “1”. False is represented by “false” or “0”.
- The Bria Professional process that interprets the settings ignores the case of the value (uppercase or lowercase), except for literals such as display names.

1.3 The Mechanism of Remote Provisioning

Each remote provisioning service involves an exchange between the login server and an individual Bria Professional client. The exchange is performed over HTTP or HTTPS.

1.3.1 Servers

You must deploy servers to handle the provisioning requests:

- The “login server”: a server to handle login requests, if you decide to implement login. This server is simply a web server that, at a minimum, can serve one plaintext or XML file.
- The “update server”: a server to handle remote update.
- The “upgrade executable server”: a server to handle remote upgrades of the Bria Professional application.

These server roles may in fact all be deployed on the same physical server: that is your decision.

You must set these servers in Bria Professional.

- The login server is set either through DHCP or through a manual process, as described in *.
- You will set the update server and upgrade executable server (if they are being used) in the provisioning response that you set the first time the user logs in.

1.3.2 Bria Professional-to-Server Exchange

The exchange between Bria Professional and the appropriate server involves the following:

- When the appropriate trigger occurs, Bria Professional sends an HTTP or HTTPS request to the server. For login, the trigger is the user pressing OK on the Bria Professional login dialog. For remote upgrade, the trigger is startup of the softphone.
- The server responds.
- Bria Professional reads the response and takes the appropriate action: starts the softphone and registers with the SIP proxy, or finds and installs the upgrade.

Use of Scripts and Macros

You may want to run an appropriate script on the given web server, to provide the information required by Bria Professional. To run a script, include it in the URL for that server.

Running scripts usually requires information about the user’s deployment. The URL for the appropriate server can include macros. When Bria Professional contacts the server, it replaces the macros with the real data and includes this information in the HTTPS request.

Your script must understand the names assigned to the macros.

For example a URL of

```
https://mycustomloginserver.com/login.php?platform=$platform$&lic=$license$
```

might become this POST used to log in the user:

```
https://mycustomloginserver.com/login.php
---------------------------------------------------------------------------------------
Username=21187
Password=rosebud
platform=win32
lic=d3874ihfd8t23975v1iu5182ruity3iusapor236u545uye0r9qwjj
```

Note that “Username” and “Password” (with initial capitals) are always sent in a login POST; the URL does not have to include macros for this data.

See “Script Samples” on page 21 for samples of some of the scripts that are mentioned in this manual.

See “Macros” on page 22 for a list of macros that Bria Professional supports.

1.3.3 Communication Mechanism

All communications between Bria Professional and the login server are performed over HTTP or HTTPS, as follows:

- Custom login uses POST.
- Remote update and remote upgrade use GET.

The remote provisioning mechanism does not support redirect.

If using HTTPS, you need a trusted certificate (not self signed). Bria Professional will only accept certificates whose authenticity can be verified through the trust chain.

1.3.4 Data Format

All the data included in the GET or POST response is in a specific format. This format is similar to that of Microsoft® Windows®.ini files.

The information is organized into three portions, which must appear in this order:

- [DATA]
- [SETTINGS]
- [##MEMORY##]

Example

```
[DATA]
Success=1
[SETTINGS]
proxies:proxy0:display_name="kokila"
proxies:proxy0:enabled="1"
proxies:proxy0:username="kpereira"
proxies:proxy0:password="dfher43d89dhferuieo98375uy8”
proxies:proxy0:domain="domainA.com"
```

[DATA]

This section contains the response to requests:

Success=<value>, a boolean. This data is required.

Failure=<message>, which is optional if the success is 0. For login, the string you enter here will be displayed in the Login dialog.

[SETTINGS]

This section contains settings to be written to persistent memory. The values will be used immediately.

At shutdown, these settings will be written to the local settings file on the Bria Professional computer.

[##MEMORY##]

This section contains settings to be written to non-persistent memory. The values will be used immediately, but only for the current session.

At shutdown, these settings will not be written to the local settings file.

CRLF

The response must end with a CRLF. If this is missing, the last line of the response is ignored.

