

Bytemark Symbiosis

Copyright © 2010-4 Bytemark Computer Consulting Ltd.

Permission is granted to copy, distribute and/or modify this documentation under the terms of the GNU Free Documentation Licence, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the licence is included in Appendix C.

COLLABORATORS

	<i>TITLE :</i> Bytemark Symbiosis		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Patrick J. Cherry, Steve Kemp, David Matthews, and James Carter	27th January 2015	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME
2009:1112	2009-11-12	Initial release.	PJC
2010:0427	2010-04-26	Renamed the project, and updated the documentation to match.	SKX
2012:0302	2012-03-02	Rewritten for the Squeeze release>	PJC
2012:0305	2012-03-05	Updated release notes for the Squeeze release.	PJC
2012:0420	2012-04-20	Added information about PostgreSQL backups, apache2 logging, and fixed the erroneous reference to <code>exim_rewrite_scan</code> in the release notes.	PJC
2014:1118	2014-11-18	Updated for Wheezy release.	JFC

Contents

I	User guide	1
1	The Bytemark Symbiosis system	2
1.1	Is there a web interface?	2
1.2	What about command-line access?	2
1.3	Is Symbiosis free software?	3
1.4	What software does Symbiosis come with?	3
1.5	About this documentation	3
2	What's new since the last release	4
3	Connecting to your server with FileZilla and SFTP	5
3.1	Connection details	5
3.2	Connecting using FileZilla	5
3.3	Common FileZilla recipes	7
3.3.1	Navigating local and remote filesystems	8
3.3.2	Creating a remote directory	11
3.3.3	Creating a remote file	14
3.3.4	Deleting files and directories	18
4	Website setup	22
4.1	Hosting a web page using your own domain	25
4.2	Handling wildcard domains	26
4.3	Testing a new domain	26
5	Configuring email	27
5.1	Creating a new mailbox	27
5.1.1	Forwarding Email	30
5.1.2	Filtering Email	30

5.1.3	Scanning email to prevent spam and viruses	30
5.1.3.1	Using SpamAssassin to detect and reject or tag spam	30
5.1.3.2	Using ClamAV to detect and reject, or tag, emails with viruses	31
5.2	Testing a new mailbox, via webmail	31
5.3	Configuring email clients	33
6	Chat server (XMPP/Jabber) Configuration	34
7	Setting up per-domain FTP Access	35
7.1	Setting FTP quotas	38
8	Managing the MySQL database	39
9	Scheduled tasks	41
9.1	The crontab format	41
9.2	Mailing the output	42
10	Automated backups	43
10.1	Offsite backups	44
10.2	Backup reporting	44
11	Keeping Your System Secure	45
11.1	Checking system notifications	45
11.2	Avoiding weak passwords	45
11.3	Keep your software current	46
11.4	Local security considerations	46
12	Connecting to your server via SSH	47
12.1	Why SSH access?	47
12.2	Using PuTTY to connect via SSH	47
12.3	Using a terminal program to connect via SSH	49
13	Configuring SSL Hosting	51
13.1	Adding an additional IP address	51
13.2	Generating an SSL key and certificate request	52
13.3	Purchasing a certificate	53
13.4	Uploading your new certificate, and optional bundle	53
13.5	Making SSL mandatory	54

II Reference	55
14 Installing and administering Symbiosis	56
14.1 Installing Symbiosis running on Debian 7.0 (wheezy)	56
14.2 Upgrading Symbiosis running on Debian 6.0 (Squeeze)	56
14.3 Release notes	58
14.3.1 Backup changes	58
14.3.2 New version of Dovecot	58
14.3.3 Command rationalisation	58
14.4 Packages installed by Symbiosis	59
14.5 Systems administration and Symbiosis	59
14.5.1 Use of <code>root</code> , and other users	60
14.5.2 Customising configurations	60
14.5.3 Other configuration styles	61
15 Website Configuration	62
15.1 Getting started	62
15.2 CGI scripts	62
15.3 Statistics	63
15.4 Testing new websites	63
15.5 Displaying the same content under two domains	63
15.6 Redirecting to the preferred website domain	64
15.7 Custom Apache configuration	64
15.8 SSL Configuration	64
15.8.1 Generating a self-signed certificate	65
15.9 Logging	65
15.10 Configuration layout	66
16 Email Configuration	67
16.1 Port Configuration	67
16.2 Accepting email for a domain	67
16.3 Email for Unix users.	68
16.4 Password files	68
16.5 Suffixes	69
16.6 Enforcing mailbox size with quotas	69
16.7 Server-side filtering using Sieve	69
16.8 Forward files	69

16.9	Vacation messages	70
16.10	Email alias lists	70
16.11	Customising SpamAssassin	71
16.12	Filtering mail using headers	71
16.13	Using real-time blacklists from Spamhaus	71
16.14	Manually blocking incoming mail from specific sources	72
16.15	Installing Roundcube webmail	73
16.16	Configuration layout	73
17	XMPP Reference	75
18	Setting up domain based multi-user FTP/FTPS Access	76
19	Firewall Reference	78
19.1	Allowing and denying access to services	78
19.2	Predefined special rules	79
19.3	An example firewall	80
19.4	Allowing web applications to make remote connections	81
19.5	Making custom additions to your firewall	82
19.6	Blocking abusive remote hosts	82
19.7	Whitelisting "known-good" IP addresses	83
19.8	Disabling the firewall	84
19.9	Configuration layout	84
20	DNS Hosting	85
20.1	Example DNS records	85
20.2	Adding a wild-card hostname record	87
20.3	Moving domains between machines using the Bytemark content DNS service	87
20.4	Configuring SPF and DKIM records	87
20.4.1	Adding SPF records	87
20.4.2	Adding DKIM records	87
21	Scheduled tasks	89
21.1	Testing the crontab	89
21.2	System scheduled tasks	90
22	Database configuration	91
22.1	Enabling remote MySQL access	91
22.1.1	Opening the firewall for MySQL	91
22.1.2	Adding a user with remote privileges	91

23 Backup Reference	93
23.1 Configuration	93
23.2 Advanced Configuration	93
23.3 Listing Backup Contents	93
23.4 Restoring From Backup	94
23.5 Recovery From Earlier Backups	95
23.6 Offsite backup storage	95
23.7 Recovering from the offsite backup storage	96
23.8 Trimming the size of the local backups.	96
24 Service Monitoring	98
III Support Guide	100
25 Troubleshooting Symbiosis	101
25.1 Database problems?	101
25.2 Firewall problems?	101
25.3 Package problems?	102
25.4 Permission problems?	102
25.5 SSL problems?	102
26 FAQ	104
27 Reporting issues	106
IV Appendices	107
A Email client setup	108
A.1 Generic client configuration.	108
A.2 Configuring Mozilla Thunderbird 10.0.	109
A.3 Configuring Windows Live Mail	111
A.4 Configuring Apple Mail	114
B XMPP client setup	117
B.1 Pidgin XMPP client configuration.	117
C GNU Free Documentation License	122

V Glossary 129

VI Bibliography 133

 27.1 Bibliography 134

28 Index 135

Preface

The Bytemark Symbiosis system makes it easy to manage website and email hosting without much prior technical knowledge. After installing a connection program, setting up a website or an email account is as easy as creating folders and files on your hard drive.

The system is based on the stable version of [Debian GNU/Linux](#), with a few light touches here and there to make things easier to use.

We have written a comprehensive [user guide](#) to help people get started with their Symbiosis install.

The complete specification is documented in the [reference guide](#).

Part I

User guide

Chapter 1

The Bytemark Symbiosis system

Symbiosis is a system that helps in the day to day tasks involved in administration of a typical server on the internet. Its goal is to simplify running web and email hosting across multiple, separate domains, along with all their associated services.

Specifically, Symbiosis handles

- [Web server configuration](#), including uploading content via FTP,
- [DNS hosting](#) using the Bytemark [content DNS system](#),
- [Email server configuration](#), including web mail, SPF and DKIM,
- [XMPP server configuration](#), providing email users with XMPP instant messaging accounts,
- [Database access](#),
- [Automatic daily backups](#) which are synchronised to your Bytemark [backup space](#), and
- [A firewall](#) and the automatic installation of [security updates](#).

Currently Symbiosis packages are available for installation on [Debian GNU/Linux](#).

1.1 Is there a web interface?

No. All typical day-to-day jobs, such as adding new web sites, or email addresses, or uploading content, can be done using SFTP, i.e. FTP over SSH, by creating files and directories. [FileZilla](#) is the recommended program for this.

This should not be viewed as a disadvantage; any confident computer user should be able to manage a Symbiosis system. Effort has been put into making the layout of the various systems as obvious as possible, and making systems *just work*.

1.2 What about command-line access?

Symbiosis has been written by an experienced team of developers and system administrators with the goal of having an easy-to-use hosting system that met their exacting standards.

Unlike other control panel systems, one of the aims of the Symbiosis system is to keep the "magic" as transparent as possible. As far as possible standard tools and techniques have been used to configure the various services on a Symbiosis system. This allows users to tailor these configurations as they need, as well as working through standard distribution updates and upgrades.

1.3 Is Symbiosis free software?

Yes! Symbiosis is both [Free Software](#) and [Open Source](#) software. All the parts Bytemark have written have been released under the [GNU General Public Licence, version 2](#) or later, or the [Apache licence, version 2.0](#). All the source code is available for scrutiny on the Symbiosis [project site](#). There is also a [issue tracker](#) to report any problems encountered, or to request improvements.

This documentation is released under the [GNU Free Documentation Licence](#) or later. It also has a [project site](#), and [issue tracker](#).

1.4 What software does Symbiosis come with?

Symbiosis uses the following software, all of which is open-source:

- [Apache 2.2](#) web server,
- [Exim 4.80](#) mail transport agent,
- [Dovecot 2.1](#) IMAP, POP3, and ManageSieve server,
- [MySQL 5.5](#) database server,
- [PHP 5.4](#) scripting language, with the following modules
 - mysql,
 - curl,
 - imagemagick,
 - mcrypt,
 - mhash,
 - xmlrpc,
 - gd,
- [SpamAssassin](#) spam filtering, and [ClamAV](#) anti-virus,
- [Pure-FTPd](#) FTP server,
- A webmail interface, using either [SquirrelMail](#) or [Roundcube](#).

1.5 About this documentation

What follows is step by step instructions to get up and running with controlling your server and setting up core services. The screen shots are taken from a Windows system, but all the programs used are also available for Mac OS X and GNU/Linux desktop systems.

Throughout the documentation, the example server used is **example.vm.bytemark.co.uk**. The example domain used is **my-brilliant-site.com**. These should be substituted as appropriate.

Chapter 2

What's new since the last release

The current release is based on Debian 7.0, code-name wheezy. Since the last release of symbiosis, the following features have been implemented.

New features

- Amazing new splash screen on boot.
- All ruby code now works with both 1.8 and 1.9.3 versions of ruby.
- Domains can now support DKIM and SPF (see Section [20.4](#)).
- Local users can now receive email (see Section [16.3](#)).
- New EventMachine based logger for Apache.
- All mail users can now have XMPP chat accounts (see Chapter [6](#)).
- Domains may now have FTP sub users (see Chapter [18](#)).

Chapter 3

Connecting to your server with FileZilla and SFTP



Before you start this chapter

1. Install FileZilla on your computer; it is freely available from its [project website](#) for Windows, Mac OS X, and GNU/Linux.
-

In this chapter you'll learn how to connect to your server ready to transfer files using the FileZilla program. It has been assumed that you have a working copy of this program installed on your desktop computer.

3.1 Connection details

A server installed with Symbiosis will be running *SSH*, and will have had the **admin** user account created. This allows you to connect via *SFTP* to administer the machine.

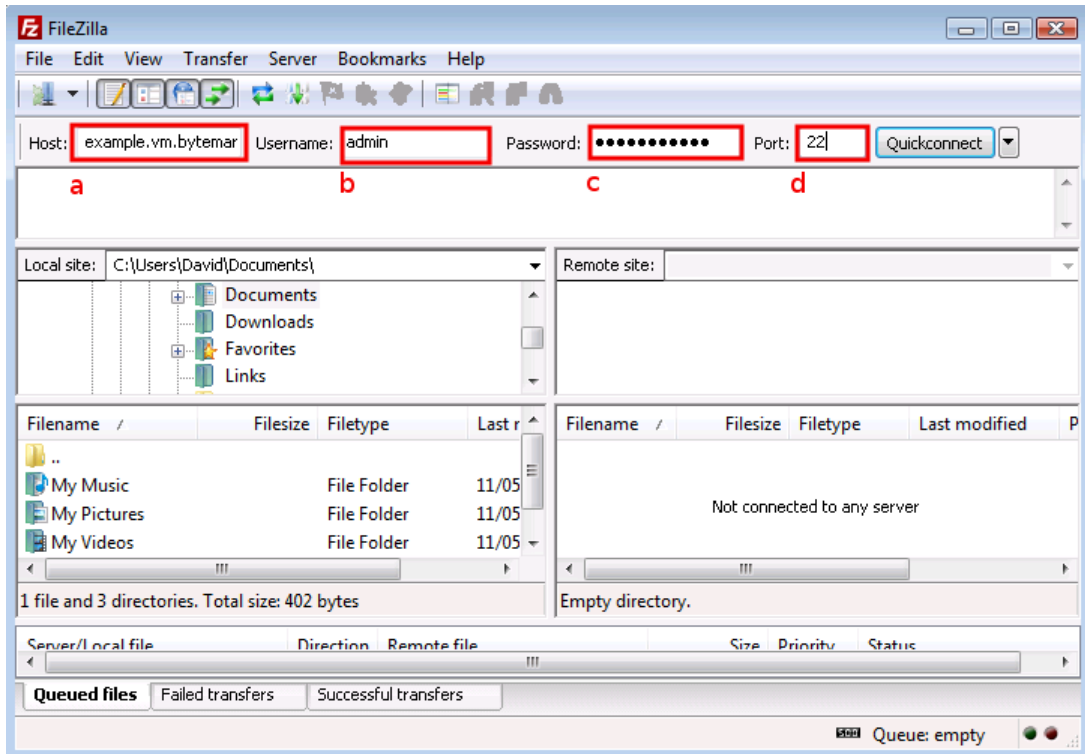
The **admin** account should be used when administering a Symbiosis system to ensure that files and directories have the correct permissions.

Usage of SFTP is mandated for administrating the machine, such that all data are passed encrypted over the network.

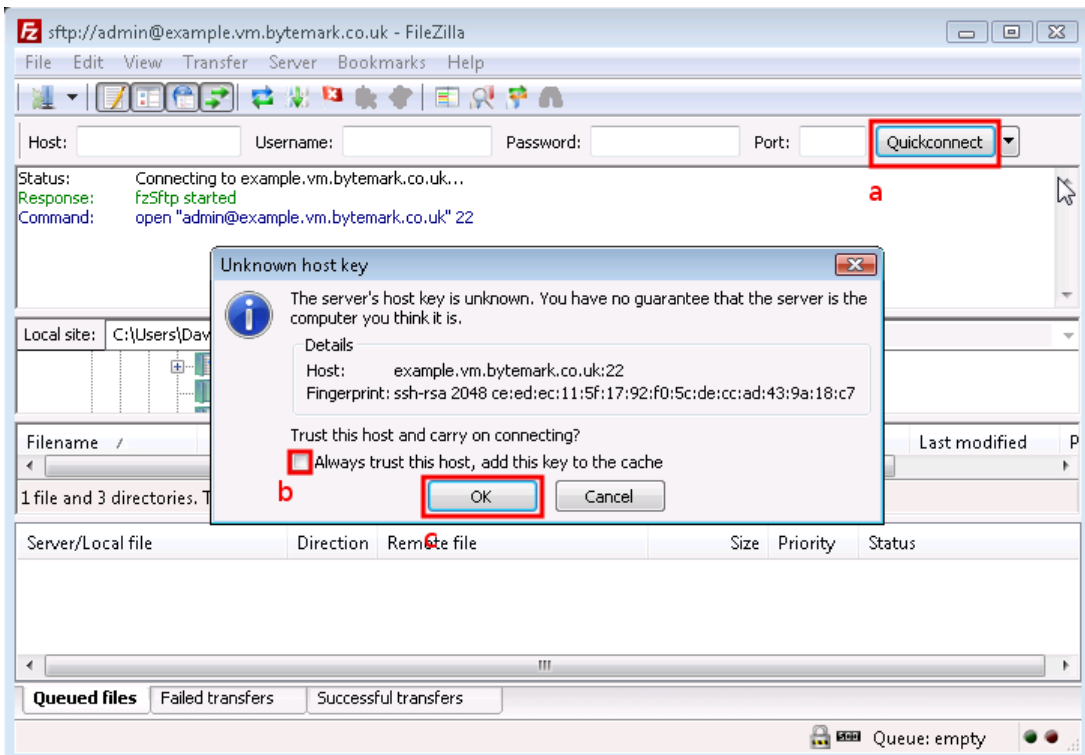
3.2 Connecting using FileZilla

Throughout the documentation the example server name **example.vm.bytemark.co.uk** should be substituted for your own server name.

1. Start FileZilla and enter the details in the text fields below the program's toolbar. The name of your server goes in the Host text field **a** and **admin** in the Username field **b**.
2. Complete the connection details by filling in the Password field **c** with the password for the **admin** user (by default, this is the same as the root password provided during the setup of your VM), and the standard SSH port number, **22** in the Port field **d**.



3. Click the Quickconnect button **a** to the right of the text fields; the first time you do this you'll get a warning message that is safe to ignore, so check its Always trust ... box **b** and click the OK button **c**.

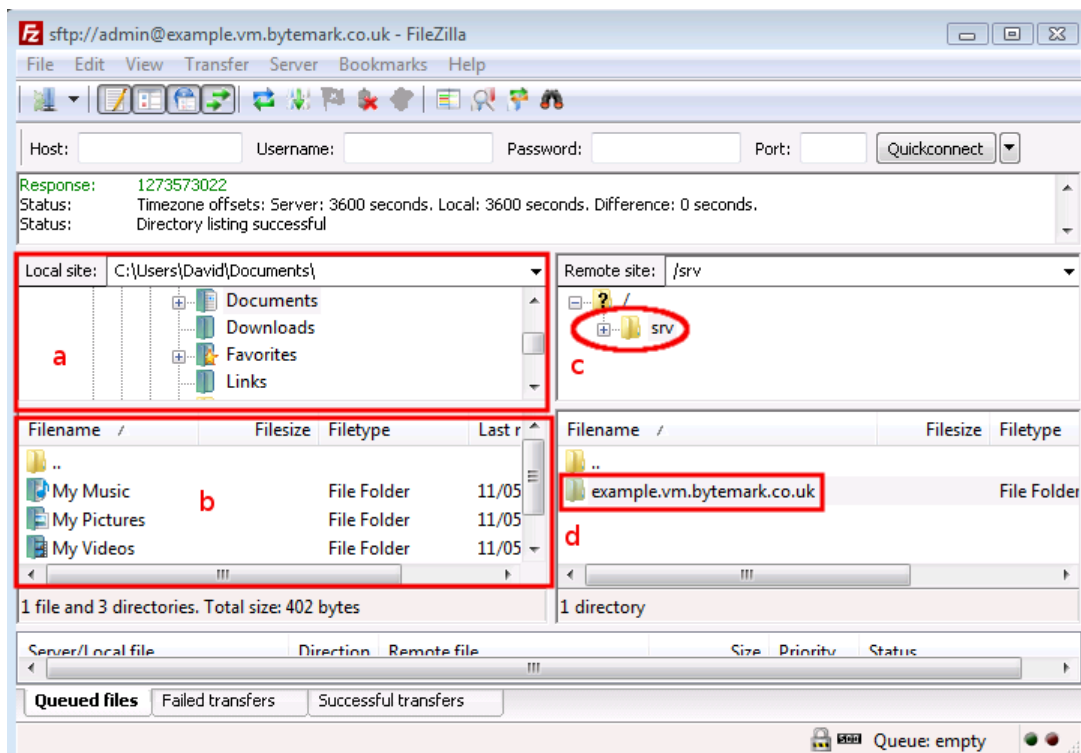


4. In the text area immediately below the Quickconnect bar you'll see messages scroll by as the connection to the server is made.

**Tip**

You won't need to enter those details each time you connect. Click the small button to the right of the Quickconnect button to reveal the `sftp://admin@example.vm.bytemark.co.uk` as a history item. In future simply select that link.

The following figure shows FileZilla's layout after successfully logging in. The display is divided into four main sections, the top left pane shows a directory tree, with the directories on your local computer, labelled **a**. Beneath that is a listing showing the contents of the currently selected local directory, labelled **b**. Then the top right pane shows the directory tree of the remote machine. When logging in as **admin** this will show `/srv/` (**c**). Finally beneath that is the contents of that directory. Initially this will only contain one directory named after the machine. In this case `example.vm.bytemark.co.uk/` is shown (**d**).



Once you've been able to successfully connect to your server, via FileZilla, you may proceed to [configure email](#), or [setup your website](#).

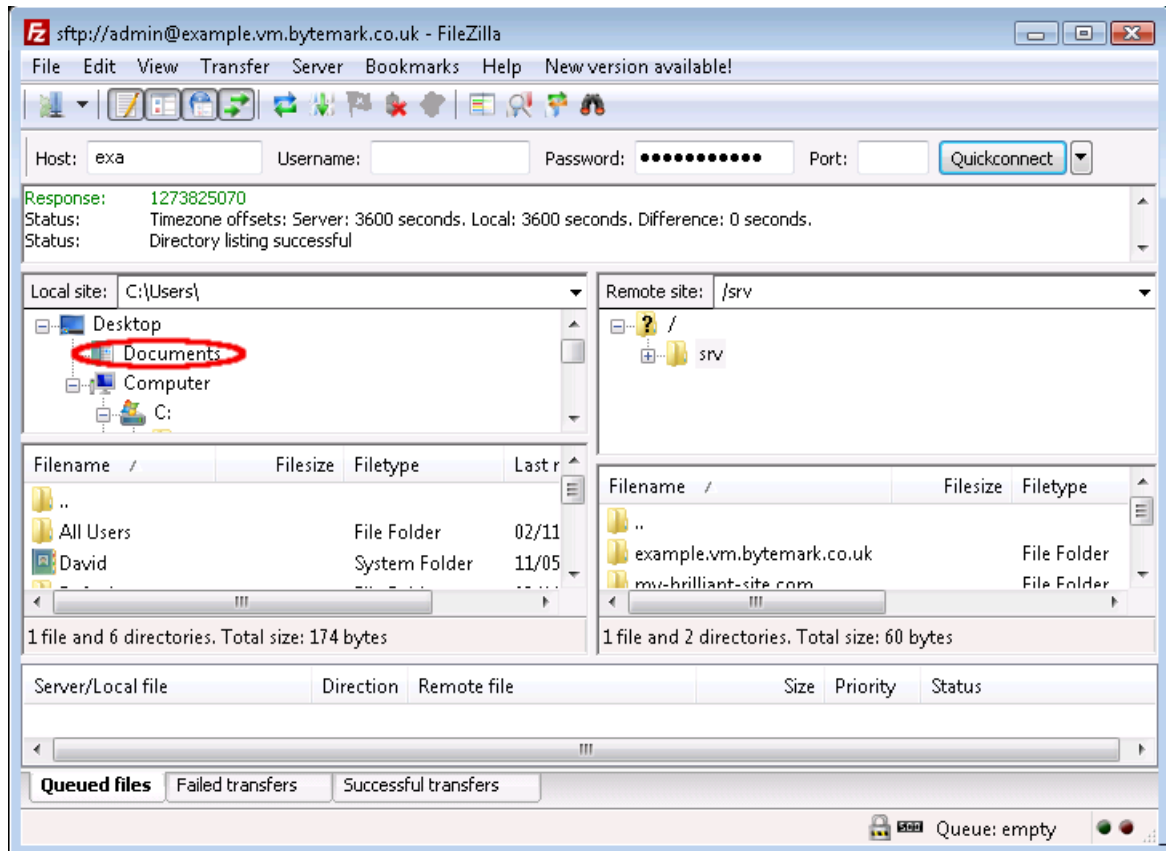
3.3 Common FileZilla recipes

This section demonstrates how to carry out some common tasks with the FileZilla client:

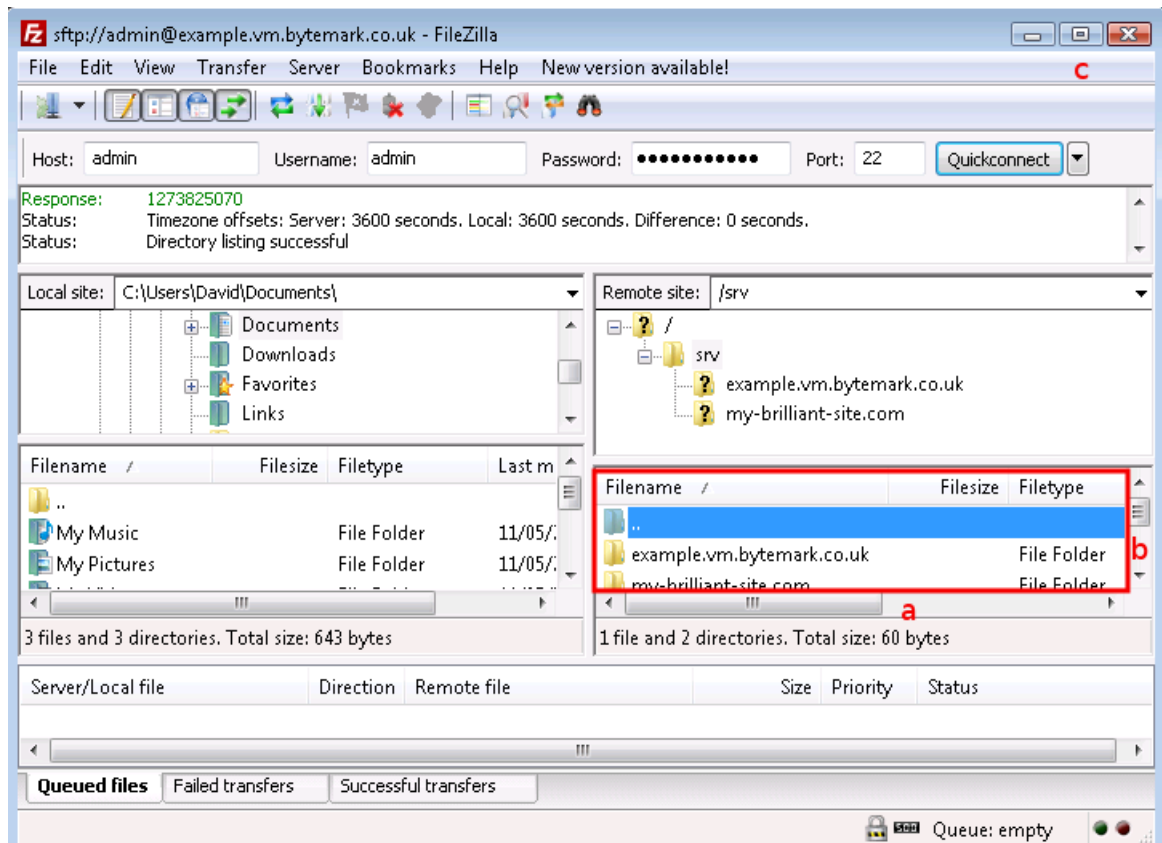
- [Navigating local and remote filesystems](#)
- [Creating a remote directory](#)
- [Creating a remote file](#)
- [Deleting files and directories](#)

3.3.1 Navigating local and remote filesystems

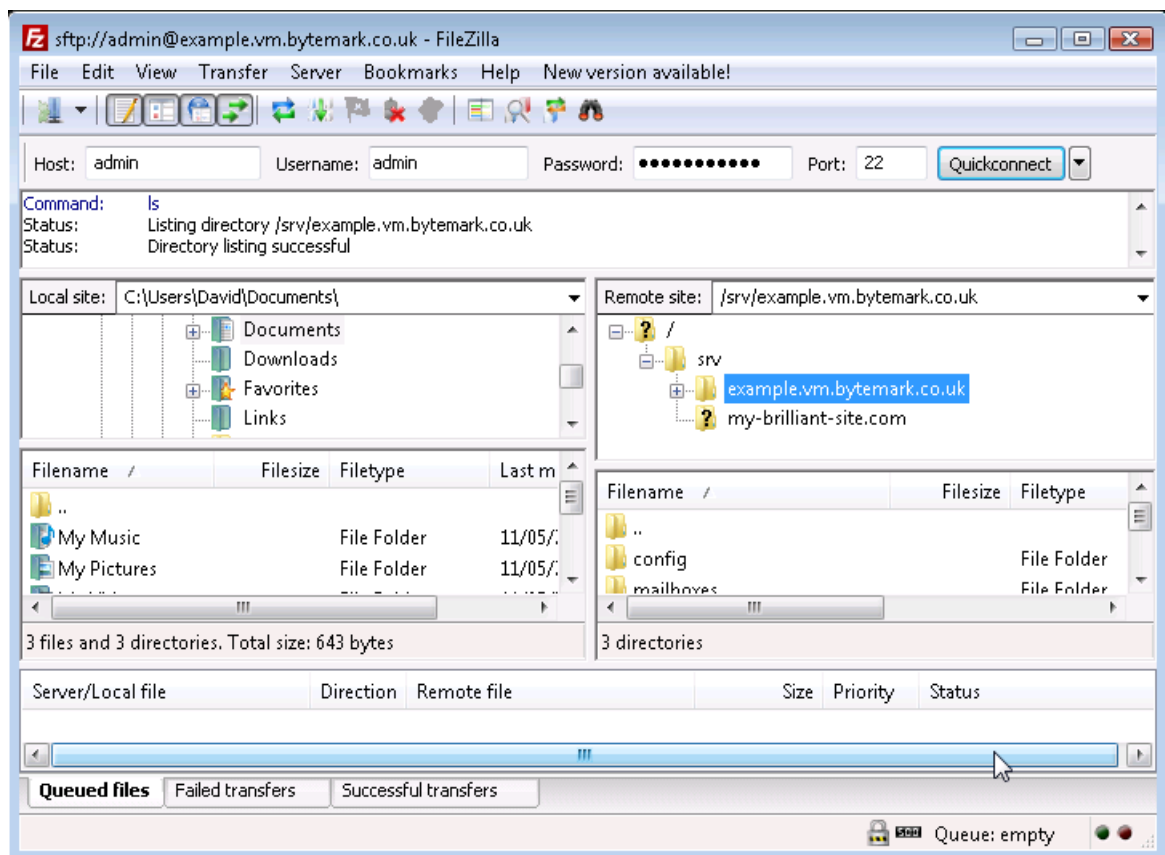
1. To open the `/srv/` directory on the server, click the + icon.



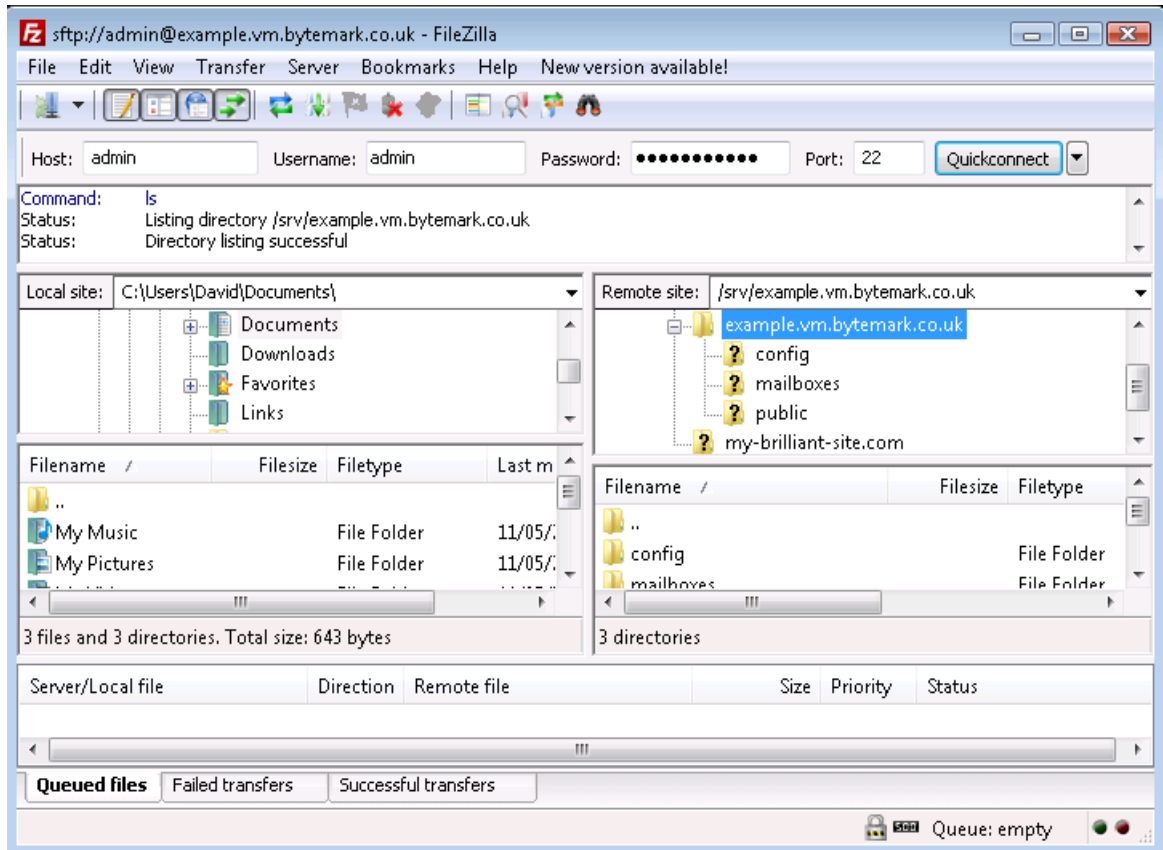
2. Notice that the folders that appear in the tree display are already displayed in the Filename window. You can use the scrollbars **a** and **b** to adjust the view; as in any desktop window you can also use the Control **c** to expand the FileZilla display to full screen or just drag it's corners in the usual way.



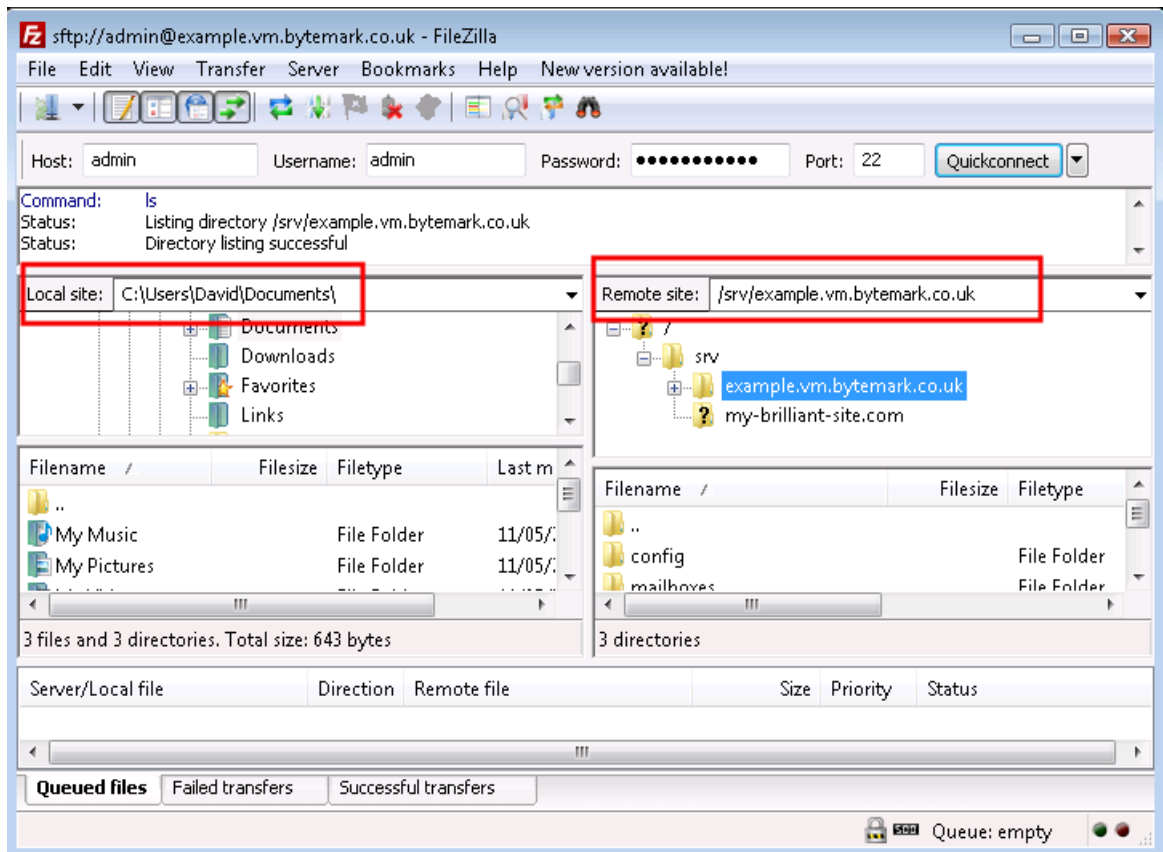
3. When you click on the example.vm.bytemark.co.uk label (not the directory icon), the + control appears. The contents of the directory are already displayed in the Filename window.



- Click the + control to see those contents as part of a tree view and notice that you now have a - control, which could be used to close this detail of the file system structure.



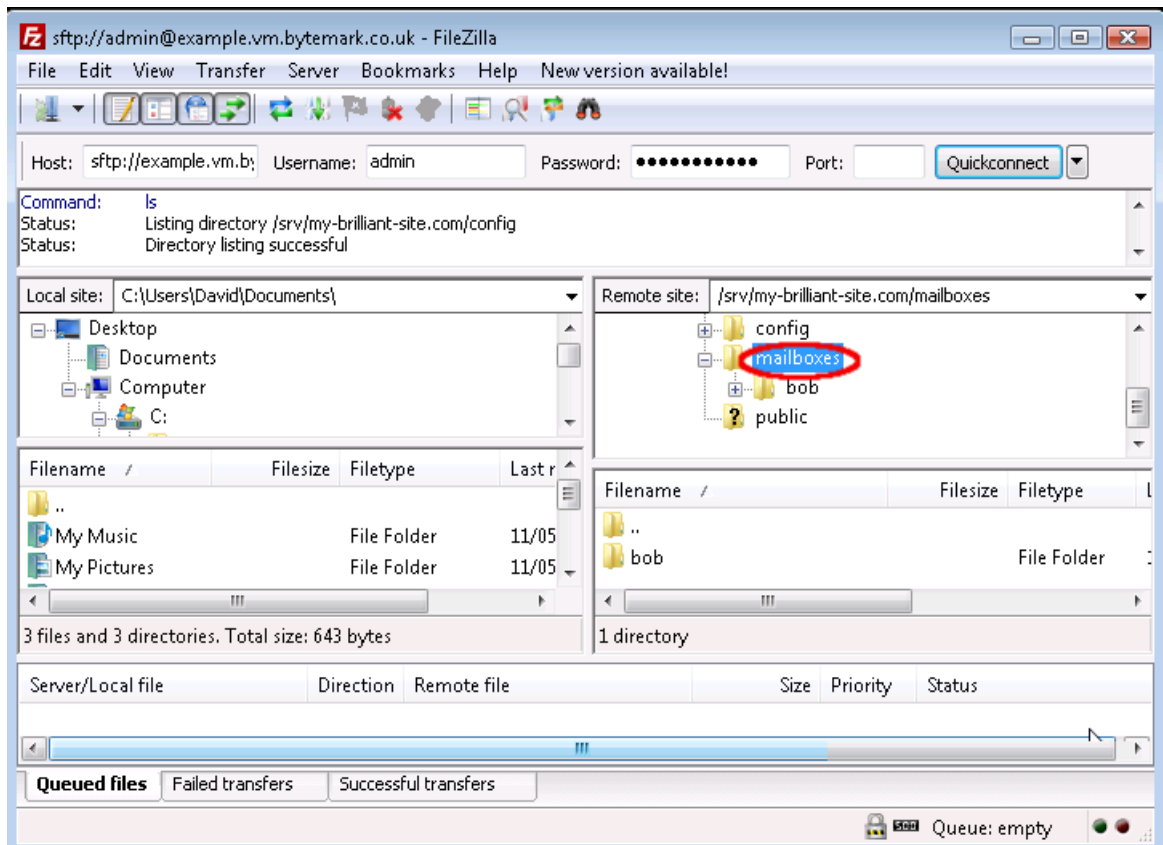
- Those operations were all carried out on the right side of the screen where the Remote site:, in this case the server **example.vm**, is represented. Comparable operations can be carried out on the left side of the screen, where the Local site:. represents the file system on the desktop machine.



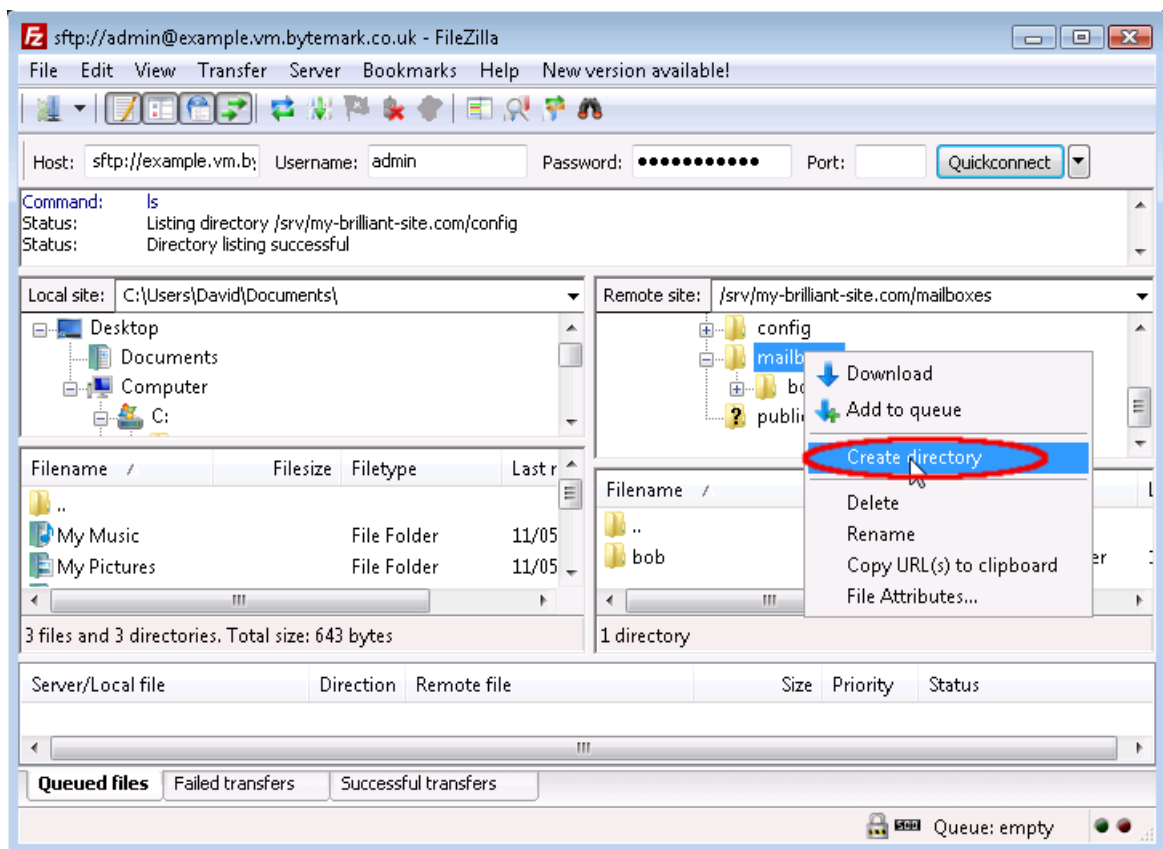
3.3.2 Creating a remote directory

In this walkthrough, a mailbox will be created for a user **alice**. That is done by creating a directory under the `/srv/my-brilliant-site.com/mailboxes/` directory. (The configuration of email is described fully in Chapter 5.)

1. Highlight the parent directory by pointing at the mailboxes label (not the icon) and left-clicking.

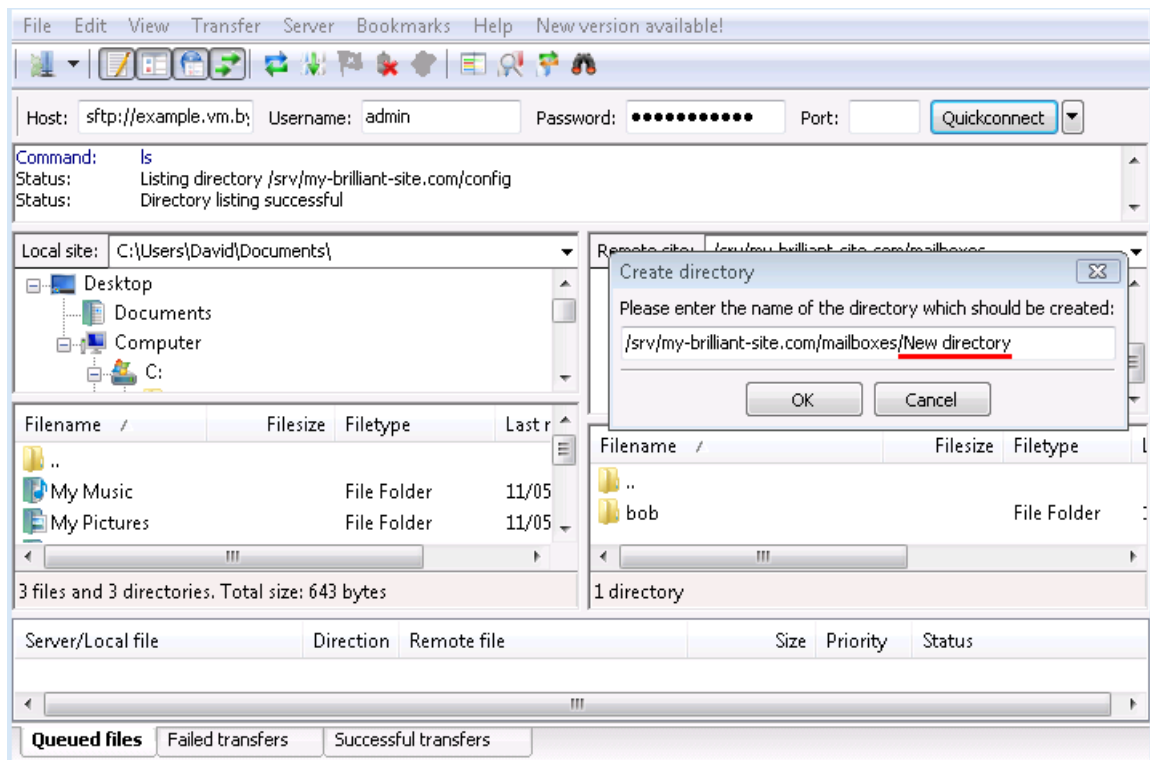


2. Right click to bring up the menu and select Create directory.

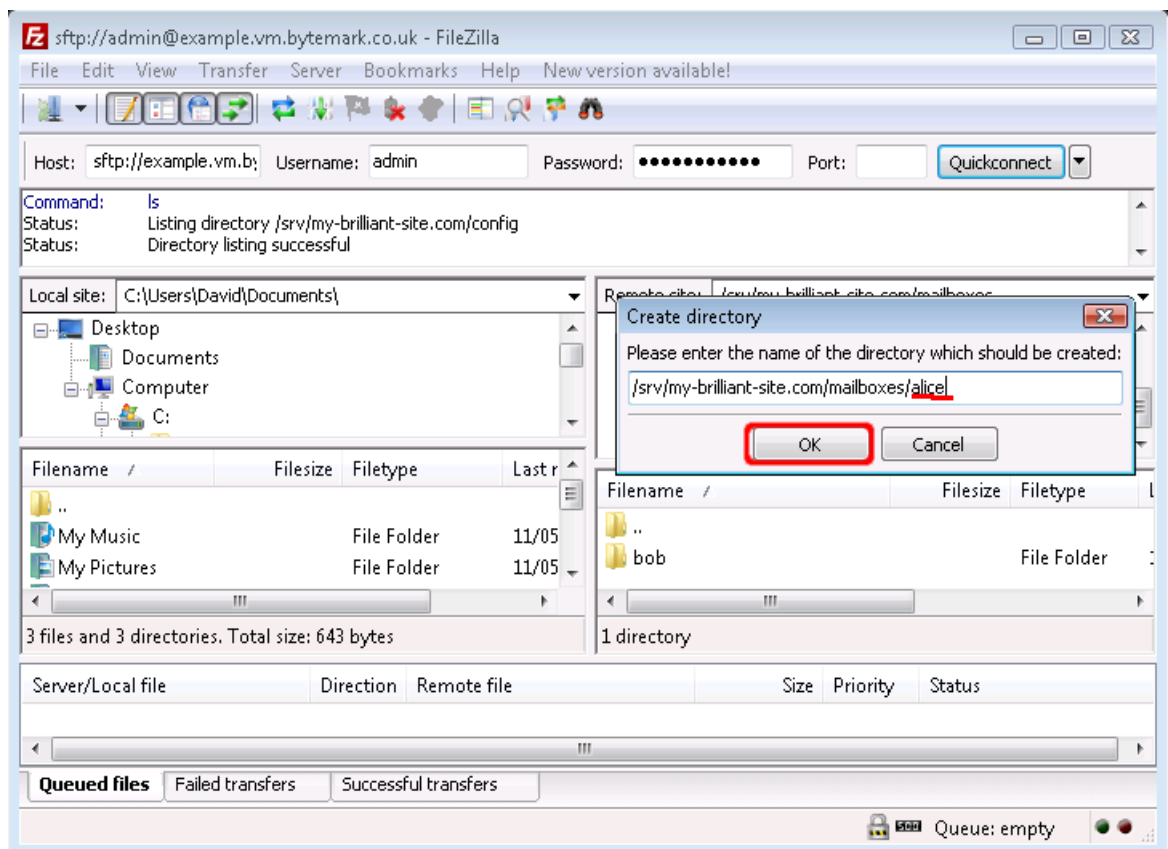


3. The Create dialog starts; the default path /srv/my-brilliant-site.com/mailboxes/. is as we

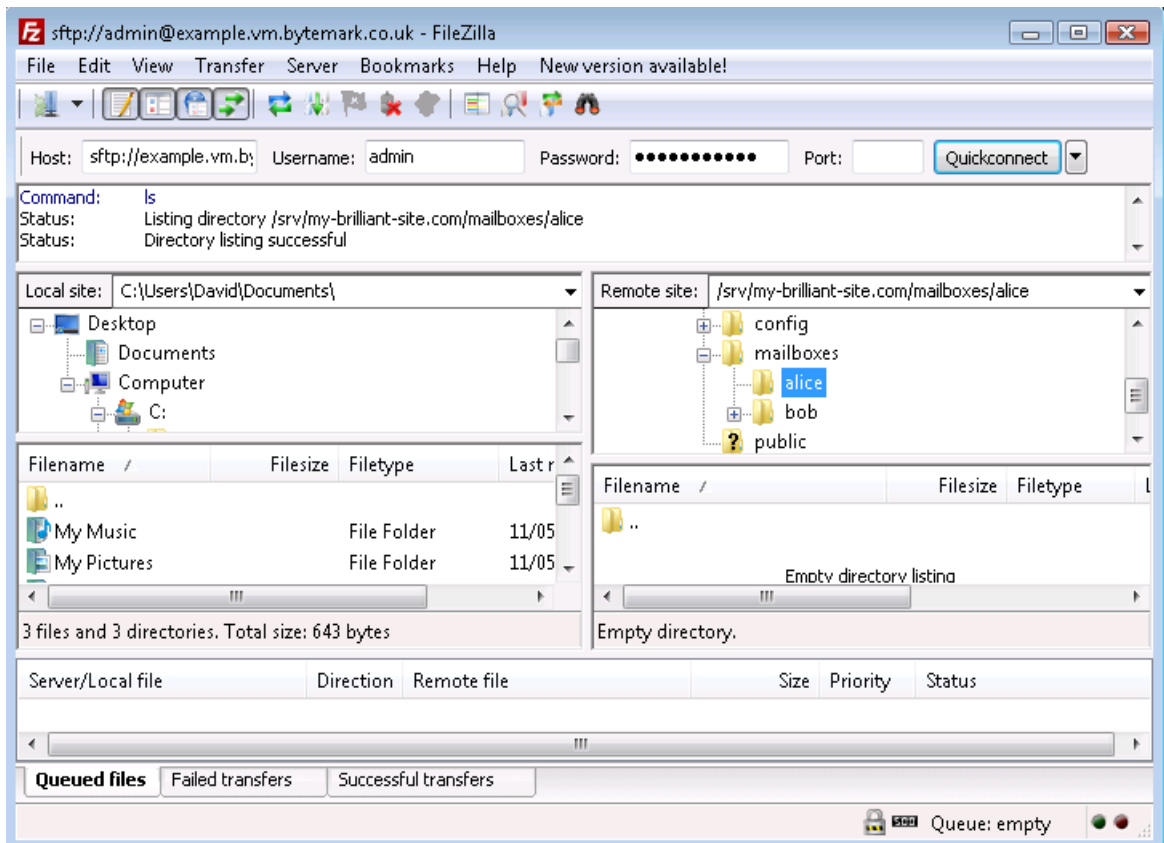
want it, but not the default name `New directory/`.