Handling and Encryption of Passwords

All “password” settings in any domain/section are handled as follows:

- Bria Professional does not interpret passwords in any way, so the value the login server passes to Bria Professional can be encrypted.
- Bria Professional encrypts the value before storing it, regardless of whether or not it is already encrypted. When a stored value is read in order to pass it to the login server, it is first decrypted.
- When a password that the user has been entered into a dialog is then passed to the login server, Bria Professional does not encrypt the value.

1.3.5 Example of an Implementation

The hardware requirements of the login server depend on what the server will do. If it will have a complicated backend database and processing involved in order to retrieve the settings that are to be provisioned, then the server should be of higher processing capabilities. Regardless, the login server is simply a web server and it only needs to serve one file for provisioning; this file is in plaintext or XML format.

The login server could be a Linux® machine with an Apache™ web server or a Microsoft® Windows® machine with an IIS web server.

For their internal deployment, CounterPath uses Debian® Linux with Apache version 2. The login server is a Pentium® 4 with 3GHz processor. This server scales to thousands of requests per second. We use the internal database of the SIP proxy (this can be a MySQL® database) which contains all usernames and passwords. The provisioning response is constructed based on login information retrieved from Bria Professional via the login PHP script.

2 Login and Configuring

2.1 Credentials Required

2.1.1 Types of Credentials

Login Credentials

Login refers to the process of signing into the VoIP service. The Bria Professional user must enter login credentials – user name and password – in order to access to Bria Professional.

Login credentials cannot be changed through provisioning.

SIP Account Credentials

Once the user has logged in, the SIP account credentials allow the user to register for your VoIP service; they are known to your SIP registrar. These credentials are user name, password, and the optional authorization user name.

SIP credentials are always required. They can be changed through provisioning.

XMPP Account Credentials

Once the user has logged in, the XMPP credentials allow the user to access the XMPP service; they are known to the XMPP server. These credentials are user name and password.

XMPP credentials are required only if you support XMPP accounts. They can be changed through provisioning.

2.1.2 Providing Credentials

When setting up a new user, give the user their login credentials, outside of Bria Professional. You do not give the user the account credentials; instead, these credentials will be sent down through provisioning.

- The account user name and login user name can be identical or different.
  - The login user name is meaningful to the user (for example, their own name).
  - The account user name follows the syntax for your accounts – it may be a number or words.
- The account password and the login password are typically different for security reasons.
  - The login password should not be encrypted, because the user will enter it manually.
  - The account password does not have to be human-readable.
- The authorization user name (one of the account credentials but not a login credential) is optional.
  - An authorization user name is useful, for example, if you allow usernames that are short and therefore easy to guess. The authorization user name is used in place of the user name to register the account with the proxy. It provides an added layer of security.

- The authorization user name is used only to register the account. It does not replace the account user name in identifying the user to the outside world.
- If you use an authorization name, make sure it is different from the user name!

2.2 Specifying the Login Server

You can identify the login server in one of these ways:

- Bria Professional discovers login server via DHCP.
- Users enter login server manually. You might choose to use this scenario, for example, in the following situations:
  - You do not want to set up DHCP in your enterprise.
  - You have set up DHCP but one of your users is using Bria Professional for the first time on a computer that is not on the enterprise LAN; for example, the user is traveling and using a new laptop.

Login with DHCP Discovery

Your DHCP server must be set up so that option 120 specifies the URL of the login server.

When Bria Professional starts, the standard Login dialog appears. When the user presses OK, Bria Professional attempts to find a login server using DHCP. If a server URL is found, then Bria Professional attempts to contact that server. If the server is contacted, then Bria Professional attempts log in to that server, as described in “The Login Process” on page 12.

If the login is successful, then account credentials and other data are sent through provisioning. This data, as well as the URL of the discovered login server, is stored locally on the Bria Professional computer. The next time the user logs in, it first goes through DHCP discovery. If discovery fails (for example, the user is temporarily not on the enterprise LAN), then Bria Professional uses the locally stored server URL.