4. Edit that, replacing `New directory/` with `alice/`, then click the OK button.



5. The `/srv/my-brilliant-site.com/mailboxes/alice/` has been created.



3.3.3 Creating a remote file

Unfortunately creating a file upon the remote server cannot be completed *directly* within FileZilla, but that limitation can be skirted around by creating the file on your local machine and then uploading it to the correct location on the server.

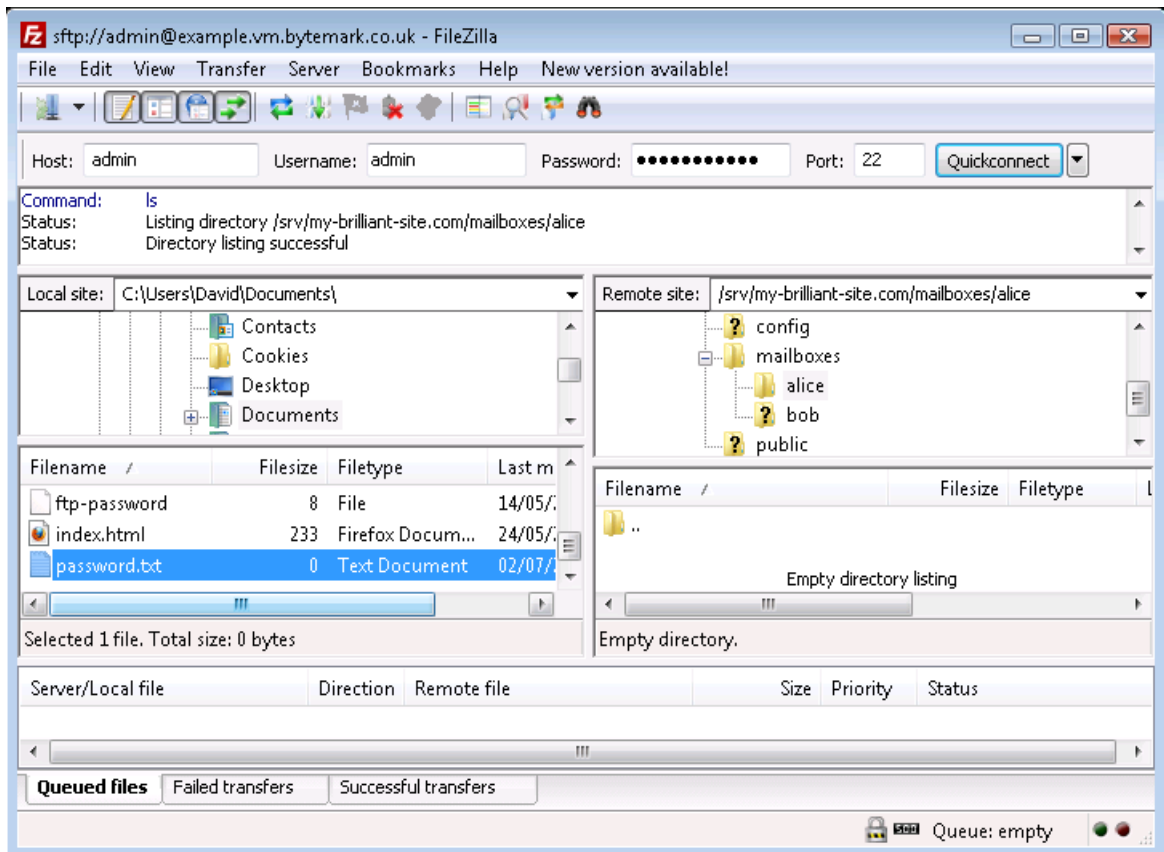


Note

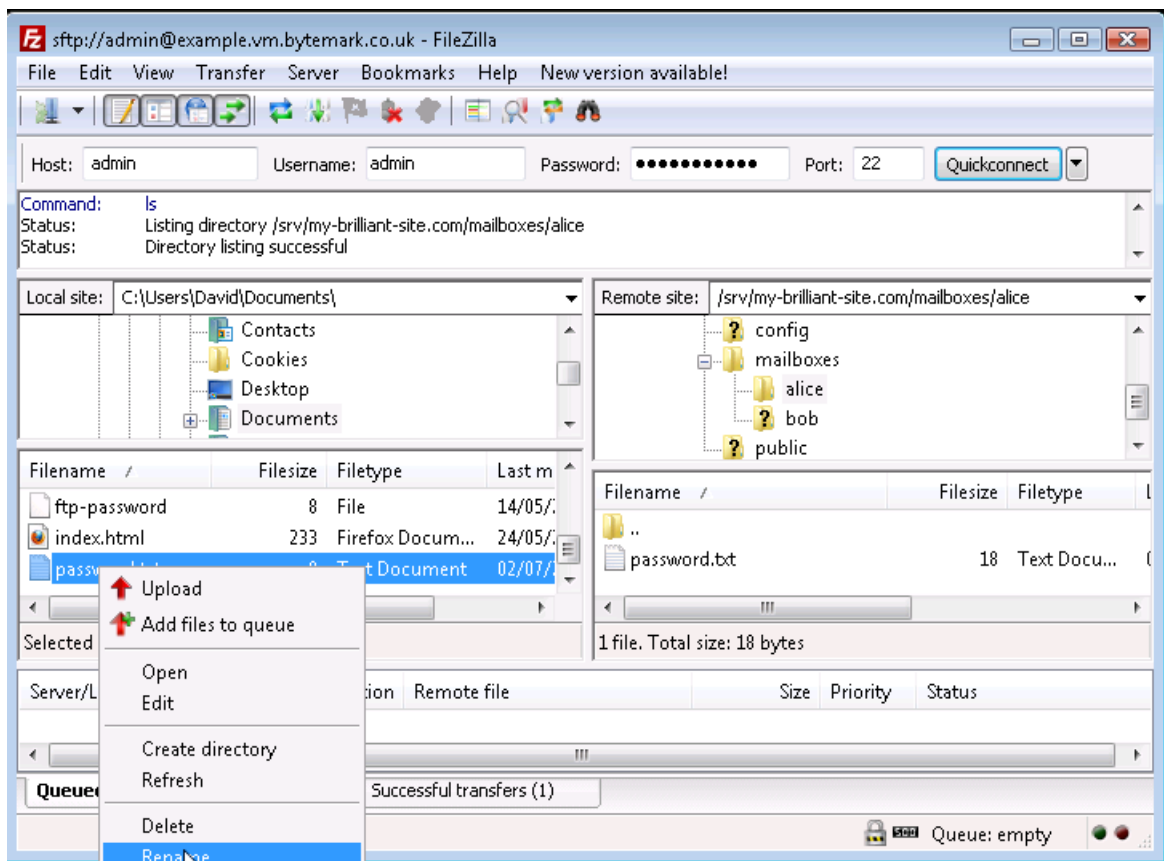
Windows desktop systems tend to silently add the .txt extension when you create a plain text file; this means that the file will need renaming before uploading.

This walkthrough demonstrates the procedure for allowing the user alice to logon to the mailbox at the my-brilliant-site.com domain; in doing so, it covers the creation, upload and rename operations.

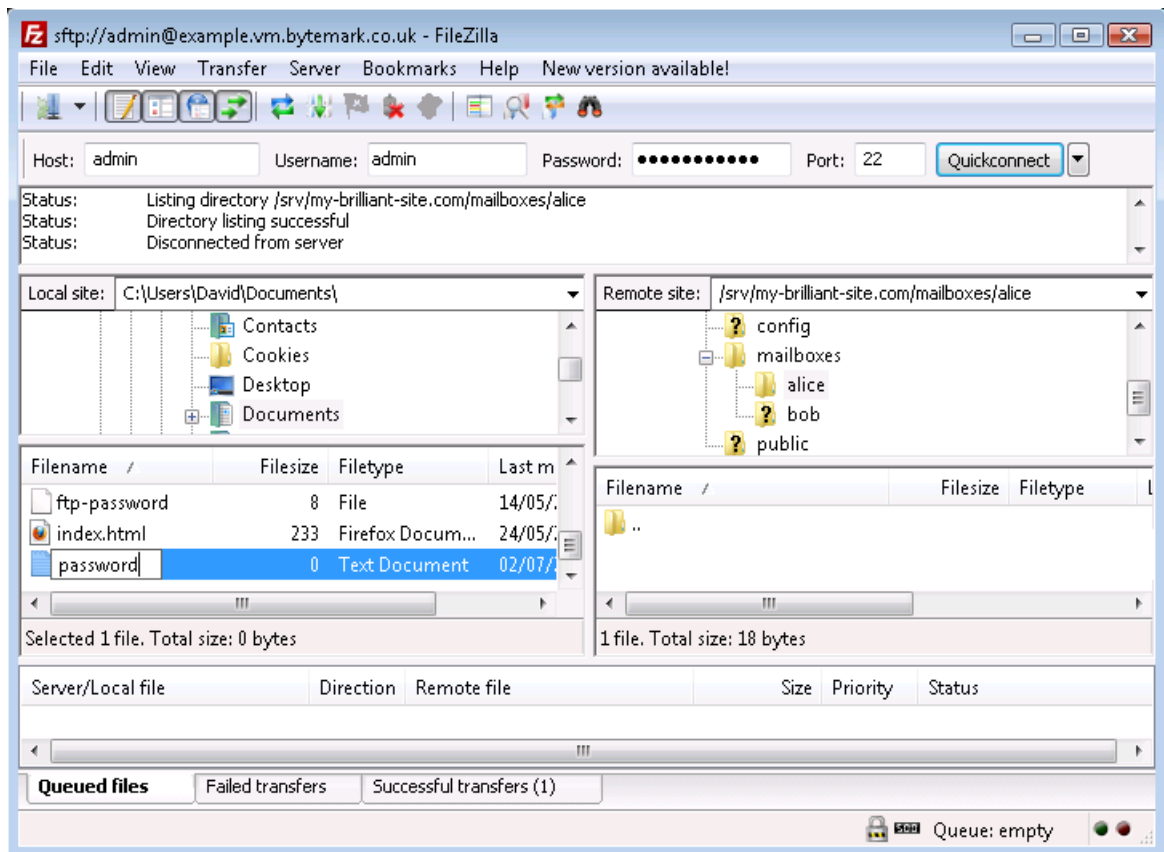
1. The Notepad program has been used to create a plain text document that contains a secure password on a single line. Although the name "password" was specified as the filename, FileZilla reveals that the ".txt" extension has been silently added to that.



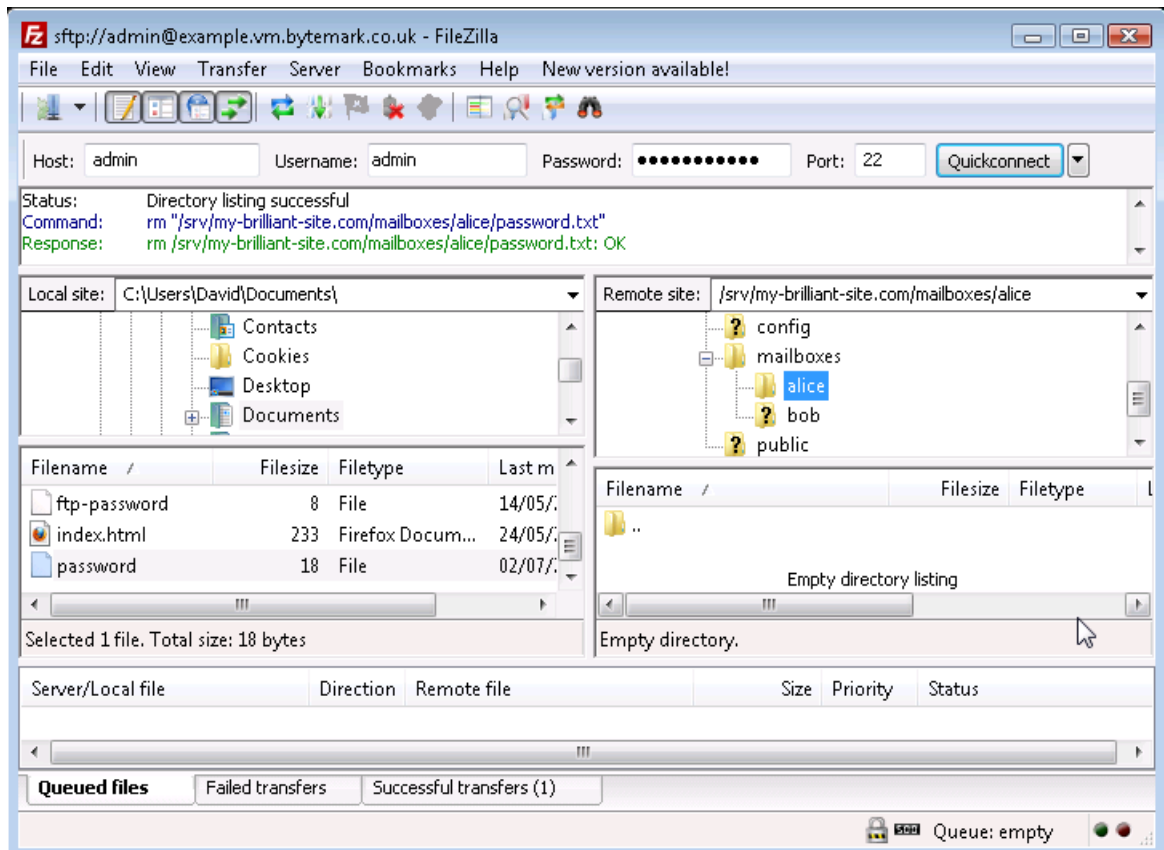
2. Right click on the `password.txt` file to bring up the menu and select the Rename option.



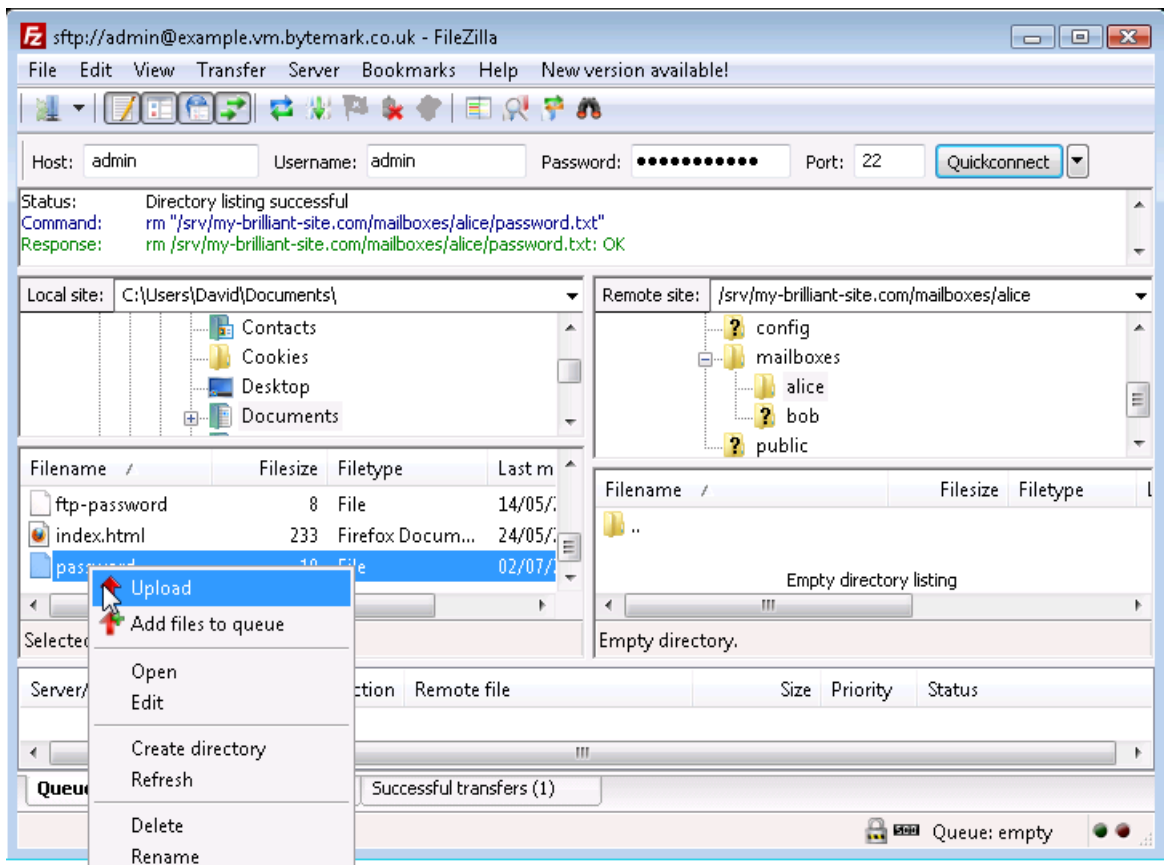
3. Rename the file by removing the unwanted .txt extension.



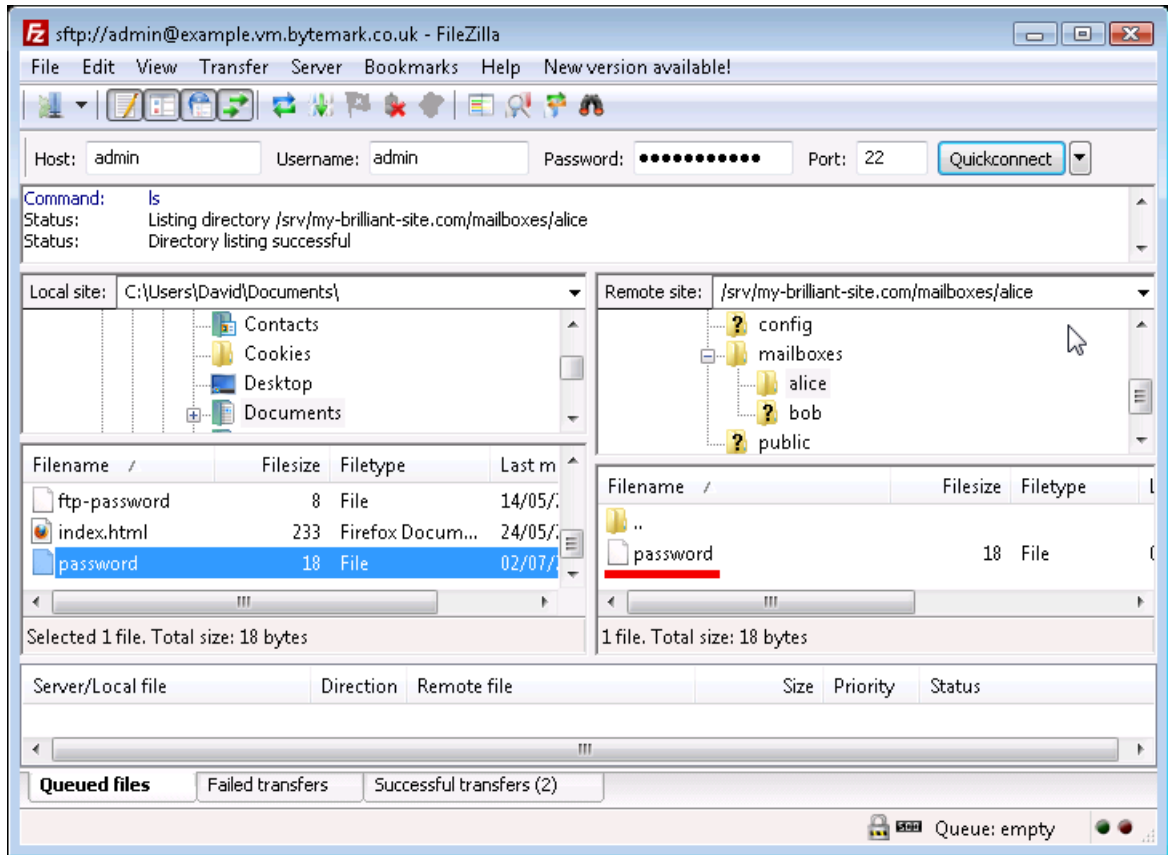
4. Press your Enter key to complete that; the file has been renamed from `password.txt` to `password`. Move to the Remote site: area on the right side of the FileZilla display and navigate to the `/srv/my-brilliant-site.com/mailboxes/alice/` directory



- Moving back to the left side of the FileZilla display, again highlight the `password` file and right click to bring up the menu. This time select the Upload option.

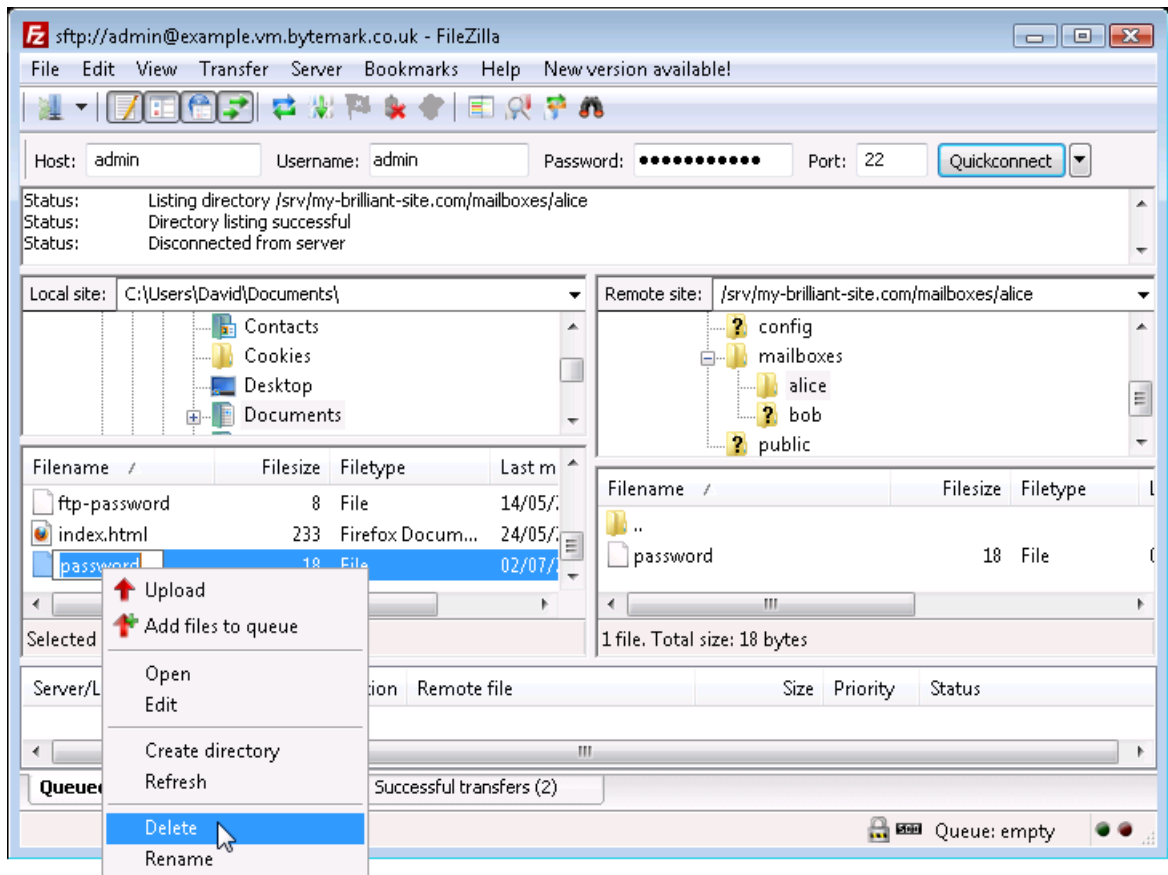


- The `password` file has been created on the server in effect, by uploading it from the local desktop machine. An alternative method of achieving this is to select the file and in the local Filename area and drag it to the Filename area of the server.

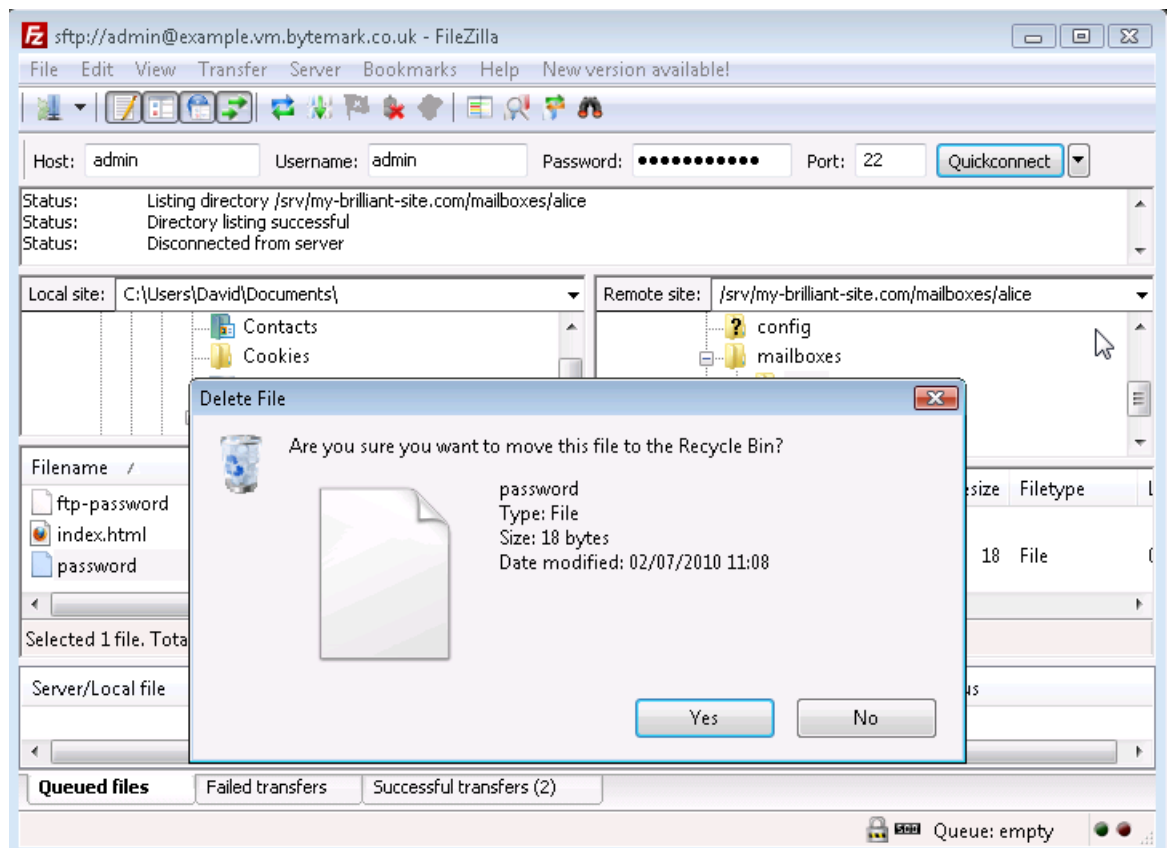


3.3.4 Deleting files and directories

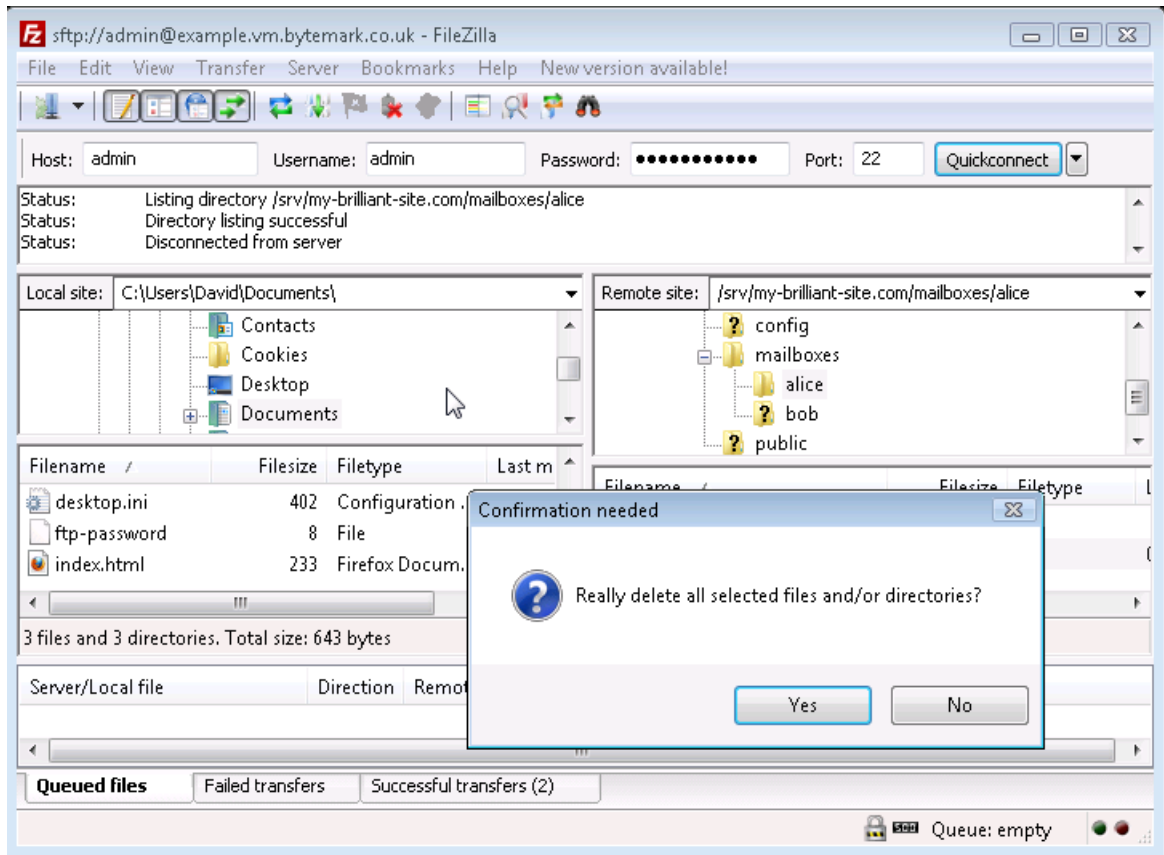
- The `password` file on the local machine is no longer needed; select and right click then choose the Delete option.



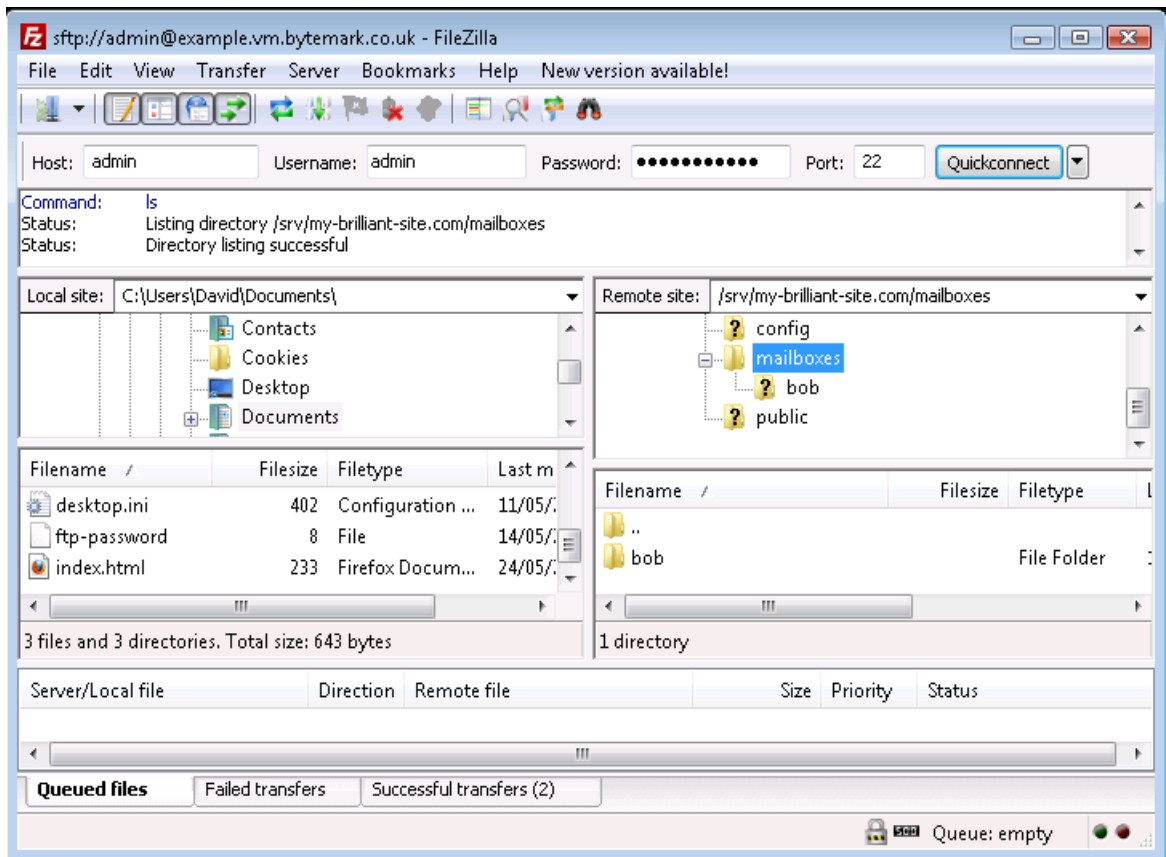
2. Confirm that you do want to delete the file by pressing the Yes button in the Delete file dialog.



- The local password file has been deleted. In the same way files and also directories can be deleted from the server, the only difference being that the Confirmation needed dialog that comes up is less detailed than the local Delete file dialog. Below is what would be seen if user `alice` was to be removed; the `alice/` directory and the Delete option have been selected.



- The Yes button in the Confirmation needed dialog was selected and the `alice/` directory has been removed.



Chapter 4

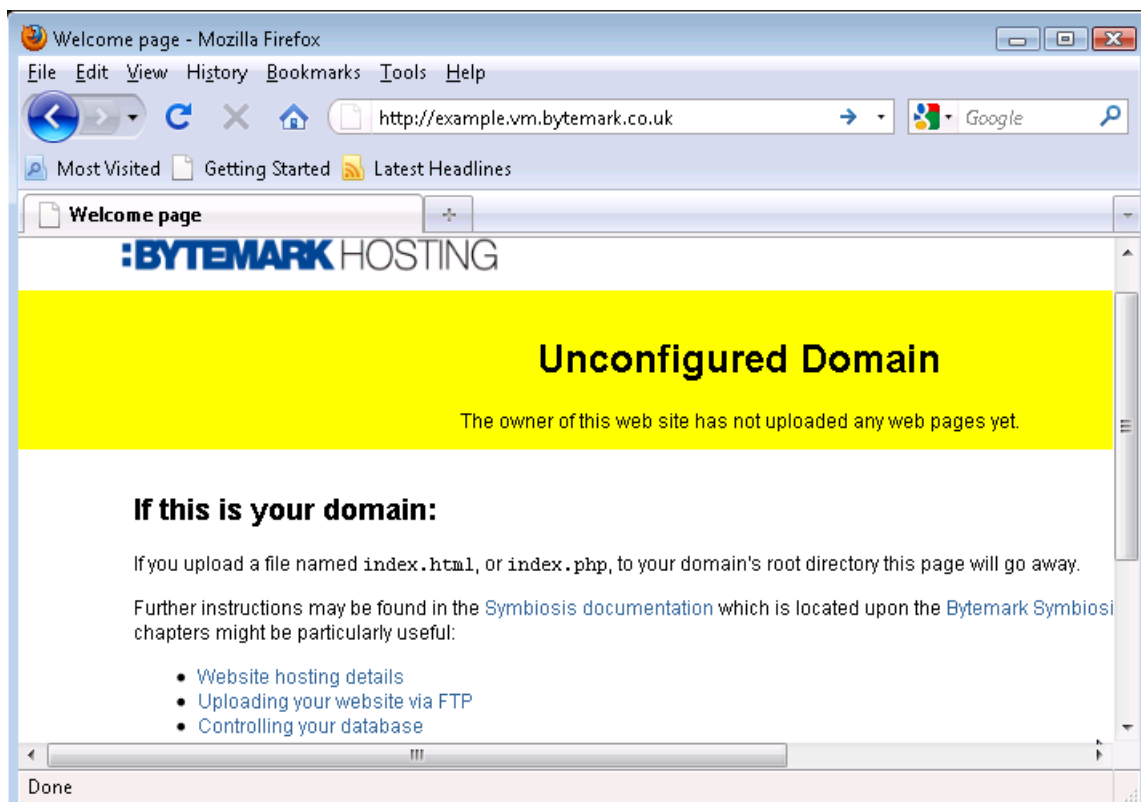
Website setup



Before you start this chapter

1. Connect to your server over [SFTP](#) using FileZilla (see Chapter 3).

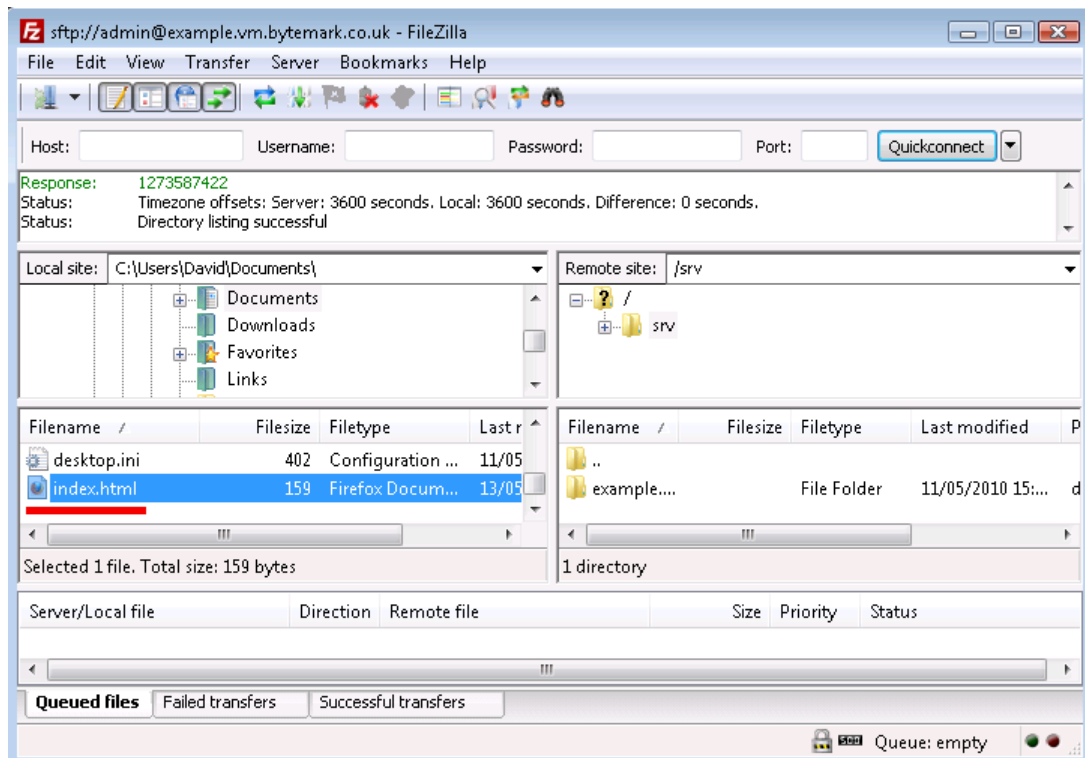
Start up your web browser and enter the machine's name in the location bar, e.g. <http://example.vm.bytemark.co.uk>. As you can see, your machine is already hosting a default page.



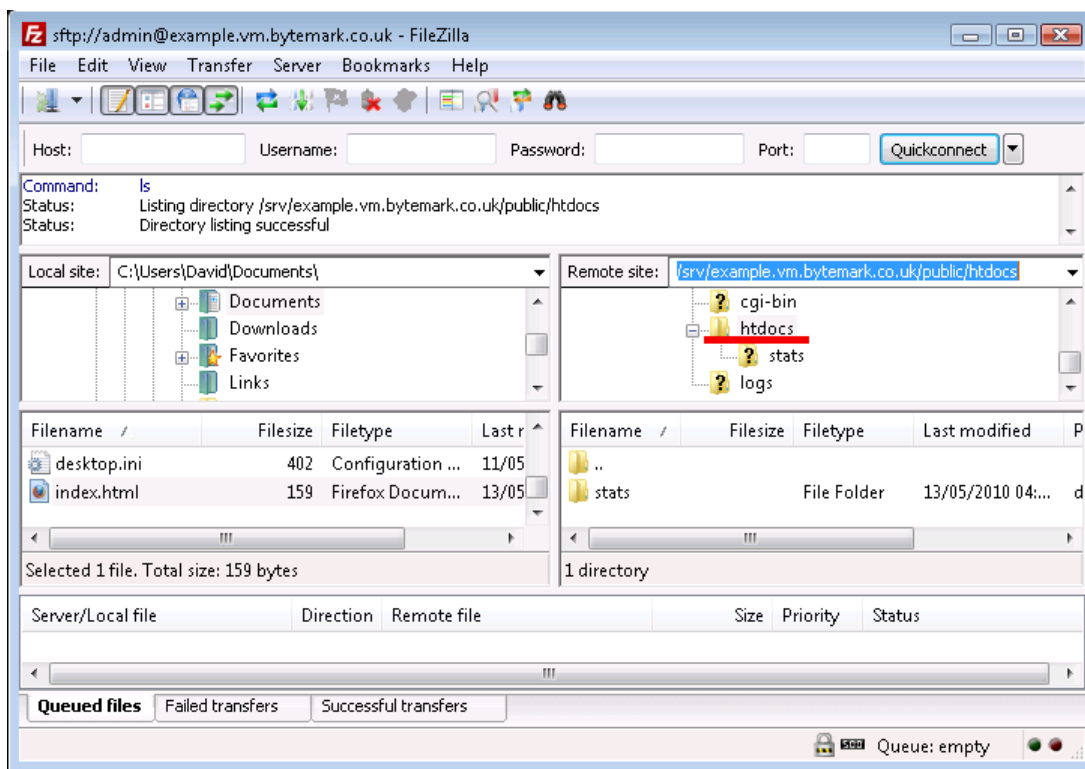
The procedure for replacing this default page with a new one is as follows.

1. Create a simple [HTML](#) file named `index.html`. It has been assumed that it has been saved in the directory called `My Documents/`.

2. Start up FileZilla and [connect to your server](#).
3. The file `index.html` should show up in the lower left-hand pane. In the right-hand pane the `/srv/` directory will be shown.

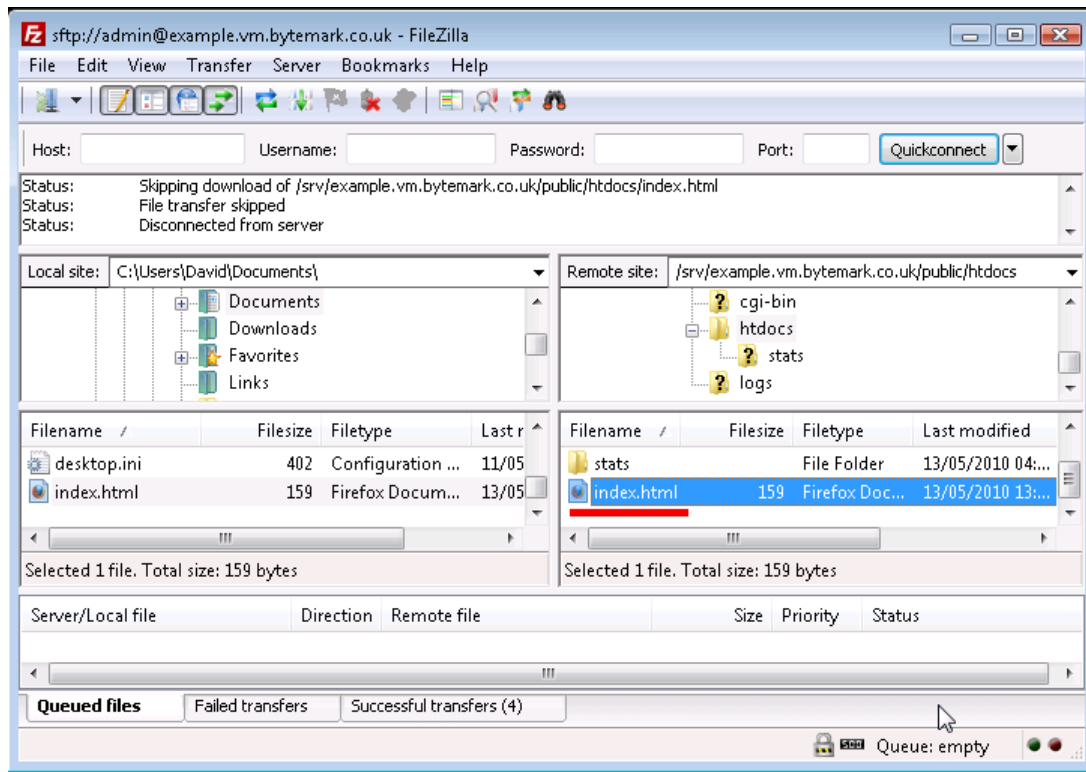


4. HTML files should be uploaded to the `public/htdocs/` directory. This can be found by revealing the contents of `srv/` by clicking the `+` to its left in the top-right hand pane.

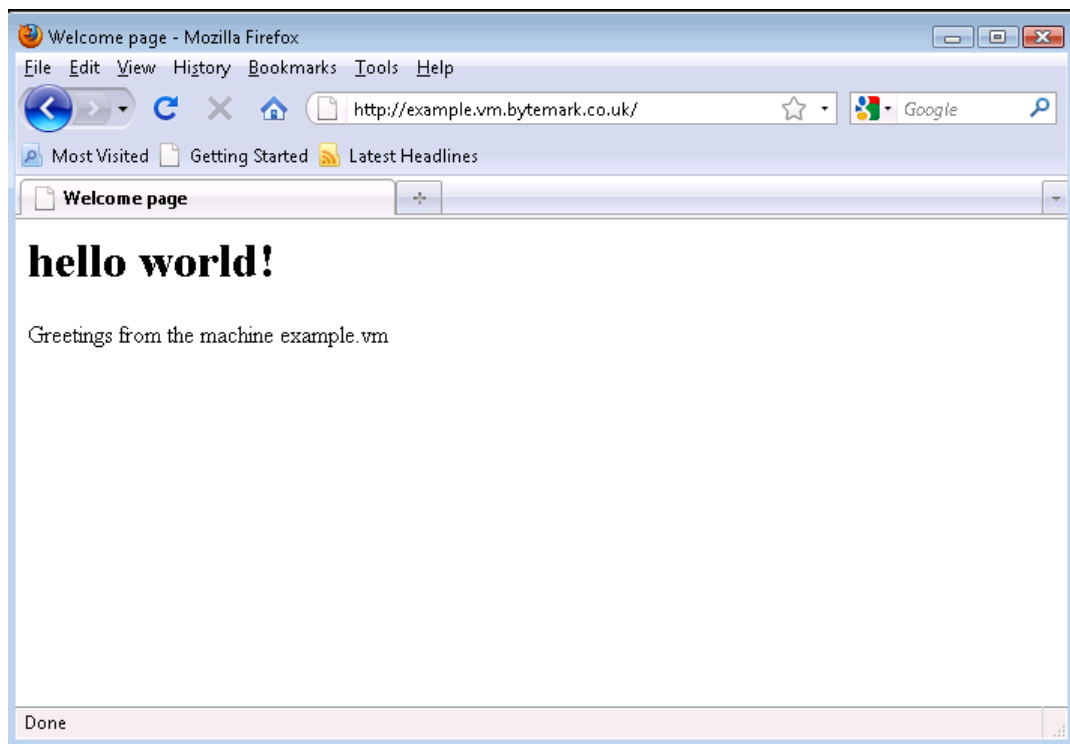


5. Now **right-click** on `index.html` in the lower left-hand pane, and select Upload from the menu. The file is

uploaded to the `htdocs/` directory on the server.



6. Refresh your web browser to see the result.



Note

This example shows uploading a web page written in HTML, called `index.html`. This file could also be written in *PHP*, in which case the file should be called `index.php`.