Login with Manually Entered Server

In this case, you must provide users with the URL of the provisioning server. The URL should include any scripts and macros you are using.

When Bria Professional starts, the standard Login dialog appears. When the user presses OK, Bria Professional attempts to find a login server using DHCP. If a server URL is not found, then the “Skip Login or Enter Server” version of the Login dialog appears.

The user should enter the server URL and sign in. Bria Professional attempts log in to that server, as described in “The Login Process” on page 12.

If the login is successful, then account credentials and other data are sent through provisioning. This data, along with the manually entered URL for the login server, is stored locally on the Bria Professional computer. The next time the user logs in, it goes through DHCP discovery and fails (as usual). Bria Professional then uses the locally stored server URL.

2.3 Login Server Fallback Behavior

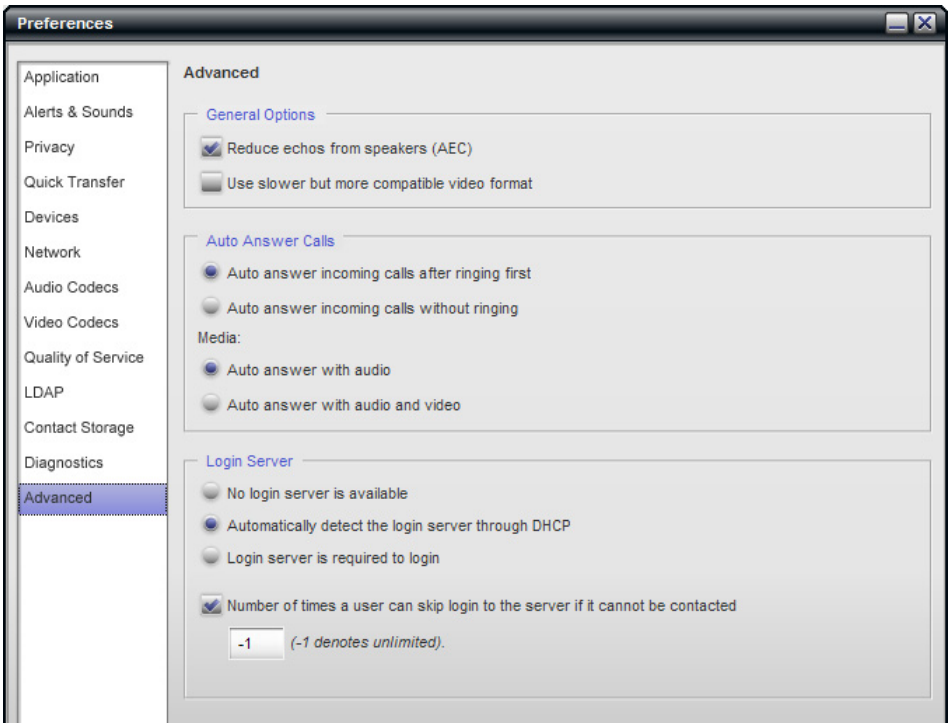
Regardless of whether DHCP is being used or not, Bria Professional allows login to be skipped when the login server cannot be contacted (for example, it is temporarily offline).

In these situations, Bria Professional displays the “Skip Login or Enter Server” version of the Login dialog. The user should press Skip to start Bria Professional without logging in. The configuration settings saved the last time the user used Bria Professional are used for this session.