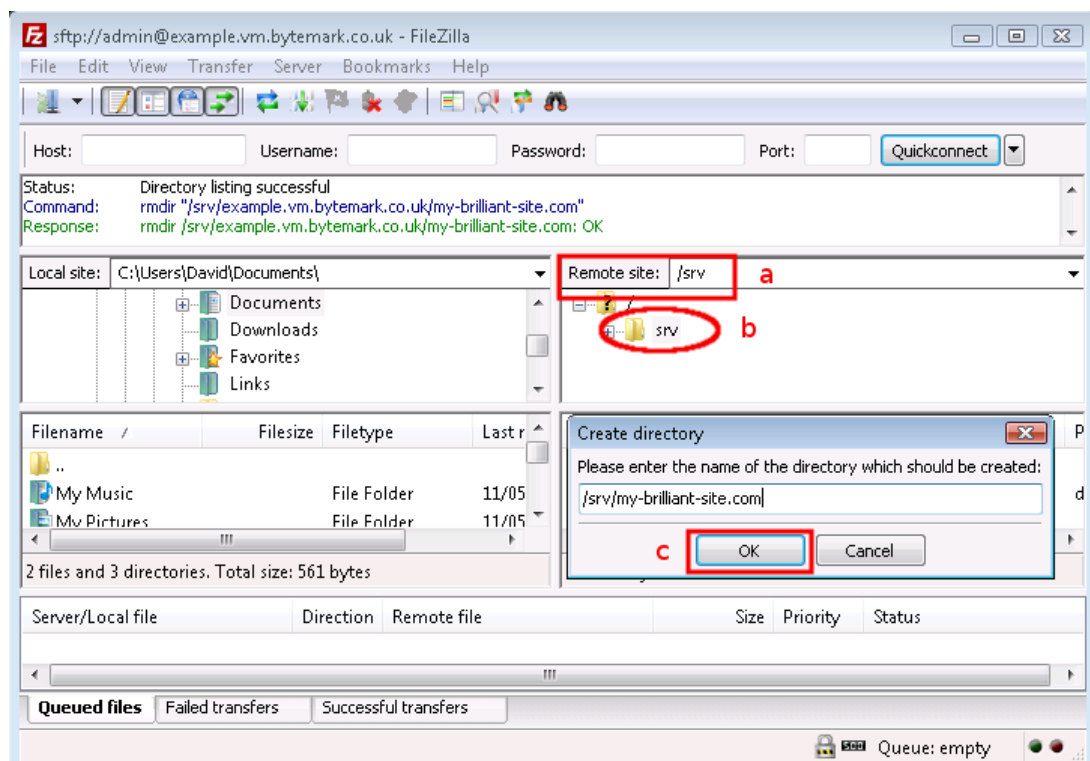
4.1 Hosting a web page using your own domain

The previous section dealt with setting up a web page using the default domain associated with the machine. A Symbiosis system can host many domains without any extra configuration. This section deals with configuring a second domain.

It has been assumed that both your server is hosted at Bytemark, and that this second domain is using the Bytemark name servers. If this is not the case, then Section 20.1 sets out the DNS records needed for the following procedure to work.

For the purposes of this tutorial, the domain **my-brilliant-site.com** is being hosted on the machine **example.vm.bytemark.co.**

1. With FileZilla connected to the server, make sure the Remote site text field is pointed at the `/srv/` directory **a**. Right click on the folder icon **b**.
2. From the right-click menu select Create directory and in the Create directory popup enter `/srv/my-brilliant-site.com`.



3. Click the OK button **c** to create the directory
4. Repeat this step to complete the domain tree with the directories `/srv/my-brilliant-site.com/public/` and `/srv/my-brilliant-site.com/public/htdocs/`
5. Create another `index.html` file.
6. Upload it as before, but this time into the `htdocs/` directory in the `my-brilliant-site.com/` directory tree.

Within a hour, the DNS records for **my-brilliant-site.com** will be generated and uploaded to the Bytemark domain name servers. Navigating to that site will then show our new index page.

At this point, the site will also be visible at both <http://my-brilliant-site.com> and <http://www.my-brilliant-site.com>. This is part of the Symbiosis setup; if different pages were required at **www.my-brilliant-site.com**, a separate directory tree should be created for **www.my-brilliant-site.com**, with a different content as needed.

4.2 Handling wildcard domains

As previously noted if there is a directory present upon the machine with the name `/srv/my-brilliant-site.com/public/htdocs/` the contents of that directory will be served for that domain, as well as any subdomains of that domain, for example:

- <http://my-brilliant-site.com/>
- <http://www.my-brilliant-site.com/>
- <http://test.my-brilliant-site.com/>
- <http://this.is.ridiculous.my-brilliant-site.com/>

By default, Symbiosis will [create DNS records](#) for only the raw domain and the `www` subdomain, as discussed in Section 4.1. To use anything other than these, you will need to create further DNS records for either the specific subdomains you wish to serve, or a Wildcard DNS record to match anything else. Please see Section 20.2.

If you wish to mandate a particular hostname for your sites that can be arranged via [mod_rewrite](#) as discussed in Section 15.6.

4.3 Testing a new domain

It is possible to test the content associated with a new domain before it has been registered or had any DNS configuration done. This is done using the **testing** prefix.

If your machine is not hosted at Bytemark, or your machine name does not end in **bytemark.co.uk** then a wild-card DNS record is needed for this to work. This is discussed in Section 20.2.

For example, to view the site **my-brilliant-site.com** which is hosted on the machine **example.vm.bytemark.co.uk**, simply head to <http://my-brilliant-site.com.testing.example.vm.bytemark.co.uk/>.

This testing URL is immediately available following the upload of the `index.html` or `index.php` files.

Note that there is no **www** at the start of the testing URL.



Note

This facility does not play well with certain directives that can be used in Apache `htaccess` files, especially rewrite rules.

Chapter 5

Configuring email



Before you start this chapter

1. Connect to your server via SFTP (see Chapter 3).
-

This chapter deals with configuring email for a domain, namely setting up mailboxes to receive email. The Symbiosis system makes this very simple, as the process of creating a new mailbox, or email account, is a simple matter of creating a few files and directories.

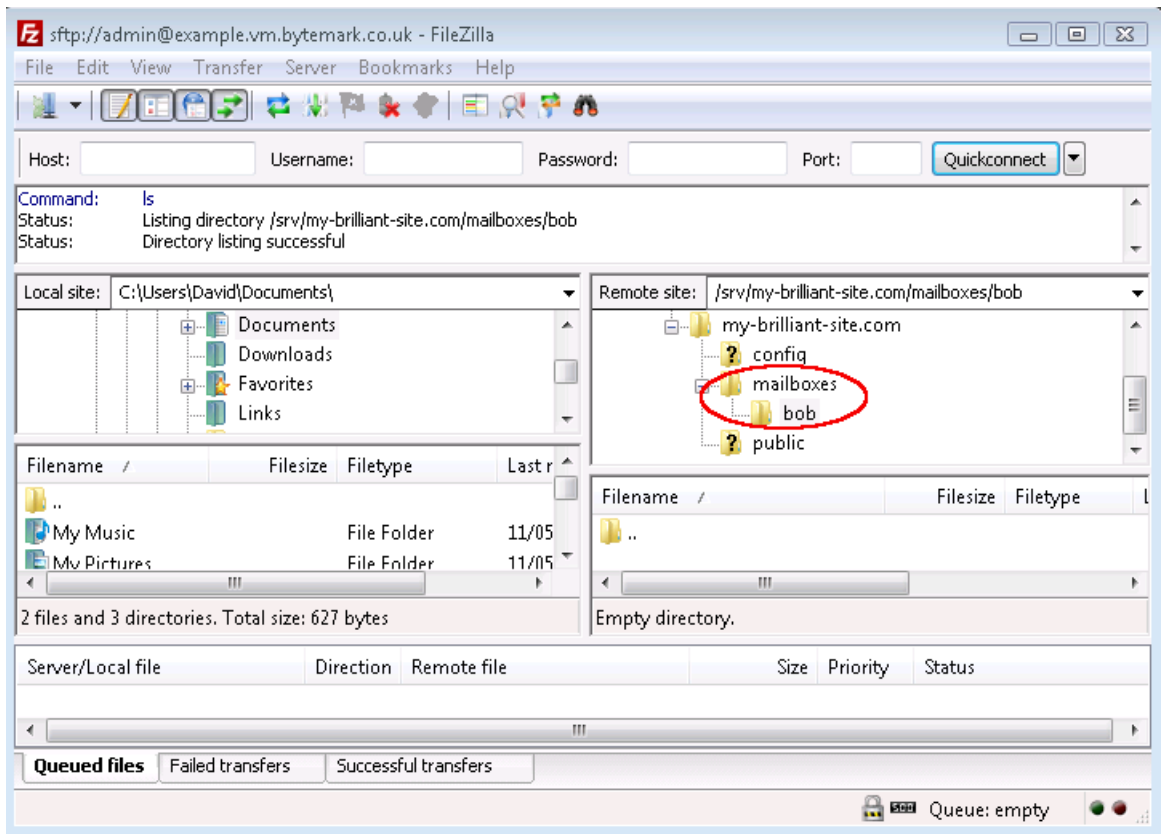
As with our previous examples we'll be using the **my-brilliant-site.com** domain for demonstration purposes, but you should substitute your own domain.

Again for example purposes we'll be demonstrating the creation of a new email account, for the user "bob", which will correspond to the email address bob@my-brilliant-site.com - you should change the name "bob" to the username(s) you desire.

5.1 Creating a new mailbox

It has been assumed that the first few steps in Section 4.1 have been followed, i.e. that a directory has been created under `/srv/` for the domain **my-brilliant-site.com**.

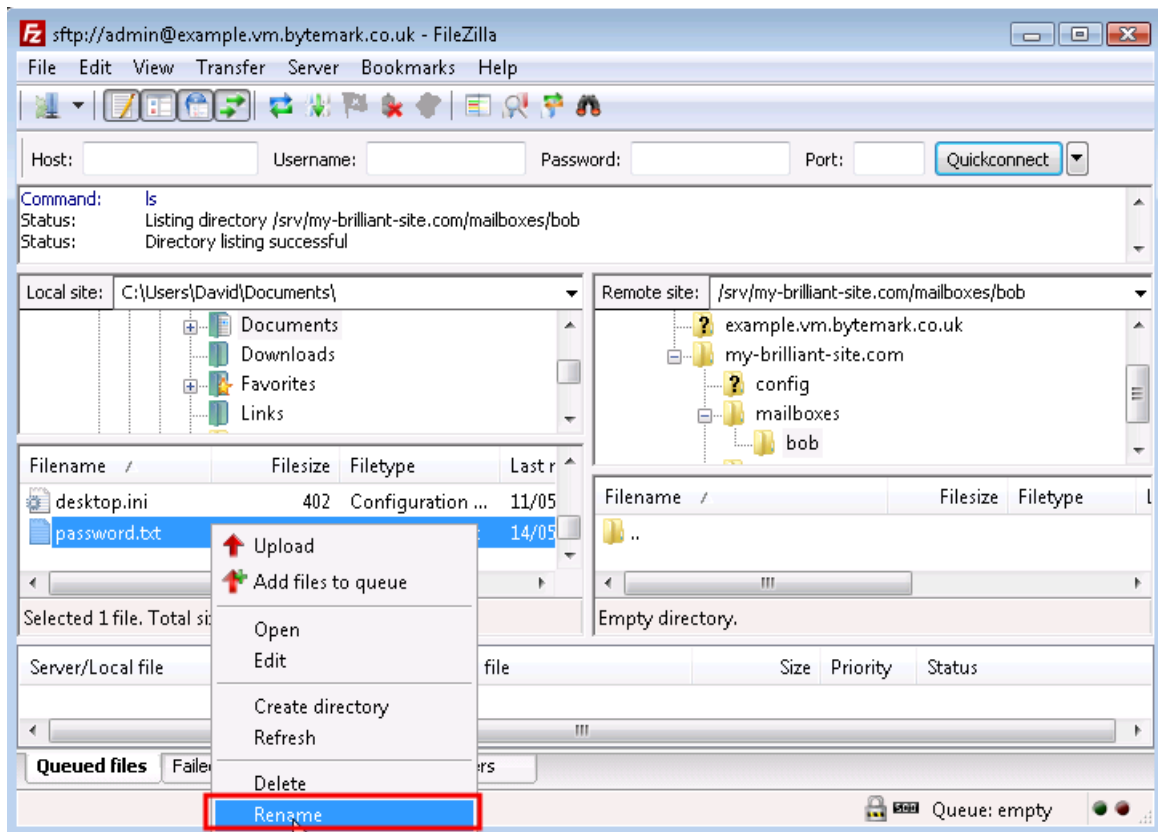
1. Start FileZilla and connect to your machine.
2. Then **right click** on the `/srv/my-brilliant-site.com/` directory and select Create directory from the menu. Set the new directory name to be `mailboxes` and press the OK button.
3. Repeat this step to create the directory `mailboxes/bob/` which makes a mailbox for the address bob@my-brilliant-site.com.



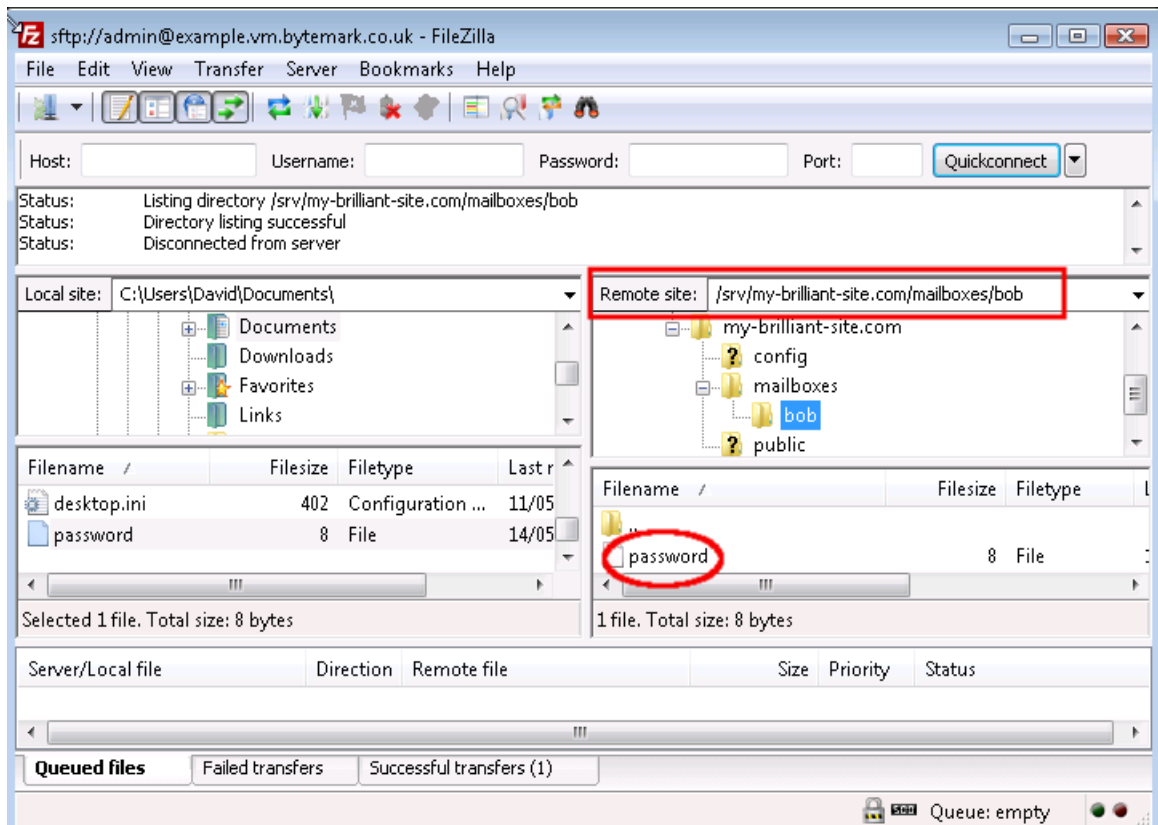
Important

Please note that lowercase is required for mailbox usernames!

- Use a text editor such as Notepad to create a file `password` on your desktop machine which contains a secure password.
- Under **Windows** a `.txt` extension will be added to the filename which is not wanted. So before you upload the file use FileZilla to rename it from `password.txt` to `password`. That is done by clicking with the right mouse button on the file in the lower right hand pane, and selecting Rename from the menu that appears.



- To upload, **right click** on the filename and select Upload from the menu, making sure that the directory /srv/my-brilliant-site.com/mailboxes/bob/ is shown in the Remote site: text area.



That is all that is needed to set up a new mailbox. To test we can immediately use the webmail application, SquirrelMail, supplied with Symbiosis.

5.1.1 Forwarding Email

If you would prefer to have emails to a new address sent on to, create a file named `/srv/my-brilliant-site.com/mailboxes/bob/forward`. In this file just enter the name of the account that mail should be forwarded to; this might be something like dave@example.com.

This file can do many other things than just forwarding email, as explained in its [reference section](#).

5.1.2 Filtering Email

Users can configure their own server-side filtering rules using the ManageSieve protocol. This is a [standard protocol](#) which can be used to create [Sieve](#) scripts which are used to filter email into folders on behalf of the user.

Sieve is a powerful language that can be used to achieve

- mail deliveries into specific folders,
- vacation messages,
- flagging or highlighting mail.

ManageSieve clients are available for [several mail clients](#), the most common of which is [Mozilla Thunderbird](#) via its [Sieve add-on](#).

5.1.3 Scanning email to prevent spam and viruses

Symbiosis comes with in-built virus and spam detection, however it is not enabled by default. There are two principal aspects to this, namely

- The use of SpamAssassin to scan each email to determine if a message is unwanted;
- The use of ClamAV to detect viruses in emails.

Each of these is configured separately, on a per-domain basis, giving choice as to which preventative measures are applied to your email.

5.1.3.1 Using SpamAssassin to detect and reject or tag spam

Email can be rejected or tagged, based on its spam score determined by SpamAssassin. This is not enabled by default, but can be enabled in much the same way as the blacklists above.

The default action is to reject, i.e. bounce, email that is determined by SpamAssassin to be spam. This can be changed to accept all email, but tag it with a header field to allow users to filter it themselves.

Once scanned, a message will have three extra headers added, for example for an innocent message might have a score of 1.2. In which case it has the following headers added.

```
X-Spam-Score: 1.2
X-Spam-Bar: +
X-Spam-Status: innocent
```


A more guilty message might have a score of 4.2, but note that this message is still deemed "innocent", since it scored less than 5, which is the default threshold at which SpamAssassin deems messages to be spam.

If a message scores more than 5, and the domain has "tag" in its `config/antispam`, then the `X-Spam-Status` header will be set to `spam`.

To enable spam scanning:

1. Connect to your machine using FileZilla
2. On the remote directory tree, navigate to `/srv/my-brilliant-site.com/config/`.
3. On your local machine create a file called `antispam`. If you want to reject email, i.e. bounce email, that is classified as spam, this file should be empty. If you'd rather accept all email, but tag it as spam, this file should contain the word `tag`.
4. Having created the file, right click on it and select upload to transfer it to the remote system. Make sure that the remote file has the correct name, i.e. no extra `.txt` extension.

5.1.3.2 Using ClamAV to detect and reject, or tag, emails with viruses

ClamAV is activated in a similar way to SpamAssassin. It can also be set to tag or reject. Again, a header is added to message that has been scanned. In this case the header that is added is `X-Anti-Virus`. This is set to `clean` if no viruses were detected, otherwise `infected`, but only if configured to tag.

To enable virus scanning:

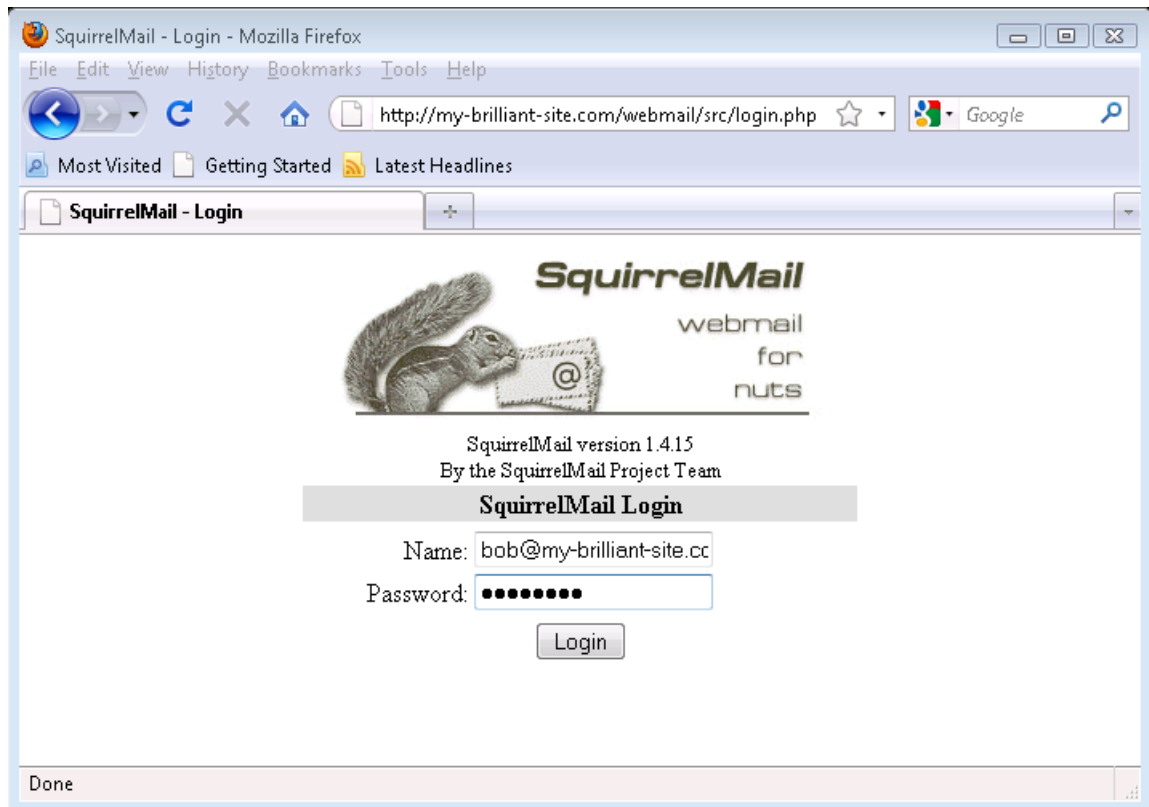
1. Connect to your machine using FileZilla
2. On the remote directory tree, navigate to `/srv/my-brilliant-site.com/config/`.
3. On your local machine create a file called `antivirus`. If you want to reject email, i.e. bounce email that has viruses in, this file should be empty. If you'd rather accept all email, but tag it to show that it has a virus in, this file should contain the word `tag`.
4. Having created the file, right click on it and select upload to transfer it to the remote system. Make sure that the remote file has the correct name, i.e. no extra `.txt` extension.

5.2 Testing a new mailbox, via webmail

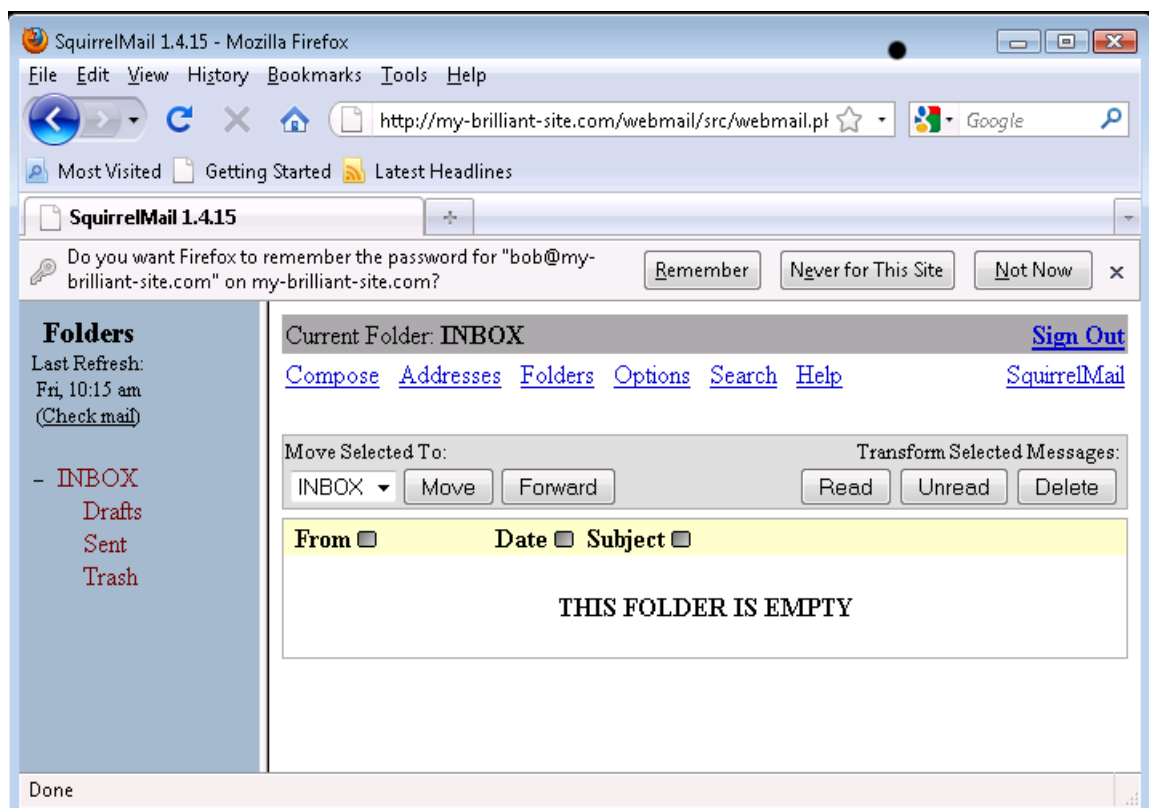
Although most users will prefer to receive and write their emails using a dedicated client (such as ThunderBird, or Microsoft Outlook) the Symbiosis system includes a mail client you can access with nothing more than a web-browser.

This section briefly documents using the Squirrelmail webmail system.

1. To log in to webmail, start your browser and head to <http://my-brilliant-site.com/webmail>.
2. Enter your email address in the Name field, and your password in the Password field.



3. Click the Login button, and assuming the Name and Password fields were correct, you will be presented with your Inbox where you can read and send email.



5.3 Configuring email clients

Configuring email clients to work with Symbiosis is covered in [Appendix A](#).

Chapter 6

Chat server (XMPP/Jabber) Configuration



Before you start this chapter

1. Connect to your server over [SFTP](#) using FileZilla (see Chapter 3).
 2. Configure at least one mailbox for your domain (see Chapter 5).
-

[XMPP](#) is now available for every user with a mail account for a domain. It is straightforward to add to any domain.

1. Create an empty file for `/srv/my-brilliant-site.com/config/xmpp`.

Once this file is in place, all email mailboxes for that domain with a password set will be able to log in to XMPP using their email address as the username.

Chapter 7

Setting up per-domain FTP Access



Before you start this chapter

1. Connect to your server over *SFTP* using FileZilla (see Chapter 3).
 2. Set up a website (see Chapter 4).
-

Fast forward to the scenario where you have a web hosting client who has designed their own site and would like to upload it themselves. However it is not necessary to grant them access to all domains on the machine, or even the config or mailboxes section of their own domain.

This is typical for a shared hosting client, and the solution is to give them *FTP* access. This limits them to the files inside the `public/` directory, i.e. only those associated with the website.

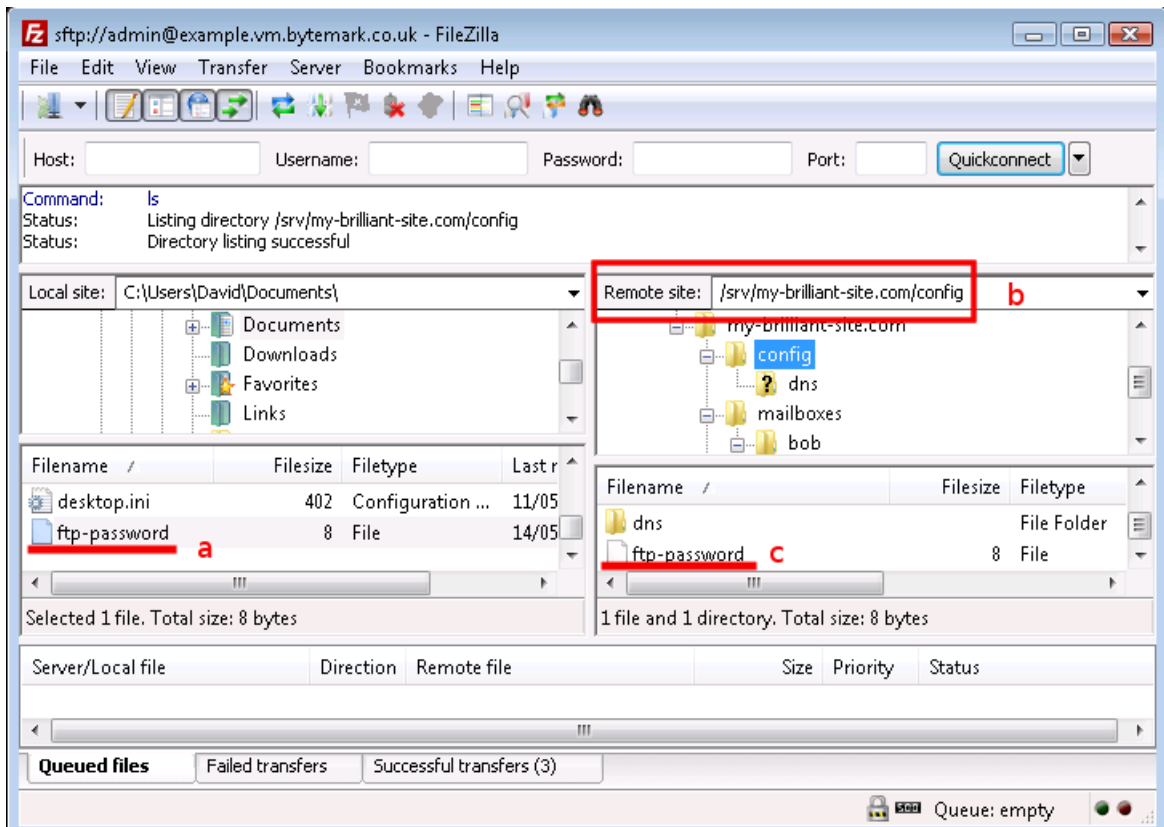


Warning

Please be aware that despite being limited to the `public/` directory when logging in over FTP, it is trivial for the user to read files elsewhere on the filesystem, for example by using specially written PHP scripts. It is advisable to grant access only to trusted people.

In this example, access to the content of the **my-brilliant-site.com** site is being given to another user, but they are only to have access to `/srv/my-brilliant-site.com/public/`. To set this up, an FTP password is being created.

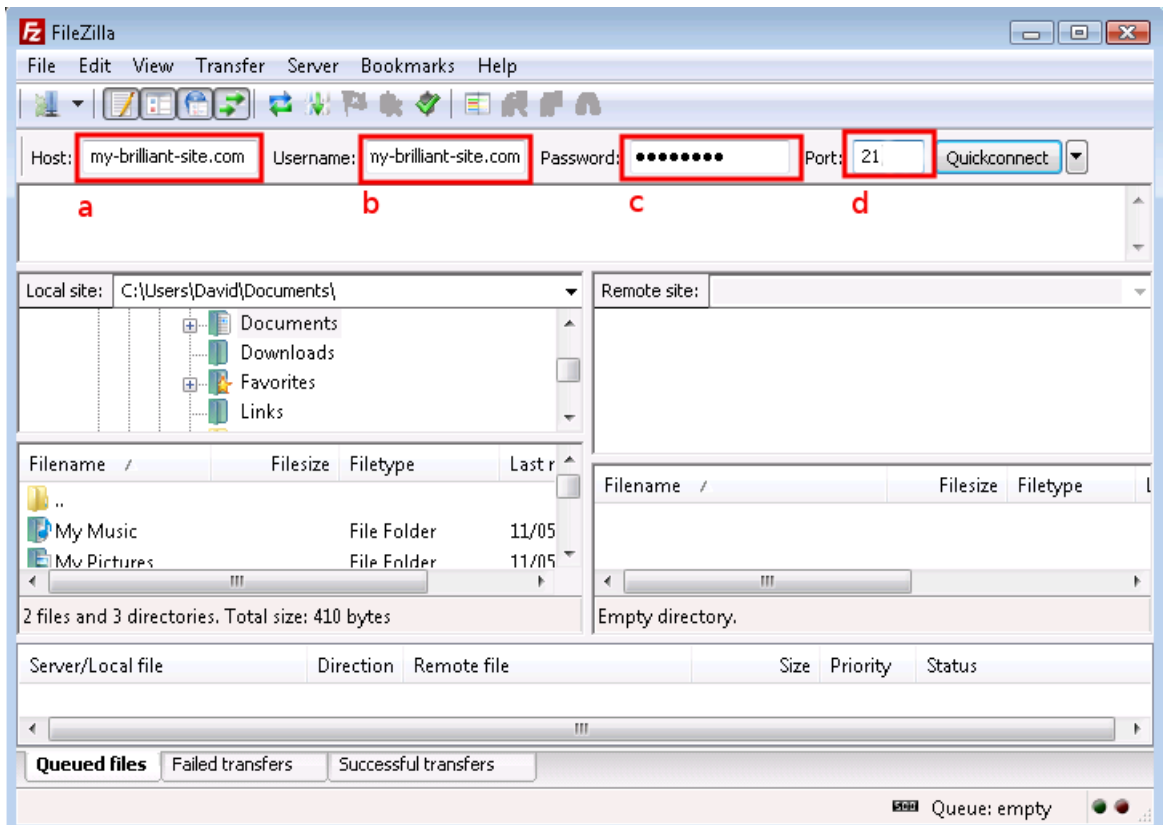
1. Connect to your machine using FileZilla.
2. Navigate to `/srv/my-brilliant-site.com/config/`
3. Create a file `ftp-password` **a** that contains a secure password your shared hosting client will use, ensure that the `config/` directory is selected **b** and upload the file, **c**. Make sure that there is no `txt` extension on this file.



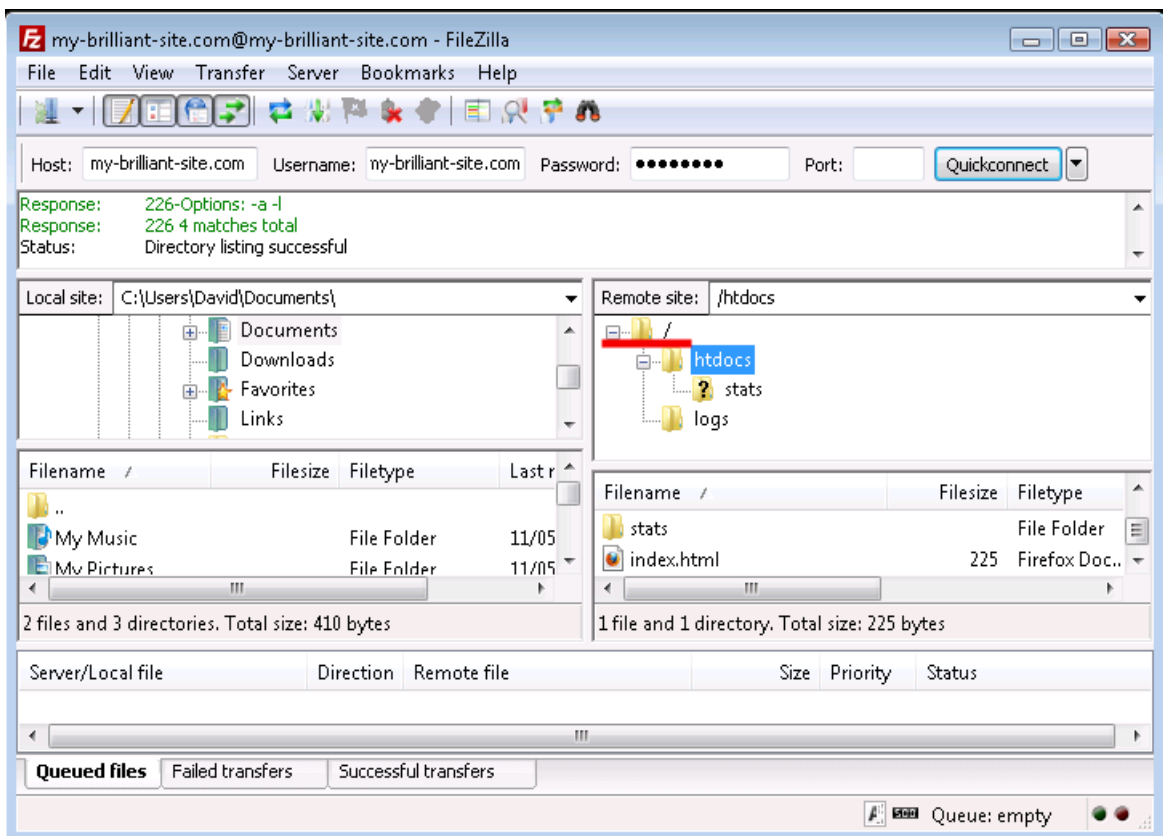
Now that is all that is needed. Access to the machine can now be granted over FTP using the username **my-brilliant-site.com** and the password being the contents of `/srv/my-brilliant-site.com/config/ftp-password`.

We will now test the connection to make sure it works, also using FileZilla, since it can be used to connect via FTP as well as SFTP.

1. Make sure FileZilla has disconnected from the machine.
2. The host **a** and the user **b** are both the domain name, in this case **my-brilliant-site.com**. The password **c** is the contents of the `ftp-password` file and for FTP the port number must be set to 21, **d**.



- Once you connect you'll notice that you only have access to directories beneath the `public/` directory (here represented as `/`) of the `my-brilliant-site.com/` directory tree, which is all you'd need if your role was limited to maintaining or setting up a web site.



7.1 Setting FTP quotas

It is possible to limit the amount of data that can be kept in a domain's `public/` directory using an FTP quota. This is done by creating a file inside the domain's `config/` directory called `ftp-quota`. Inside this file should be a number of bytes at which the quota is set.

The number can have a suffix of `k`, `M`, `G`, or `T` representing kilo-, mega-, giga-, or terabytes respectively.

For example, to prevent the author of **my-brilliant-site.com** from putting more than 150MB inside their `public/` directory, create a file called `/srv/my-brilliant-site.com/config/ftp-quota` with the contents `150M`. This will limit their space usage to 150,000,000 bytes.

**Note**

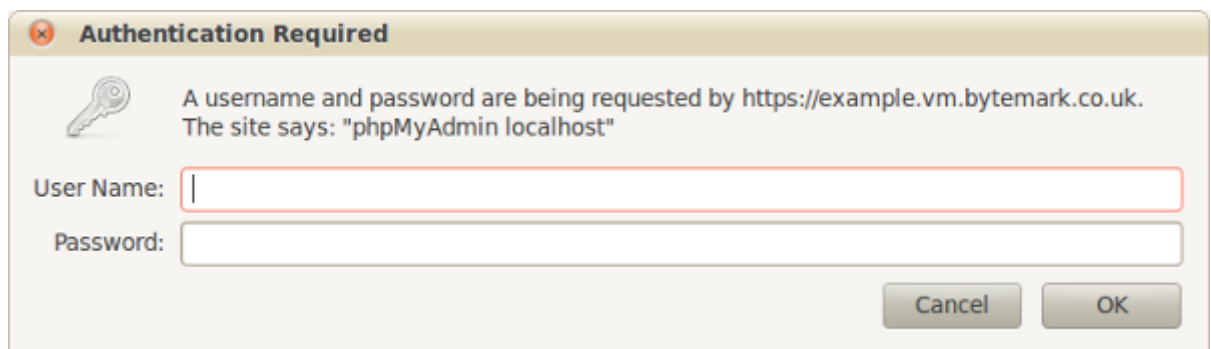
Please be aware that the FTP quota will include all log data from the web-server in `public/logs/`, as well as the automatically generated statistics in `public/htdocs/stats/`.

Chapter 8

Managing the MySQL database

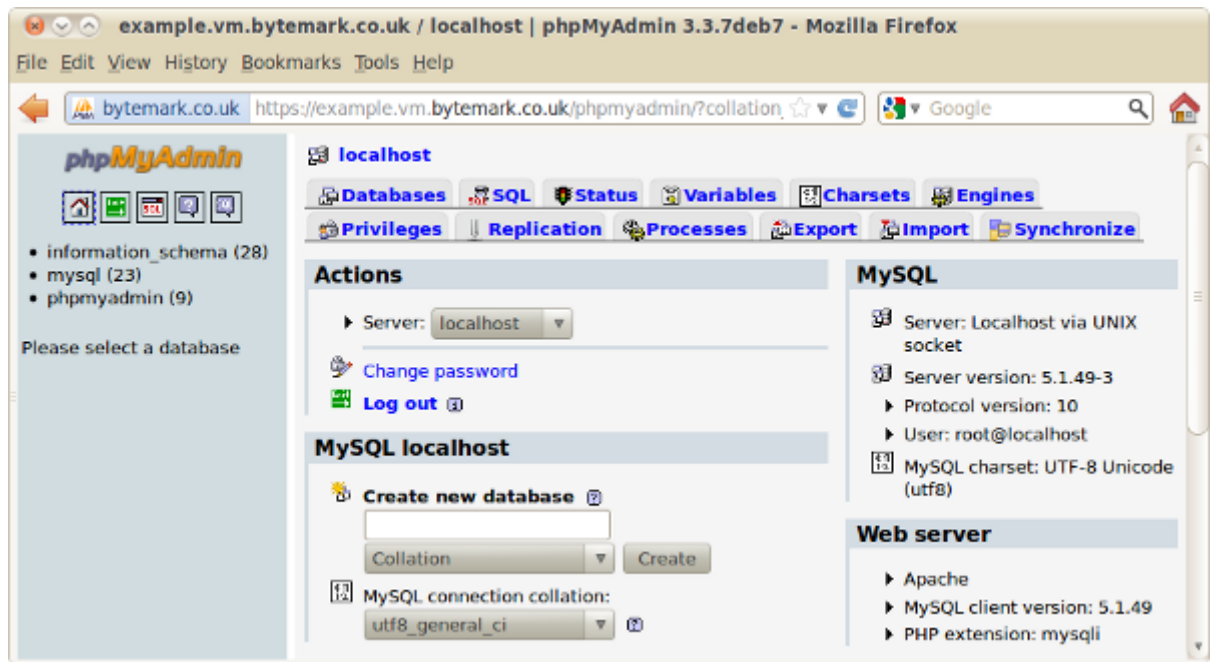
The Symbiosis system comes with the MySQL database installed and running. It can be managed by use of the phpMyAdmin program. The following instructions show how to connect to the database on the machine **example.vm.bytemark.co.uk**.

1. Start your web browser.
2. Navigate to <http://example.vm.bytemark.co.uk/phpmyadmin/> and enter the authentication details. The user is **root** and the password is the same as that of the **admin** user.



The image shows a standard web browser authentication dialog box. The title bar reads "Authentication Required". Inside the dialog, there is a key icon on the left and a message on the right: "A username and password are being requested by https://example.vm.bytemark.co.uk. The site says: 'phpMyAdmin localhost'". Below the message are two input fields: "User Name:" followed by a text box containing a vertical cursor, and "Password:" followed by a password field. At the bottom right of the dialog are two buttons: "Cancel" and "OK".

3. Press the OK button to be log in.



From here new databases and database users can be created as needed. phpMyAdmin is further documented on its [home page](#).

Chapter 9

Scheduled tasks

Each domain has the ability to run its own scheduled tasks via a file known as a **crontab**. This file enables jobs to be run on at specific times on specific days.

The format is the same as the well-known **crontab** file used on many Linux systems.

A domain's crontab is found at `config/crontab`. For example, the crontab for **my-brilliant-site.com** would be found at `/srv/my-brilliant-site.com/config/crontab`.

9.1 The crontab format

The file is a list of jobs, one per line. Each line specifies first the times and days at which a job should run, followed by the command to run.

The first five fields, which are separated by spaces, specify the time and date at which the job should run. The rest of the line is interpreted as the command.

Field	Allowed values
minute	0-59
hour	0-23
day of month	1-31
month	1-12 (or names, see below)
day of week	0-7 (0 or 7 is Sunday, or use names)

In addition an asterisk can be used to indicate for every allowed value. For example, to execute the command **echo Hello Dave**. at 18:40 every day, the crontab line would read as follows.

```
40 18 * * * echo Hello Dave.
```

Three-letter names can also be specified for use in instead of numbers for days of the week and months.

Weekdays Sun, Mon, Tue, Wed, Thu, Fri, Sat

Months Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec

Any output generated by a command will be sent to the **root** account, unless specified otherwise. If no output is generated, no email will be sent.

The fields can be specified in the following ways:

- As a comma separated list, e.g. 1, 2, 3, 6
- As a range, e.g. 1-3 would mean 1, 2, 3
- As a range with a step, e.g. 0-30/2, would mean 0, 2, 4, 6 and so on until 30.
- Or any combination of the above three, e.g. 1, 2, 10-12, 20-24/2 would mean 1, 2, 10, 11, 12, 20, 22, 24.

Ranges can also be specified across "boundaries". For example 22-2 in the hour field will be interpreted as 22, 23, 0, 1, 2; Nov-Feb in the month field will mean 11, 12, 1, 2.

There is also a selection of shortcuts available:

- @hourly—every hour, on the hour,
- @daily or @midnight—every day at midnight,
- @weekly—every week at midnight on Sunday,
- @monthly—every month, at midnight on the first day of the month,
- @yearly or @annually—every year, at midnight on 1st January.

The full crontab format is explained in more detail in the [crontab \(5\)](#) manual page.

9.2 Mailing the output

The output can be emailed to any recipient by specifying the **MAILTO** parameter at the top of the file.

For example, we would like to mail any output from our commands to bob@my-brilliant-site.com.

```
#
# send any output to Bob
#
MAILTO=bob@my-brilliant-site.com
#
# run at 9am every Monday - Friday
#
0 9 * * 1-5 wget http://www.my-brilliant-site.com/cron.php
```

If **MAILTO** is not set, or no recipient is specified, then the output will be sent to the domain directory's owner, e.g. if `/srv/my-brilliant-site.com/` were owned by `admin`, the output would get sent to admin@example.vm.bytemark.com.

Chapter 10

Automated backups

The automated backup system contained in the Symbiosis system protects against accidental deletion or corruption of file. These backups are designed to run once per day and archive the contents of a number of important system directories.

Having backups stored locally is not sufficient to provide real protection from accidents though, as they might be removed or deleted. Therefore your local backups should be archived to a remote machine.

For **Bytemark customers** the backup script is configured to attempt to do this, using the remote backup space provided by Bytemark, as documented upon the [Bytemark support site](#). This remote backup space is provided for legacy VMs and dedicated hosts only.

Your system will maintain full backups of the following locations:

- /etc/
- /home/
- /root/
- /srv/
- /usr/local/
- /var/mail/
- /var/lib/
- /var/backups/mysql/
- /var/backups/postgresql/

Additionally each MySQL and PostgreSQL database you have upon your system will also be exported and backed up.

The backup system uses the backup2l program, which is configured to backup the files in the above locations into the directory `/var/backups/localhost/`. For more information about backup2l, please refer to its [manual page](#).

10.1 Offsite backups

As mentioned above, the backup script will attempt to ensure that your local backups are uploaded to a remote server, to protect against data loss if your system fails catastrophically.

For **Bytemark customers** using legacy VMs or dedicated servers this location *should* be determined automatically.

If this process fails, or you are not a Bytemark customer, you can specify the correct location in `/etc/symbiosis/dns.d/backup.name`. This should be a fully-specified rsync path.

10.2 Backup reporting

Every day, when runs it generates output saying what has been backed up, and if there were any errors during the backup process. This email will get sent to the **root** account of the local hostname, eg `"root@example.vm.bytemark.co.uk"`.



Note

It's important to realise that the automated backups, especially their transfer to the remote backup space, is done on a best effort basis. You should carefully check the backup2l report for errors and from time to time practise recovering files at random from the remote server, to ensure that there are recoverable backups.

Chapter 11

Keeping Your System Secure

This chapter describes the features we provide to help increase your system security, and offer tips and suggestions on what you can do to help ensure your system remains secure.

11.1 Checking system notifications

The Symbiosis system is comprised of many components, each working together to deliver a complete solution to your hosting needs. Different systems and components of your server will generate email notifications to alert you of important events and warnings. It is important that such emails are read.

By default all system-generated emails will be delivered to the root user of your primary domain. (This is the first domain which is configured when your machine is setup, and will probably be a name such as **example.vm.bytemark.co.uk**.)

Rather than make it mandatory that you read the root mailbox it is suggested that you [configure email forwarding](#) such that mail sent to **root@example.vm.bytemark.co.uk** is delivered to your personal email address.

11.2 Avoiding weak passwords

A common means of compromising machines what is called a "dictionary attack", this involves a remote user (or computer) trying to connect to a server with a collection of thousands of usernames and passwords.

This dictionary of usernames and passwords will include common choices such as a username of "test" and a password of "test", along with many other less-likely looking candidates. The Symbiosis Firewall has a blacklisting program that detects attacks via various protocols, including SSH, and configures the firewall to block further connections. This is documented in Section [19.6](#).

**Note**

This important security measure can catch you out if you repeatedly attempt to access the server using incorrect credentials, as you're likely to find your own IP address becomes blacklisted. See Section [25.2](#) for help with this situation.

The best defence is to ensure that when you add users, or change system passwords, that you never ever choose simple passwords which might be liable to be guessed, or included in an attackers' dictionary.

There is a regular test on all the passwords used to access email and FTP under Symbiosis, the output of which will get sent to the **root** email account. Please see the note in [earlier in this chapter](#) regarding email notifications.

11.3 Keep your software current

Over time security bugs can be found in software packages, and if such a problem is discovered in a package you're using then your machine is at risk until it has been updated.

The Symbiosis system is configured to automatically download and install appropriate security updates to the packages in the base operating system and from the Symbiosis repository itself.

However if you've chosen to install additional applications such as Wordpress you **must** ensure that you look for updates regularly. Often this can be done by subscribing to the application's announcements mailing list.

11.4 Local security considerations

When granting FTP access to your machine, it is important to bear in mind that the person who uses that login can trivially access other files on the system. Various methods could be used, including uploading PHP or CGI scripts.

There are ways to mitigate the effect of this access, including setting permissions such that sensitive files are not world-readable, and hashing passwords. However the safest way to manage this problem is to ensure that only trusted users are given FTP access.