Changing the Fallback Behavior

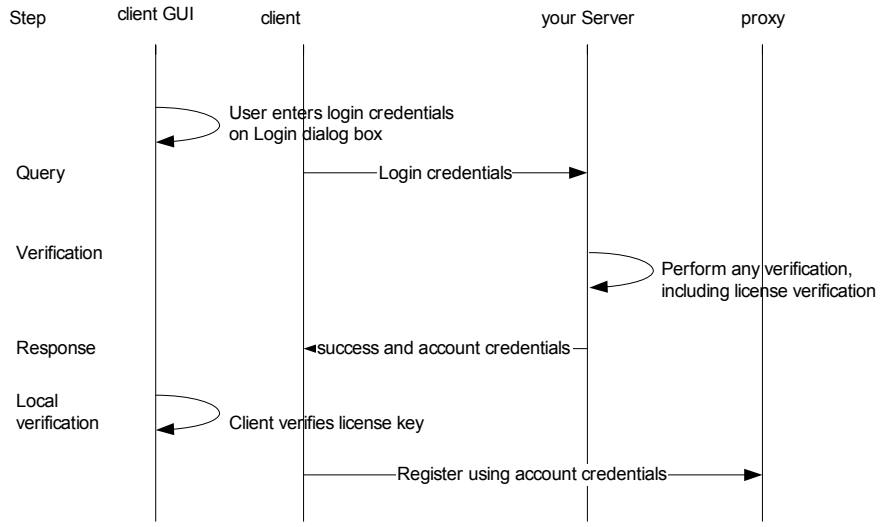
Once the user has started Bria Professional, they can display the Preferences > Advanced page and change the login behavior.

You should give the users clear instructions on the dangers of changing these settings. These settings are really shown on the GUI to allow you, the administrator, to bootstrap Bria Professional when provisioning is not yet set up, as described in the Bria Professional Administrator Guide.



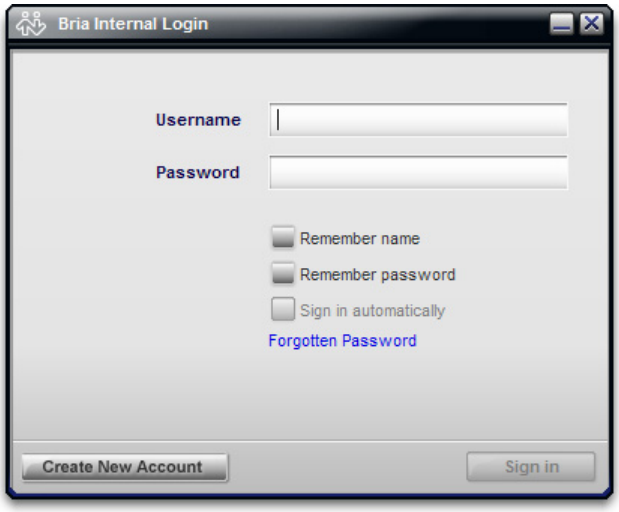
2.4 The Login Process

The login server must be set up to handle the following procedure.



Login Procedure Is Invoked

The Login dialog box is displayed. The user enters the required information and presses Sign in.



Query Step

Bria Professional sends the data from the Login dialog. The data is encoded application/x-www-form-urlencoded.

The data is sent to the login server (the server specified in feature:custom_login:server) in an HTTP POST. The value will be blank if the branded Login dialog box does not include the corresponding field.

For example a URL of

```
https://mycustomloginserver.com/login.php?platform=$platform$&lic=$license$
```

might become this POST used to log in the user:

```
https://mycustomloginserver.com/login.php
---------------------------------------------------------------------------------------
Username=21187
Password=rosebud
platform=win32
lic=d3874ihfd8t23975v1iu5182ruity3iusapor236u545uye0r9qwjj
```

where:

- “Username” and “Password” (with initial capitals) are always sent in a login POST; the URL does not have to include macros for this data.
- platform and lic are macros used by the login script; see “Use of Scripts and Macros” on page 5.

License Key Management

You should set up Bria Professional to include the license key in the data sent to the login server. There are two ways to send this data:

- Include one of the license macros in the URL. See page 22.
- Set the setting feature:custom_login:always_include_license_in_post to true.

Verification Step

Your login server should perform any suitable verification on the sent data, according to your business rules.

Typically, this verification will include one of the following checks on the license key:

- For a new deployment (no license key was included in the query), determine that the query is valid, and if so, obtain a license key to send back to the user.
- For an existing deployment (the license key was included in the query), do nothing.