Chapter 12

Connecting to your server via SSH

So far we have looked at making connections and transfer files using the [SFTP](#) protocol. For certain operations it is also necessary to connect using the [SSH](#) protocol. This is used to gain command-line access to the machine.

There are two connection programs are documented, depending on your desktop environment, and your preference.

12.1 Why SSH access?

So far connecting using [SFTP](#) has been documented. This is used to manage files on the machine. From time to time it can be necessary to have to run commands on the machine itself. This is where SSH comes in.

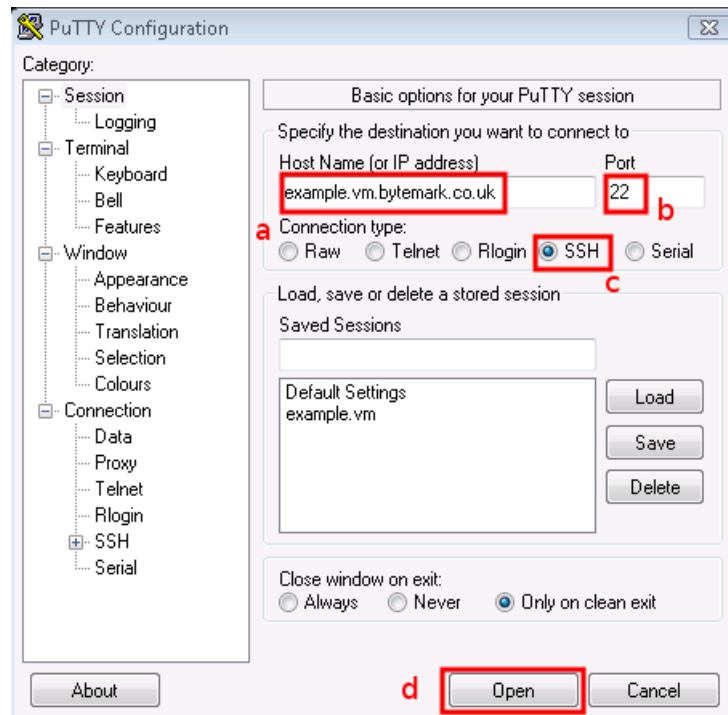
SSH allows *shell* access to a machine, which provides the ability to run commands directly on that machine. Shell access is the equivalent of the MS-DOS or **cmd** prompt on Windows PCs, or the terminal on machines running Mac OS X or Linux. SSH is an encrypted protocol, like SFTP, ensuring that all commands and passwords pass between your computer and the server are protected against eavesdroppers.

For Bytemark customers, SSH is also used to access the [Console Shell](#) of the machine.

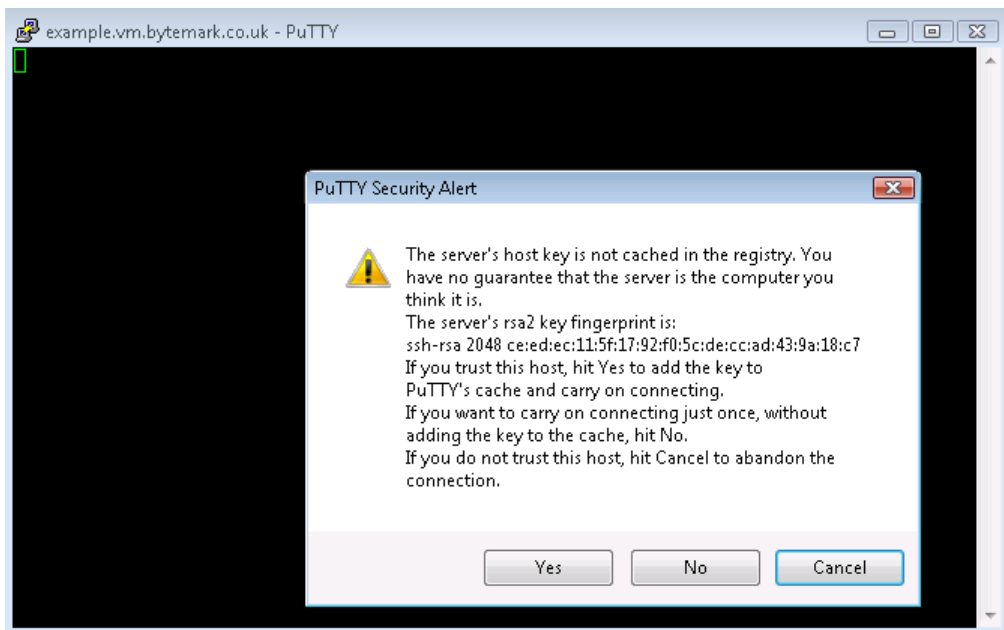
12.2 Using PuTTY to connect via SSH

PuTTY is a free and open-source SSH application, and available for download from its [homepage](#). It is available for both Windows and Linux desktop machines.

1. Start PuTTY; Under Windows you may get a Security Warning—if so you need to click the Run button.
2. Enter your server's name in the Host Name field **a**; the Port field **b** should read 22, and the SSH radio button **c**. Click Open **d** to start the connection.



3. The PuTTY command window will open; the first time you do this, as with FileZilla, you get a Security Alert, warning that the key is untrusted. It is safe to say Yes the first time you connect to your machine.



4. At the Login as prompt type **admin** and press enter. Then the `[label]admin@example.vm.bytemark.co.uk's password|` prompt appears. Enter the password followed by enter. Nothing is displayed when the password is entered.
5. You'll log in and get presented with the `admin@example:~$` prompt, ready to accept commands.



```
example.vm.bytemark.co.uk - PuTTY
login as: admin
admin@example.vm.bytemark.co.uk's password:
Linux example.vm.bytemark.co.uk 2.6.32.11-kvm-i386-20100423 #1 SMP Fri Apr 23 10
:10:21 BST 2010 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
admin@example:~$
```

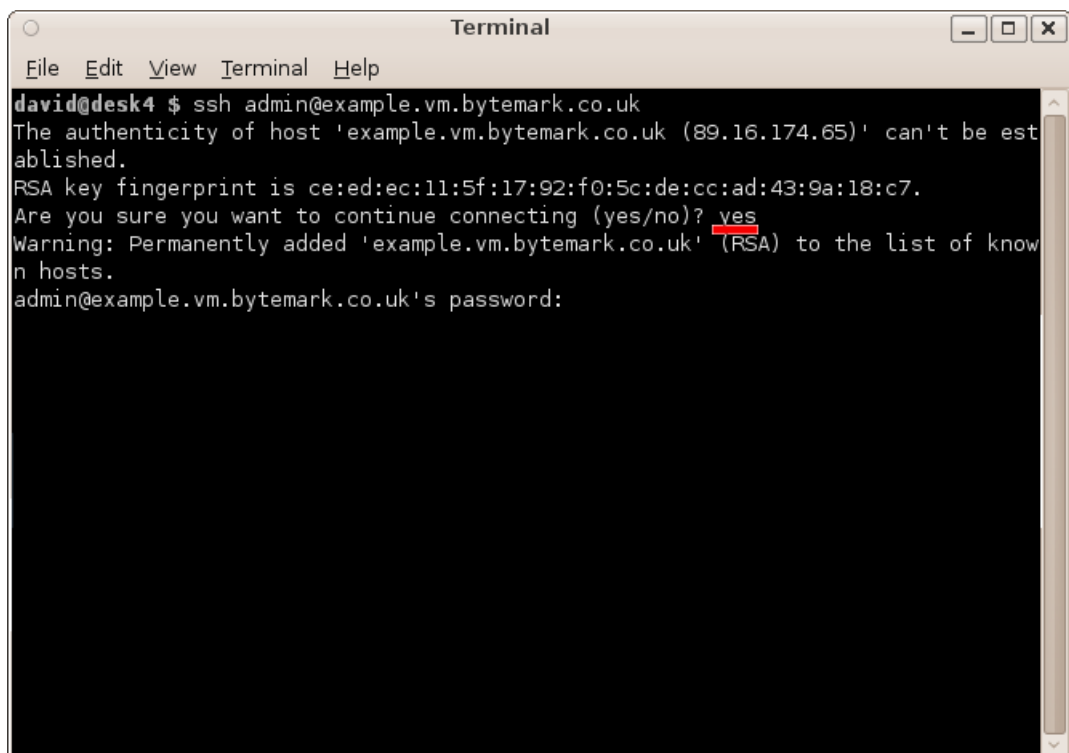
**Tip**

When the machine is setup, the **root** and **admin**, as well as the mysql database **root** passwords are all the same.

12.3 Using a terminal program to connect via SSH

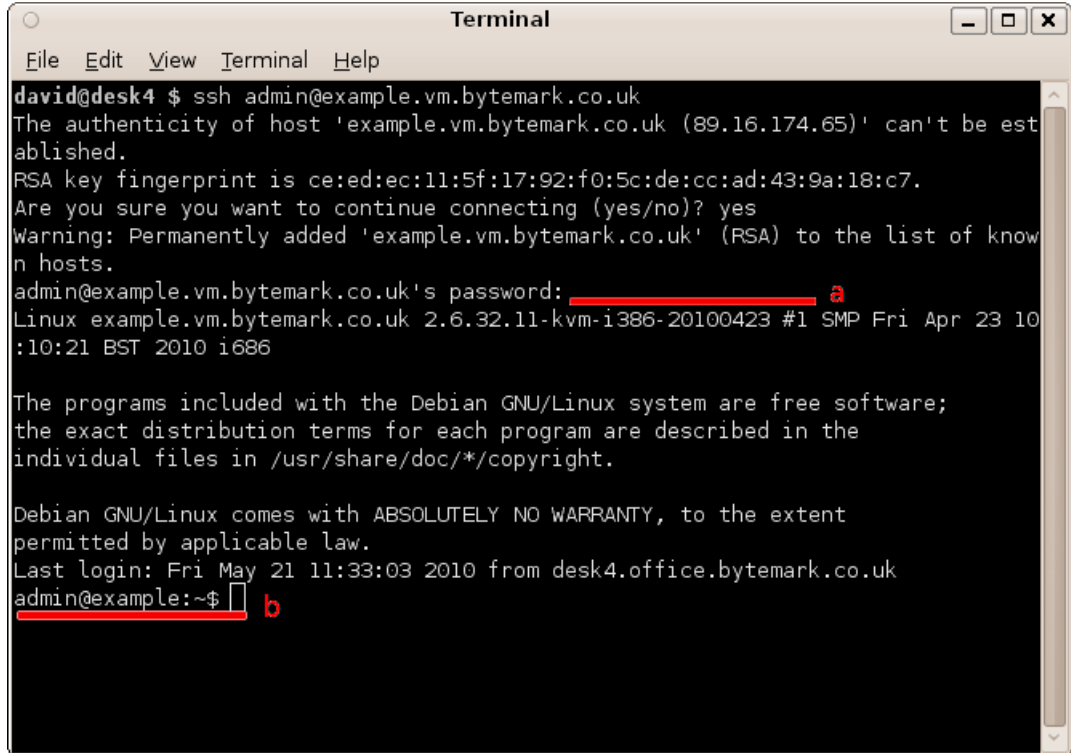
Both Linux and Mac OS X desktop machines tend to come with the **ssh** command available.

1. Open a terminal emulator and enter the command **ssh admin@example.vm.bytemark.co.uk**.



```
Terminal
File Edit View Terminal Help
david@desk4 $ ssh admin@example.vm.bytemark.co.uk
The authenticity of host 'example.vm.bytemark.co.uk (89.16.174.65)' can't be est
ablished.
RSA key fingerprint is ce:ed:ec:11:5f:17:92:f0:5c:de:cc:ad:43:9a:18:c7.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'example.vm.bytemark.co.uk' (RSA) to the list of know
n hosts.
admin@example.vm.bytemark.co.uk's password:
```

- The first time you connect to your machine a warning message about the authenticity of the host will appear. The **first time** you connect to the machine, we can assume the host is authentic. So enter "yes", to accept the key and to continue connecting.
- The connection has been made as user **admin**, so enter the admin user password , **a**; that does not get echoed to the screen. At the end of the dialogue you see the prompt, `admin@example:~$`, **b**.

A terminal window titled "Terminal" with a menu bar (File, Edit, View, Terminal, Help) and window control buttons. The terminal output shows an SSH session from a host named "david@desk4". The user runs the command "ssh admin@example.vm.bytemark.co.uk". The terminal displays a warning about the host's authenticity, the RSA key fingerprint, and a prompt to continue connecting. The user responds "yes". A warning message states that the host has been permanently added to the list of known hosts. The terminal then prompts for the password of the "admin" user. The password "a" is entered and is not echoed. The terminal shows the system version "Linux example.vm.bytemark.co.uk 2.6.32.11-kvm-i386-20100423 #1 SMP Fri Apr 23 10:10:21 BST 2010 i686". It then displays the Debian GNU/Linux system's free software disclaimer and the last login information: "Last login: Fri May 21 11:33:03 2010 from desk4.office.bytemark.co.uk". Finally, the terminal shows the prompt "admin@example:~\$" with a cursor and the letter "b" indicating the next step in the process.

```
david@desk4 $ ssh admin@example.vm.bytemark.co.uk
The authenticity of host 'example.vm.bytemark.co.uk (89.16.174.65)' can't be est
ablished.
RSA key fingerprint is ce:ed:ec:11:5f:17:92:f0:5c:de:cc:ad:43:9a:18:c7.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'example.vm.bytemark.co.uk' (RSA) to the list of know
n hosts.
admin@example.vm.bytemark.co.uk's password: _____ a
Linux example.vm.bytemark.co.uk 2.6.32.11-kvm-i386-20100423 #1 SMP Fri Apr 23 10
:10:21 BST 2010 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri May 21 11:33:03 2010 from desk4.office.bytemark.co.uk
admin@example:~$ b
```

Chapter 13

Configuring SSL Hosting



Before you start this chapter

1. Connect to your server via SSH (see Chapter 12).
-

Each time you need to add a new [SSL](#) site to your Symbiosis system you need to:

1. Acquire an additional dedicated [IP address](#).
2. Configure the site to use that IP address
3. Generate an SSL key and certificate request.
4. Buy or generate an SSL certificate.
5. Upload the new certificate

One additional IP address is needed per SSL certificate. This means that every time you wish to add an SSL certificate to an existing site, it will need to be run under its own IP address.

13.1 Adding an additional IP address



A note on primary IPs

First you must have an additional IP address routed to your machine. Your hosting company should be able to do this. The IP address can either be IPv4 or IPv6. The primary IP of the machine is not usable for SSL, unless you are using the server's self signed certificate.

Once your machine has been allocated an additional IP Address, you must tell your machine to accept traffic addressed to both your original and new IP addresses.

It has been assumed that the site requiring the new IP address is already configured as described in Section 4.1.

1. Use FileZilla to connect to the machine as **admin**.

2. Create the file `/srv/my-brilliant-site.com/config/ip` with the new IP address in it.

Within an hour Symbiosis will have added this new address to your machine's network interface, updated the domain's DNS data and uploaded them to the name servers, as well as reconfigured the Apache web server to use the new IP for that domain.

13.2 Generating an SSL key and certificate request

In order to purchase an SSL certificate, you need to generate an SSL key and a certificate request on the Symbiosis machine.

1. Connect to your machine over SSH as **admin** (see Chapter 12)
2. Change to the `config/` directory of the site that needs the SSL certificate. In our example, we run `cd /srv/my-brilliant-site.com/config`.
3. First we generate the key. To do this run `openssl genrsa -out ssl.key 2048`. This generates a 2048-bit key with no passphrase.
4. Next we generate the certificate request. We run `openssl req -new -key ssl.key -out ssl.csr`. This produces a series of prompts. It is important that the correct information is entered at each prompt. In our case the exchange runs as follows.

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or
a DN.  There are quite a few fields but you can leave some blank For
some fields there will be a default value, If you enter '.', the field
will be left blank.
-----
Country Name (2 letter code) [AU]:GB
State or Province Name (full name) [Some-State]:North Yorkshire
Locality Name (eg, city) []:York
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Bytemark Hosting
Organizational Unit Name (eg, section) []:.
Common Name (eg, YOUR name) []:www.my-brilliant-site.com ❶
Email Address []:bob@my-brilliant-site.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []: ❷
An optional company name []:
```

- ❶ This is the name of the website that the SSL certificate is for. This **must** be correct. We've put **www.my-brilliant-site.com** because that is the name of the site we're going to advertise and use.
- ❷ Do not enter a challenge password.

With that request, you can buy a new certificate. To view the request, run `cat ssl.csr`. It will look like

```
-----BEGIN CERTIFICATE REQUEST-----
AIIB4zCCAUwCAQAwwgaIx CzAJBgNVBAYTAkdCMRMwEQYDVQQIEwpNYW5jaGVzdGVy
MRMwEQYDVQQHEwpNYW5jaGVzdGVyMQ0wCwYDVQQKEwRCbGFoMQ8wDQYDVQQLEwZU
aGluZ3kxhJAcBgNVBAMTFW15LWJyaWwxaWFudC1zaXR1LmNvbTEpMCCGCSqGSIb3
```

```
DQEJARYacm9vdEBteSlicmlsbG1hbnQtY210ZS5jb20wgZ8wDQYJKoZIhvcNAQEB
BQADgY0AMIGJAoGBAMrTiaLKyvsxDz9WHhY5xJvHVKD+dmAuzpv2HichYeJjQTTl
gXdfrrZjVWm45ZJy9TEcB5DM0qsQBSqseMner7YvAJJ3PlTd7o3RkjzttlorPle7
hAkpKLW2dQAvnr3RtK2w8mK+OdJYPSJfzoChCKlG64Un2VmgDfAiNMS4GCi1AgMB
AAGgADANBgkqhkiG9w0BAQUFAAOBgQBx1I52EXnKRL1YfPYIA8CXUeFRZzDbuVKQ
+fWp5Ig5BANBldMnRePY29RH7yJ2YRXTWHfo6erWT4DZVkJhLpWwBTqB/kGcJEjv
zN7D78VSSQzEb2fOcRcx9fWmiIcIWINisjBv9gBbGH7L3UosOtdzEWyzpEjb+Or
nL4UrZV3JA==
-----END CERTIFICATE REQUEST-----
```

The entire output (including the `BEGIN` and `END` lines) should be copied and pasted into the appropriate part of the form when purchasing.

13.3 Purchasing a certificate

There are generally two types of SSL certificate: those that are self-signed, and those that are signed by a third-party. Self-signed certificates are free, but cause warnings to be produced in people's browsers. Third-party certificates are purchased, and hopefully generate no warnings.

For an example of what a warning might look like in your browser, go to <https://example.vm.bytemark.co.uk>.

Purchasing a certificate is straightforward. The first part is the hardest: picking a supplier. There are many available, for example [RapidSSL](#), [Verisign](#), or [Comodo](#).

During the purchase process, you will be asked for the certificate request. Instructions on how to do this are shown in Section [13.2](#).

Once purchased, you should end up with a new certificate, and possibly a "bundle". These should be downloaded onto your local computer. Installation of these is described in Section [13.4](#).

13.4 Uploading your new certificate, and optional bundle

Now we have our certificate, we need to upload it on to our machine. If you've generated the certificate on the machine, you can safely skip this procedure.

1. Connect to your machine using FileZilla.
2. Navigate to the `config/` directory of your domain, using the directory tree in the top right pane. We navigate to `/srv/my-brilliant-site.com/config/`.
3. Find your new certificate and bundle (if applicable) on the local machine, and upload both to the remote machine.
4. Once uploaded, we need to rename the files. This can be done by clicking on the filename in the lower right pane and selecting `Rename` from the menu.
 - The certificate should be renamed to `ssl.crt`.
 - The bundle (if applicable) should be renamed to `ssl.bundle`.

Once this procedure has been completed we can move on to the next section.

13.5 Making SSL mandatory

Once you've configured the SSL certificate, as described in the previous sections, you'll find that your site is accessible to users over HTTP and HTTPS.

If you prefer to ensure that each visitor to your website uses the SSL-protected site you can make it mandatory by creating an empty file called `config/ssl-only`. This should cause the site to be reconfigured to redirect all traffic to the SSL-secured site.

Part II

Reference

Chapter 14

Installing and administering Symbiosis

Symbiosis will install well on a freshly-installed Debian 7.0 system. Currently it is only available for *i386* and *amd64* architectures, running on the Linux kernel.

It is designed to be as friendly as possible for beginners, whilst maintaining flexibility for more experienced systems administrators. Later in this chapter we'll spell out a few basics to bear in mind when working with a system running Symbiosis.

14.1 Installing Symbiosis running on Debian 7.0 (wheezy)

Installing on a fresh Wheezy system is relatively simple. First, add the following to `/etc/apt/sources.list.d/symbiosis.list`:

```
#
# Bytemark Symbiosis Packages
#
deb      http://symbiosis.bytemark.co.uk/wheezy/ ./
deb-src  http://symbiosis.bytemark.co.uk/wheezy/ ./
```

Once that is in your sources, run:

```
apt-get update
apt-get install --install-recommends bytemark-symbiosis
```

At the end of this process, you should have a fully functioning Symbiosis system with all of the features documented here available to you for use.

14.2 Upgrading Symbiosis running on Debian 6.0 (Squeeze)

Debian have [comprehensive release notes](#), of which chapter 4 covers the recommended upgrade procedure. We have provided a shorter version for this, which is immediately below:

The first thing to do is make sure that you have backups. These should be kept in `/var/backups/localhost`, and they should be up to date.

**Note**

Any modifications you may have made to Symbiosis scripts will likely be lost during the upgrade, so you should be prepared to reapply these changes after the upgrade.

Next, alter `/etc/apt/sources.list`. Change all instances of the word `squeeze` to `wheezy`. If you have backports, you can remove them, and any entries for Squeeze LTS should also be removed. Then change the Symbiosis repository lines to match those shown in the previous section.

You can then proceed with the upgrade by running:

```
apt-get update
apt-get dist-upgrade
```

Questions asked during the upgrade

Q: *Restart services during package upgrades without asking?*

A: Yes

Q: *The following services could not be restarted for the PAM library upgrade: pure-ftpd.*

A: OK

Q: *Configuration file /etc/securetty modified: install the package maintainer's version?*

A: Y

Q: *Configuration file /etc/crontab modified: install the package maintainer's version?*

A: Y

Q: *What do you want to do about modified configuration file dovecot.conf??*

A: install the package maintainer's version

Q: *Configuration file /etc/sudoers modified: install the package maintainer's version?*

A: Y

Q: *Configuration file /etc/php5/apache2/php.ini.dpkg-symbiosis: install the package maintainer's version?*

A: Y

Q: *Configuration file /etc/phpmyadmin/config.inc.php modified: install the package maintainer's version?*

A: Y

If you see failures related to dovecot, refer to the release notes (Section [14.3](#)).

You may find that `symbiosis-mysql` `symbiosis-phpmyadmin` have been removed; if so, simply reinstall them:

```
apt-get install symbiosis-mysql symbiosis-phpmyadmin
```

You might get hourly php5 emails stating:

```
PHP Warning: PHP Startup: Unable to load dynamic library '/usr/lib/php5/20100525+ ↵
  lfs/suhosin.so
```

This may be fixed by running:

```
apt-get remove --purge php5-suhosin
```

That should be everything; you may have been asked other questions if you have installed extra packages on your system - answer them as you see fit.

14.3 Release notes

This release of Symbiosis includes a number of new features that are summarised in Chapter 2.

14.3.1 Backup changes

The backup regimen has changed. It is now set to 2 levels of 5 differential backups (from 3 levels of 8 differential backups) resulting in a new set of backups every 30 days, as opposed to every 584 days. Two sets of backups will be kept. This will mean that your machine will definitely need to have enough space for two full backups.

14.3.2 New version of Dovecot

Dovecot has undergone a major upgrade since squeeze. If you've tweaked your dovecot config, it may need attention. If all else fails, try removing `/etc/dovecot/dovecot.conf` and then running:

```
apt-get install -f
```

14.3.3 Command rationalisation

There have been many command changes since the original Lenny release of Symbiosis. Not all of them made much sense, so commands have moved around quite a bit to follow the pattern **symbiosis-package-command**. The old commands have symbolic links in place pointing to the new commands:

```
symbiosis-check-mailbox-password -> symbiosis-email-check-password
symbiosis-check-ftp-password    -> symbiosis-ftpd-check-password
symbiosis-create-sites          -> symbiosis-httpd-configure
symbiosis-create-mass-hosting-sites -> symbiosis-httpd-configure
symbiosis-rotate-logs          -> symbiosis-httpd-rotate-logs
symbiosis-apache-logger         -> symbiosis-httpd-logger
symbiosis-generate-stats        -> symbiosis-httpd-generate-stats
```



Note

These links will be removed in jessie.

There were also commands left over from the Lenny release. These have been removed:

```
symbiosis-create-ssl -> symbiosis-httpd-configure
checkpassword        -> symbiosis-email-check-password
symbiosis_checkpassword -> symbiosis-email-check-password
firewall             -> symbiosis-firewall
firewall-whitelist  -> symbiosis-firewall-whitelist
firewall-blacklist -> symbiosis-firewall-blacklist
```

14.4 Packages installed by Symbiosis

Each component that makes up Symbiosis is separately packaged as follows. Each package can be installed individually if needed.

bytemark-symbiosis Meta-package that pulls in the core requirements for a Symbiosis system, and as well as recommending all packages needed for a complete Symbiosis system.

symbiosis-backup Organises and configures backup2l to backup vital parts of the system, and rsync them to a remote location.

symbiosis-common Contains the core libraries that Symbiosis uses to operate.

symbiosis-cron Provides the per-domain crontab service.

symbiosis-dns Adds automatic DNS generation and upload to the system. Ties in with the Bytemark DNS service.

symbiosis-email Configures Exim and Dovecot for use with Symbiosis.

symbiosis-firewall Maintains the iptables and ip6tables firewalls, as well as providing automatic blacklisting and whitelisting.

symbiosis-ftp Configures pure-ftpd to work with Symbiosis.

symbiosis-httpd Configures the Apache web server.

symbiosis-key Adds the Bytemark Symbiosis key to apt.

symbiosis-monit Provides service monitoring.

symbiosis-mysql Brings in MySQL version 5.1, and configures it to bind to all interfaces, not just localhost, for remote access.

symbiosis-pam Brings in two PAM dependencies to make the system more secure—one checks passwords and warns when they are weak, the other sets per-user temporary directories.

symbiosis-phpmyadmin Brings in phpMyAdmin, and configures it to use HTTP authentication.

symbiosis-updater Adds daily automatic updates from a specified apt `sources.list` file.

symbiosis-webmail Adds webmail functionality, using either Squirrelmail (with Avelsieve) or Roundcube.

symbiosis-xmpp Adds an XMPP/Jabber chat server.

14.5 Systems administration and Symbiosis

Symbiosis is an attempt to encourage best practice at all times in systems administration, whilst keeping things as simple as possible, and free of surprises. As a result there are a few general rules to bear in mind when tinkering with your system.

14.5.1 Use of `root`, and other users

As far as possible Symbiosis will discourage you from using `root` when logging in and configuring the system. This primarily applies to

- Anything in the `/srv/` directory
- The firewall configuration in `/etc/symbiosis/firewall`

For example, if a directory in `/srv` is owned by a system user or group, i.e. one with a UID/GID less than 1000, then it will not show up to various tasks, including, but not limited to,

- Email and FTP logins
- Cron tasks in `config/crontab`
- Apache logging to `public/logs/`
- Mail delivery to mailboxes.

In short, try not to use `root` if at all possible.

However it is perfectly possible to configure separate domains in `/srv/` to be owned by different users, as long as they are non-system users, i.e. ones with user IDs greater than 1000. All programs will respect these permissions.

14.5.2 Customising configurations

Lots of configuration on the system is automatically generated to make Symbiosis work as it does. In previous releases of Symbiosis this meant that files would get overwritten without notice. However as of the Squeeze release in February 2012 configuration files are handled more conservatively.

Two things to watch out for. If a configuration file has

```
# DO NOT EDIT THIS FILE - CHANGES WILL BE OVERWRITTEN
```

written in it, then there is a high chance that any changes will be overwritten. It has to be the exact wording and spacing above for overwriting to take place, so if that sentence is removed from the configuration then it **will not** get overwritten.

Similarly many files are generated from templates, for example DNS and apache snippets. These will now have a checksum at the bottom of the file.