Response Step: Failure

If there is a problem with any of the data, the login server should return failure data in the following format:

```
[DATA]
Success=0
Failure="<message> "
<CRLF>
```

Response Step: Pass

If the server can handle the request, it should return a success message and the account credentials. It can also return other settings that can be specified only at login.

Example with a license key passed in the SETTINGS section:

```
[DATA]
Success=1
[SETTINGS]
proxies:proxy0:user_name="KPereira"
system:license:key=”e48jey45379ryeioo8a7e934q8dhfudufoladskiuwb”
[##MEMORY##]
proxies:proxy0:password="rosebud"
<CRLF>
```

where:

- success: this line is required.
- Settings: the username will be saved at shutdown.
- ##Memory##: the password will not be saved at shutdown.
- The response must end with a CRLF.

Local Verification

Bria Professional next takes one of these actions, depending on the response received from the server:

- If the response was a failure, then the Login dialog apppears again. The process goes back to “Login Procedure Is Invoked” on page 12.
- If the response was a success, then Bria Professional verifies that the license key is valid. The license key is whatever is currently stored locally: it could be the license key that was sent down in the reponse, or it could be the license key that was already stored locally at startup, or it could be empty.
  - If the key is valid, Bria Professional starts.
  - If the key is not valid or is empty, then the Enter License Key dialog appears. When the user enters the license key (obtained outside of Bria Professional, for example in an email sent to all new customers), the Bria Professional verifies that the entered license key is valid. If the key is valid, Bria Professional starts.

3 Updates and Upgrades

Remote Updates

You can configure Bria Professional to check with the update server at specified intervals for changes to the user's settings.

Remote Upgrades

There are two ways to support upgrades to Bria Professional.

Use CounterPath Upgrade Server

The default behavior is to obtain upgrades from the CounterPath upgrade server. Whenever CounterPath puts a new executable on its upgrade server, your users' Bria Professional will automatically download the upgrade and prompt the user to install it.

No work on your part is required for this option. Bria Professional is already configured to do this. You do not have to set up any servers.

Set up your Own Upgrade Server

You can set up an upgrade server and deploy upgrades yourself. One of the advantages of this option is that you can test the new executable on a few deployments before rolling it out to all your users. See page 18.

3.1 General Setup

Remote update and remote upgrade are controlled by Bria Professional settings. To change the default values to values suitable for your deployment, change them through remote provisioning, the first time the user logs in. After that, change them as necessary through remote provisioning or remote update.

Domain:Section	Setting	Comment
feature:auto_update	code_server_url	The web server for remote upgrades of the executable. Default is empty.
feature:auto_update	config_server_url	The web server for remote update. Default is empty.
feature:auto_update	block_timer_t3_s	See below for a description. Default is 10 seconds. Typically, leave the default.
feature:auto_update	deffer_timer_t2_s	See below for a description. Default is 60 seconds. Typically, leave the default.
feature:auto_update	update_check_initial_t1_s	See below for a description. Default is 20 seconds. Typically, leave the default.

feature:auto_update	update_check_t1_s	See below for a description. Default is 86400 seconds (24 hours). Typically, leave the default.
feature:auto_update	timer_factor	See below for a description. Default is 1.00

Timer Settings

Remote upgrades and remote updates rely on four timers in the user's settings. The timers control how frequently Bria Professional contacts the update and upgrade executable servers.

All values are in seconds. You can use the timer_factor setting to convert the values on your side into seconds.

- update_check_initial_t1_s.
- update_check_t1_s.
- deffer_timer_t2_s.
- block_timer_t3_s.

Automatic checks for remote upgrades and automatic checks for remote updates are performed at the same point: when the user starts Bria Professional. The interaction of the timers occurs as follows:

1. Bria Professional starts.
2. The timer update_check_initial_t1_s starts.
3. When update_check_initial_t1_s expires, Bria Professional checks its state of business (whether or not the user is busy with a call or instant message session).
    - If Bria Professional is busy, deffer_timer_t2_s starts. When this timer expires, Bria Professional checks its business again. deffer_timer_t2_s continues restarting and expiring until Bria Professional is no longer busy (when the user hangs up from an active call).
    - If Bria Professional is not busy, it contacts the servers.
4. The timer block_timer_t3_s is set when a check is initiated at Bria Professional. Another check is not allowed as long as block_timer_t3_s is still active. This timer ensures that provisioning checks are not performed too often, and is especially useful for protecting against potential hacker requests (which may arrive with frequency). The timer block_timer_t3_s is typically shorter in duration than update_check_t1_s.
5. After the first check, the cycle starts over at step 1 using the timer update_check_t1_s (not update_check_initial_t1_s).

Changing Timer Settings

New timer settings will take effect as follows:

- If the timer is not running when the server sends new settings (and they are saved on the Bria Professional computer), then the setting take effect immediately. The next time the timer is loaded, the new setting will be used.
- If the timer is running, the new setting takes effect after the timer expires and is reloaded. This means if update_check_t1_s still has 23 hours to go, it will be changed only after 23 hours. However, if the session is restarted, the new setting will take effect.

3.2 Remote Update

3.2.1 Setting Up

- Set up Bria Professional as described on page 15.
- Set up the update server to handle the procedure described below.

3.2.2 How Remote Update Is Performed

Assuming that the timers are not all set to zero, this procedure runs “in the background” for as long as Bria Professional is running.

1. When triggered by the timer, Bria Professional checks for remote updates by sending a GET to the update server

   For example, the value of feature:auto_update:config_server_url might be:

   https://myupdatesettingsserver.com?language=$language$&build=$build$&name=$loginame$

   This URL could result in a GET to your webserver of:

   myupdatesettingsserver.com?language=EN&build=16835&name=kpereira

2. The update server must response with the following:

   [DATA]
   Success=0
   <CRLF>

   or

   [DATA]
   Success=1
   [SETTINGS]
   feature:auto_update:update_check_t1_s=”3600”
   <CRLF>

   where:

   - success: 1=true (there are updates) or 0=false (there are no updates).
   - The [SETTINGS] section contains the changed settings. See “Data Format” on page 6 for details.
   - The response must end with a CRLF.

3.3 Remote Upgrade

3.3.1 Setting Up

- Set up Bria Professional as described on page 15.
- Set up an upgrade server as follows:
  - You can use a script to include logic that determines a given deployment needs an upgrade. See below for an example. Obtain the sample upgrade script from CounterPath and modify it to suit your needs.

    Or you can skip the script and manually set up your upgrade server to simply provide a success response when an upgrade is available and a failure response at other times.
  - If you are using scripts, set the URL for the upgrade server to include the script and any macros (for example, the language and the build macros).
  - When you want to deploy an upgrade, place it on the “upgrade location”.

3.3.2 How Remote Upgrade Is Performed

Assuming that the timers are not all set to zero, this procedure runs “in the background” for as long as Bria Professional is running.

Bria Professional Sends a GET

When triggered by the timer, Bria Professional checks for available upgrades by sending a GET to the upgrade executable server.

- For example, if you are using scripts, the value of feature:auto_update:code_server_url might be:

  https://executablepgradeserver.com/exe_upgrade.php?build=$build$&language=$language&name=$loginame$

  This URL could result in a GET to your webserver of:

  https://executablepgradeserver.com/exe_upgrade.php?build=38740&language=USEnglish&name=kpereira

- Or if you are not using scripts, the value is simply the URL of the upgrade server:

  https://executablepgradeserver.com

Server Response

The upgrade executable server must respond with the following:

```
[DATA]
Success=0
<CRLF>
```

or

```
[DATA]
Success=1
Mandatory=1
version=60000
url=https://executableupgradeserver.com/newversion.exe
<CRLF>
```

where:

- `Success`: 1=true (there is an upgrade) or 0=false (there is no upgrade).
- `Mandatory`: 1=true. This response is optional; the deafult is "0". Bria Professional handles the upgrade differently depending on this response; see below.
- `version`: identifies a build stamp set by Bria Professional during build time. Bria Professional uses this version to determine whether to prompt the user to install the upgrade; see step .
- `url`: the absolute path to the installer software for the new version.
- The response must end with a CRLF.

The response **cannot** include a [SETTINGS] section. In other words, none of the user's current settings can be changed via this response.

If no upgrades are found, Bria Professional will recheck periodically for available upgrades. See "Timer Settings" on page 16 for details.

Handling of the Upgrade

If an upgrade is available, Bria Professional compares the build number of the application on the user's computer to the build number specified in the response (60000 in the above example).

- If the response has the same number, Bria Professional does not prompt the user to download
- If the response has a *different* number, Bria Professional prompts the user to download the upgrade.
  - If the user initiates the download, Bria Professional will download the installer and save it to the local Bria Professional program folder. Bria Professional will also prompt the user to exit in order to install the new version. The user can install immediately or postpone installation.
  - If the user declines the upgrade and the upgrade is optional, Bria Professional will enter its timing cycle and display the download prompt again at the appropriate time. See "Timer Settings" on page 16.
  - If the user declines a mandatory upgrade, Bria Professional shuts down
  - If the user declines with "do not ask me again" (possible only with an optional upgrade), Bria Professional will not check again for upgrades during the session.

"Install Later" Handling

If the user declines to install the downloaded version, then the next time Bria Professional is started the user will be prompted to install the newer version.One of the following will occur:

- If the user initiates the installation, Bria Professional will install the new (local) version.
- If the user declines, Bria Professional will start the original version and will enter its timing cycle, displaying the download prompt again at the appropriate time. See "Timer Settings" on page 16.
- If the user declines with "do not ask me again", Bria Professional will start the original version and will not prompt to install again during the session.

Bria Professional starts the version installed most recently. The automatic check scenario will be initiated as described in the previous section. The downloaded installer will not be deleted, to enable manual rollback, if required.

A Script Samples

Contact CounterPath to obtain sample scripts.

These sample scripts, written in PHP, are intended to illustrate a possible implementation. They are not intended to be used without modification. You should write scripts suitable to your environment, in an appropriate scripting language.

login.php

Custom login script. Bria Professional passes in the username and password. After verification, if the login credential is correct, the server will write the proper settings into the response and send it back to Bria Professional.

See “Use of Scripts and Macros” on page 5 for an example that uses this script.

exe_upgrade.php

Bria Professional passes in the buildstamp. You may want to revise the script to also pass in the platform. It returns success is true or false, plus the URL where the upgrade of the Bria Professional executable is located.

See “Remote Upgrade” on page 18.

B Macros

Macro	Description	Value
$acc_passwdn$	where n is an account. The password for the specified SIP account (for deployments that support more than one SIP account). Stored as a setting.	
acc_usern	where n is an account. The username for the specified SIP account (for deployments that support more than one SIP account). Stored as a setting.	
$build$	The last five digits of the executable name. For example, if the executable file is BriaProfessional_Win32_2054p_12345, the build is “12345”. This number also appears in the About box (Help > About).	For example, 12345
$computerid$	Unique ID for this computer	
$computername$	From the operating system	
$hashlicense$	A hash of the license key. This allows the license key to be sent in a secure way	
IP	The IP address of this computer	
$language$	The language of the installed application.	USEnglish, French, German, Italian, Portuguese-Brazil, Spanish,
$license$	The license key.	
$loginname$	The login username. This is the username the user enters in the Login dialog and is not necessarily the same as the SIP username. See page 9.	
$loginpassword$	The login password. This is the password the user enters in the Login dialog and is not necessarily the same as the SIP password. See page 9.	
MAC	The MAC address of the machine running Bria Professional.	
$platform$	The operating system platform.	win32, mac, UNDEFINED.
$release$	The two-digit release.	For example, 2.5.
$winusername$	The Microsoft Windows user name	
$winversion$	The Microsoft Windows version.	WINNT4, WIN2K, WINXP, WINVISTA, OTHEROS.