```
# Checksum MD5 586732ff59e60115d0ec1c4905c72773
```

This checksum allows Symbiosis scripts to establish if the template used to generate the snippet has changed, if the data used in the generation has changed, or if the file itself has been edited. For example if an IP address is changed by editing `config/ip`, then that would allow the apache snippet for that domain can be updated, as can the DNS snippet.

This also means that sysadmins can edit the templates, and allow them to regenerate, or edit the snippets themselves safe in the knowledge that their changes will not get overwritten.

14.5.3 Other configuration styles

The Backup2l, Dovecot, and Exim configuration files are generated not with a template, but with a collection of snippets, which are joined and checked using a Makefile. This allows extra configuration snippets to be added in to the configuration.

If it is deemed necessary, sysadmins can add extra snippets to these configurations. The basic procedure is to read the configuration file, and decide where the extra directives need to go. This is made easier by the fact that through the configuration files comments are added showing where each part comes from.

```
# -----
# /etc/exim4/symbiosis.d/10-acl/40-acl-check-mail/00-header
# -----

# ACL that is used after the MAIL command
acl_check_mail:

# -----
# /etc/exim4/symbiosis.d/10-acl/40-acl-check-mail/90-default
# -----

# Allow anything not already denied to connect
  accept
```

In this example, if an extra directive were required in this ACL, then a file could be created in `/etc/exim4/symbiosis.d/10-acl/40-acl-check-mail/`, maybe with the filename `10-do-stuff`. To create the new configuration, we'd then need to run `make` in `/etc/exim4/`. This would regenerate `/etc/exim4/exim4.conf`, and perform a basic syntax check. If happy with the new configuration, then `exim4` could be restarted.

The equivalent Dovecot configuration is in `/etc/dovecot/symbiosis.d/` which generates `/etc/dovecot/dovecot.conf`. The Backup2l configuration is in `/etc/symbiosis/backup.d/conf.d/`, which generates `/etc/symbiosis/backup.d/backup2l.conf`.

Chapter 15

Website Configuration

This is a detailed break down of all the configuration options and files available when configuring website hosting for a domain.

Throughout this chapter, as with the rest of this documentation, the domain `my-brilliant-site.com` is used as an example.

All configuration for the domain `my-brilliant-site.com` will be performed inside the `/srv/my-brilliant-site.com/` directory.

The Bytemark Symbiosis project uses the popular Apache2 software for serving your websites, and this comes complete with PHP5 along with many of the most popular PHP extensions.

15.1 Getting started

This is covered in more detail in Chapter 4.

All the files required for a website for the domain **my-brilliant-site.com** are kept in `/srv/my-brilliant-site.com/public/htdocs/`.

- If this directory does not exist, a **404 Not Found** error will be returned.
- If this directory exists, but is empty, then a default page is shown.
- The index file can be written in *HTML* or *PHP*, and should be called `index.html` or `index.php` respectively.
- Once this directory is present, both <http://my-brilliant-site.com> and <http://www.my-brilliant-site.com> will show the same content, i.e. there is no need to name the site with a **www** prefix.
- If different content is required for <http://www.my-brilliant-site.com> then that should be put in `/srv/www.my-brilliant-site.com/public/htdocs/`.

15.2 CGI scripts

If you wish to use CGI scripts for your domain, then simply copy them to a directory named `cgi-bin/` beneath the `public/` directory. They must all be marked as executable. This means setting the permissions to **755**. In FileZilla, right click the file and select File Permissions... from the menu. The file should have **Execute** set for the owner, group, and public permissions.

For example, for **my-brilliant-site.com** the scripts would live in `/srv/my-brilliant-site.com/public/cgi-bin/`.

Any **executable** files in that directory will now be treated as CGI scripts for your domain. For example if you created the file `/srv/my-brilliant-site.com/public/cgi-bin/test.cgi` This would be referred to as: <http://my-brilliant-site.com/cgi-bin/test.cgi>

15.3 Statistics

Each hosted website will have visitor statistics automatically generated and accessible at <http://my-brilliant-site.com/stats/>. These statistics will be updated once per day, and the raw access logs will be made available as `/srv/my-brilliant-site.com/public/logs/`.

These daily statistics can be disabled by creating the file `config/no-stats`.

For example, for **my-brilliant-site.com**, creating the file `/srv/my-brilliant-site.com/config/no-stats` will ensure that statistics are no longer generated for that domain. If you wish to remove any existing statistics, remove the directory `/srv/my-brilliant-site.com/public/htdocs/stats/`.

It is also possible to customise the statistics generated by editing the file `config/webalizer.conf`. This file is documented at [the Webalizer project website](#).

If there are many sites on the same machine, then it is possible to customise all the sites' Webalizer configurations by editing the template that is available at `/etc/symbiosis/apache.d/webalizer.conf.erb`. Configuration files will be updated when the statistics are next generated, but only for sites whose configurations either do not exist, or have not been edited by hand.

15.4 Testing new websites

You can view new websites before any DNS changes are made.

For example, if the virtual machine **example.vm.bytemark.co.uk** is hosting **www.my-brilliant-site.com**, i.e. the directory `/srv/my-brilliant-site.com/public/htdocs/` has been created, then the website can immediately be viewed at <http://my-brilliant-site.com.testing.example.vm.bytemark.co.uk>.

There are some important things to note though: - There is no **www** part added to the domain name—it is just the directory name prepended to **.testing.example.vm.bytemark.co.uk**. - This testing alias isn't guaranteed to work in all cases, for complex site setups it might not work entirely as expected. - The testing alias only allows the testing of websites. Therefore FTP logins, email delivery, or checking is explicitly unsupported.

15.5 Displaying the same content under two domains

In this scenario, you have registered two domains for example **my-domain.com** and **my-domain.co.uk**, but you want the same content to be served at both addresses. There is no need to create two separate directory structures, you can just set up one directory structure and then create a soft link (aka symbolic link or symlink) to the second.

1. Once the `my-domain.com` directory structure has been completed, log on to your machine as admin over SSH.
2. Run the command `ln -s /srv/my-domain.com /srv/my-domain.co.uk`
3. A soft link of the entire `my-domain.com` directory is created, the top level directory being named `my-domain.co.uk`.

Browsing to `my-domain.co.uk` will show the same content that appears at `my-domain.com`.

15.6 Redirecting to the preferred website domain

If a document tree were created in `/srv/my-brilliant-site.com/public/` then that site would be available under two hostnames:

- <http://my-brilliant-site.com/>
- <http://www.my-brilliant-site.com/>

There are people who prefer to use only a single name, and to automatically redirect visitors using the *wrong* name to using the preferred name. This can easily be achieved by using Apache's `mod_rewrite` facility.

If you prefer all visitors see the `www`-based site you could create the file `/srv/my-brilliant-site.com/public/htdocs/.htaccess` with the following contents:

```
RewriteEngine on
RewriteCond %{HTTP_HOST} !^www.*$ [NC]
RewriteRule ^(.*)$ http://www.%{HTTP_HOST}/$1 [R=301,L]
```

This examines each incoming request, and if the hostname doesn't begin with "www." then it is prepended to the request and a redirect is issued.

15.7 Custom Apache configuration

It is perfectly possible to alter the way Symbiosis configures Apache, either for an individual domain, or for all domains hosted on the server.

Symbiosis hosts sites on a server in one of two ways, based on the IP address that site has configured. If it uses one of the server's primary IP addresses, then it is assumed that the site is hosted using the "mass-hosting" configuration. If the site has a secondary IP assigned then Symbiosis generates an individual snippet for that site, and Apache is configured to use that snippet when dealing with HTTP requests for that domain. Both configuration techniques are configured using a template, which allows the server's administrator to fiddle with, and tweak the configuration.

In `/etc/symbiosis/apache.d/` there are a number of templates that are used to generate configuration snippets for both the mass-hosting, as well as individual sites.

15.8 SSL Configuration

[Secure Sockets Layer](#) is a technique used to encrypt communication between two machines on a network. It uses a system of public and private keys to encrypt and decrypt the connection—the public key is used by the sender to encrypt, and the private key is used by the receiver to decrypt. This protocol is used not only for transactions involving a web server and browser, but also by the email servers and their clients.

In addition to the public key encryption, there is a system of trust that validates that the certificate presented actually belongs to the server that is presenting it. This system involved having the certificate signed by a trusted authority. Web browsers and email clients tend to come with a selection of certificates from trusted authorities pre-installed, which allows them to verify a previously unseen certificate as valid.

Having a certificate signed by a trusted authority involves having varying degrees of identity checks made, and paying a fee. Vendors that are able to sell you a certificate include [Rapid SSL](#) and [Comodo](#).

As standard, a Symbiosis machine will come with an SSL certificate installed. However it will be a "self-signed" certificate, i.e. one that has not been signed by a trusted authority. This means that whenever a program connects

to your machine using SSL a warning will be shown saying something along the lines of *unable to verify certificate because the issuer is unknown*. This **does not** affect the security of the connection,

Verifying a self-signed certificate as trusted can be done using the certificate's *fingerprint*, on the machine using **openssl**. The default certificate on a Symbiosis machine is kept in `/etc/ssl/ssl.crt`. First, the fingerprint of the certificate needs to be determined. To do this run the following, as root:

```
openssl x509 -noout -in /etc/ssl/ssl.crt -fingerprint
```

That should output something similar to

```
SHA1 Fingerprint=B8:C7:1B:3F:EC:94:F2:9F:77:BC:09:60:CD:E3:EF:E0:04:F4:23:6A
```

Now that we have the fingerprint, we can compare it against that presented in a browser or email client. The fingerprint of a certificate should be shown in the application's certificate viewer, allowing a comparison to be made between the fingerprint on the machine, and the one being presented in the application.

The nature of [SSL](#) is such that only one certificate can be used per service per [IP address](#). This typically means that a new [IP address](#) is needed for a website that needs a new SSL certificate.

15.8.1 Generating a self-signed certificate

If you do not wish to purchase a new certificate, you can use generate your own certificate as follows. This assumes you've completed the instructions for generating a key and certificate request in [Section 13.2](#).

1. Log on to your machine as **admin** over SSH.
2. Change to the `config/` directory of your domain. In our example, we run **cd /srv/my-brilliant-site.com/config**.
3. Now run **openssl x509 -days 365 -req -in ssl.csr -signkey ssl.key -out ssl.crt**. This will produce output similar to the following. Note that the information entered in the certificate request is shown.

```
Signature ok
subject=/C=GB/ST=North Yorkshire/L=York/O=Bytemark Hosting/OU=/CN=www.my- ↵
brilliant-site.com/emailAddress=bob@my-brilliant-site.com
Getting Private key
```

This has now generated the certificate, and saved it in **ssl.crt**. This certificate is valid for a year from the date generated.

15.9 Logging

Requests for sites are logged to one of two places. If a request is received for a site that exists on the machine, then that request is logged to `public/logs/access.log`. If that request generates an error, then it is logged to `public/logs/error.log`. These logs are updated as requests are received.

If a request is received for a domain that is not present on the box, then it is logged to `zz-mass-hosting.access.log` if it received on the primary IP of the machine. If the request comes on any other IP then it is logged to `other_vhosts_access.log`. Both of these last two files are located in `/var/log/apache2`.

15.10 Configuration layout

Here is an example configuration layout for the domain `my-brilliant-site.com`, all of which is contained under `/srv/my-brilliant-site.com/`.

config/no-stats If this file exists, no statistics will be generated for this domain. Existing statistics in `/public/htdocs/stats/` will not be removed automatically.

config/ssl-only If this file exists, traffic will be redirected to the SSL version of the website.

config/webalizer.conf This is the Webalizer configuration file for this domain.

public/cgi-bin/ This is the directory which may be used to hold [CGI scripts](#) for your domain.

public/htdocs/ This is the directory from which content is served for the URLs <http://my-brilliant-site.com/> and <http://www.my-brilliant-site.com/>. If this directory does not exist visitors will be shown an error page.

public/htdocs/stats/ This directory will be automatically created, if it isn't already present, and updated with statistics referring to the number of visitors to your website.

public/logs/access.log This file contains the Apache webserver access log for the domain. It will be archived daily, and removed after 30 days.

public/logs/ssl_access.log This file contains the Apache webserver access log for the domain when accessed over SSL.

public/logs/error.log This file contains the Apache webserver error log for the domain, if the domain has been configured to run under its own IP address. It will be archived daily, and removed after 30 days. If the site does not have its own IP address, then errors are logged to `/var/log/apache2/zz-mass-hosting.error.log`.

public/logs/ssl_error.log This file contains the Apache webserver error log for the domain when accessed over SSL, if the domain has been configured with its own IP address.

Chapter 16

Email Configuration

This is a detailed break-down of all the configuration options and files available when configuring how email is handled for a domain. Throughout this chapter, the domain `my-brilliant-site.com` is used as an example. Thus all the configuration for `my-brilliant-site.com` will be inside the `/srv/my-brilliant-site.com/` directory.

16.1 Port Configuration

The mail servers have been set up with standard port assignments as follows. These are all the standard ports for the protocols.

Service	Port	Encryption
POP3	110	TLS
IMAP	143	TLS
SMTP	25 or 587	TLS
POP3	995	SSL
IMAP	993	SSL
SMTP	465	SSL
Sieve	4190	TLS

16.2 Accepting email for a domain

In order for a domain to be configured to accept email, one of two things must be present. Either the domain must have a `mailboxes/` directory present, or one of the files `config/default_forward` or `config/aliases` must be present.

For example, if the domain **my-brilliant-site.com** would like to host mail normally, i.e. one mailbox per user hosted on the same machine, then the directory `/srv/my-brilliant-site.com/mailboxes/` should be created. Then in there, one directory per user should be created. If **bob@my-brilliant-site.com** would like to receive mail, then `/srv/my-brilliant-site.com/mailboxes/bob/` should be created.

Assuming that this is the only directory inside `/srv/my-brilliant-site.com/mailboxes/` then only mail addressed to **bob@my-brilliant-site.com** will be accepted. Any other mail addressed to **my-brilliant-site.com** will be rejected.

If you would like to accept all mail for **my-brilliant-site.com**, regardless of who it is addressed to, then create the file `/srv/my-brilliant-site.com/config/default_forward`. The contents of this file should be a single

email address, or a comma-separated list of email addresses. For example, to forward all mail to **bob@my-brilliant-site.com**, regardless of who it is addressed to, then `/srv/my-brilliant-site.com/config/default_forward` should contain `bob@my-brilliant-site.com`.

If you would like the domain **nomail.my-brilliant-site.com** not to receive any mail at all, then remove the directory `/srv/nomail.my-brilliant-site.com/mailboxes/` and ensure that the file `/srv/nomail.my-brilliant-site.com/config/default_forward` does not exist.

16.3 Email for Unix users.



Before you start this section

1. Both a unix user and a normal Symbiosis email user can be set up to receive email to the same address. The normal user will always take precedence over the unix user and have their mail delivered to their inbox first, so take care when using this feature!

A new feature in this release is the ability to have unix users with email accounts based in their home directories. These will receive emails for the host name of the machine, which you can find out by running `hostname` on the command line. The result of this will display the latter half of the email address the system users will get, eg, the part after the `@`. The other half will be dictated by their username, eg, "admin" or "my-user".

To start with, we create a `.password` file in, eg, `/home/my-user/`. This can contain either a plaintext password (not recommended) or a password encrypted with the [symbiosis-encrypt-password](#) utility.

Once the password file is in place, the new user will start to be able to start receiving email. Logins over SMTP, IMAP, and POP3 will all work identically to a normal email user, with the same ports and SSL/TLS requirements.

Unix users' Maildir directories will reside in `/home/my-user/Maildir` by default. This allows these users to use system mail readers such as `mutt` in order to read and send email, obviating the need to use IMAP and SMTP.

These users are also able to control the following files:

- `.forward` - to control forwarding of emails.
- `.vacation` - to set a vacation (holiday) message.
- `.sieve` - to set up Sieve filters.

16.4 Password files

The password for a mailbox should be set by the contents of a file named `password` inside a user's mailbox directory. The contents of this file may be in plain text, or encrypted.

To encrypt a password on the command line, you can run the following command, substituting "my password" for your password. This encrypts the password using the SHA-512 algorithm.

```
echo "my password" | symbiosis-encrypt-password > password
```

This just uses the standard `crypt` function available under most Linux platforms, as well as perl and PHP.

16.5 Suffixes

All email addresses can be used with a suffix. This allows people to filter their email by the To: address. The separator between the local part and suffix is the + sign.

For example, Bob signs up to a shopping site at <http://example.com>. He might use `bob+example@my-brilliant-site.com` his email address when signing up, such that he can filter all email from that shop.

16.6 Enforcing mailbox size with quotas

Symbiosis can enforce users' mailbox size with quotas. This will prevent mail from being delivered to a user if their mailbox grows too large.

A default quota for each individual mailboxes in a domain can be specified in `config/mailbox-quota`. A per-mailbox quota can be defined in a file named `quota` which resides in a user's mailbox directory.

These files both have the same format, which is just a number of bytes over which mail should not be delivered. This number can have a suffix of k, M, or, G which represent kilobytes, megabytes, and gigabytes, or ki, Mi, or Gi to represent kibibytes, mebibytes, and gibibytes, respectively.

For example, to limit the size of each mailbox for the domain **my-brilliant-site.com** to 200MB, i.e. 200,000,000 bytes, put `200M` in `/srv/my-brilliant-site.com/config/mailbox-quota`.

To grant **bob@my-brilliant-site.com** a 1GiB quota, i.e. 1,073,741,824 bytes, put `1Gi` in `/srv/my-brilliant-site.com/mailboxes/bob/quota`.

Quotas in a user's mailbox directory take precedence over the default quota.

16.7 Server-side filtering using Sieve

Sieve is a [standard language](#) that users can employ to filter their email on the server. Additionally using any one of a number of [clients](#), users can edit their filtering rules without needing shell access to the server.

Each user can create a number of scripts in a directory called `sieve.d/`, with the current script being kept in a file called `sieve`.

Only one of these scripts can be active at a given time for each user; add to an existing file rather than creating a new one if you require extra filters.

16.8 Forward files

There are two methods of forwarding email. The first is a per-mailbox forwarding service, and the second is a per-domain service. For the per-user service, a file named `forward` should be put in a user's mailbox directory. The per-domain service uses the same file format as the per-user service, but the file should be uploaded to `config/default_forward` instead.

For example, **bob@my-brilliant-site.com** would set up a file called `/srv/my-brilliant-site.com/mailboxes/bob/forward`.

If all the mail for **my-brilliant-site.com** needed to be forwarded elsewhere, then the file would be called `/srv/my-brilliant-site.com/config/default_forward`.

Both of these files can be interpreted in two ways. Firstly they can be a comma separated list of email addresses. For example, if Bob wanted to forward his email onto Charlie and Dave, his forward file might read

```
charlie@example.com, dave@example.com
```

The second way these files are interpreted is as an Exim filter file. The full specification is documented at [the Exim project site](#).

Here are some examples of what is possible.

To forward mail on, but keep a copy

```
# Exim filter
unseen deliver charlie@example.org
unseen deliver dave@example.com
```

To rewrite all mail for a domain to **example.com**. This is probably best used in `config/default_forward`.

```
# Exim filter
deliver $local_part@example.com
```

The [Exim documentation](#) has further examples of what is possible.

16.9 Vacation messages

It is possible to set a vacation message for a user by putting a message in file called `vacation` in the user's mailbox directory.

For example, for **bob@my-brilliant-site.com**, the message would go in `/srv/my-brilliant-site.com/mailboxes/bob/vacation`. On Bob's return, the people who received vacation messages are logged to `/srv/my-brilliant-site.com/mailboxes/bob/vacation.log`. Once he's read it, that file, along with `/srv/my-brilliant-site.com/mailboxes/bob/vacation` and `/srv/my-brilliant-site.com/mailboxes/bob/vacation.db` should **all** be removed.



Important

Vacation messages can cause irritate other email users by replying to mailing lists, email bounces, and so on. Every effort is made to stop this from happening, but it is by no means fool-proof.

16.10 Email alias lists

Each domain can have a list of aliases. This is just a file that contains a list of local parts, and a list of places they should be sent on to. This file should be in the `config/` directory and is named `aliases`.

For example, **my-brilliant-site.com** has a list of dummy addresses that should be sent on to Bob. So the aliases file would be kept at `/srv/my-brilliant-site.com/config/aliases` and contains the following.

```
webmaster    bob@my-brilliant-site.com
chairman     charlie@example.com
staff       bob@my-brilliant-site.com, charlie@example.com, dave@example.com
```

This ensures that `webmaster@my-brilliant-site.com` is sent to `bob@my-brilliant-site.com`; `chairman@my-brilliant-site.com` is sent to `charlie@example.com`; `staff@my-brilliant-site.com` is sent to `bob@my-brilliant-site.com`, `charlie@example.com`, and `dave@example.com`.

16.11 Customising SpamAssassin

The configuration for SpamAssassin for the **admin** user is kept in `/srv/.spamassassin/user_prefs`. Here you can adjust what score is needed to reject spam, and which tests are used during scanning. This file will only appear after a mail has been received with spam detection turned on, but one can be created and configured before this occurs.

The file contains comments and instructions, and further tips can be found on the [SpamAssassin wiki](#).

In brief, to cause **more mail** to be rejected, you need to reduce the threshold score. Therefore change the line reading `#required_score 5` should be changed to `required_score 4`. Notice that the `#` has been removed at the start of the line to un-comment it.

Similarly if mail is being rejected, you can increase the score.

Further instructions can be found on the [SpamAssassin wiki](#).

There is no facility to train the SpamAssassin Bayesian learner yet.

16.12 Filtering mail using headers

Headers are added to messages when spam or virus scanning is enabled. These can be used by email clients to filter email, for example in to spam or quarantine folders.

With spam scanning enabled, any email that **is** accepted has the following headers added

- `X-Spam-Score`
- `X-Spam-Bar`
- `X-Spam-Status`

The score is determined by SpamAssassin, and is the basis for acceptance or rejection. The higher the score, the more certain SpamAssassin is that the message is unwanted. The default threshold for rejection is 5.

The bar is a length of pluses or minuses that provide an easy-to-parse representation of the score. A positive score is given pluses, a negative score minuses. For example a score of 5.6 would be represented as `+++++`; a score of -2.2 would be represented as `--`.

The status is always either `innocent` or `spam`, depending on the score.

When virus scanning is enabled, the header `X-Anti-Virus` is added to messages that have been scanned. This is set to either `infected` or `clean`.

16.13 Using real-time blacklists from Spamhaus

There are three lists from Spamhaus that can be used to reject email based on the sender's IP address, namely

The Spamhaus Block List (SBL) a list of addresses from which Spamhaus does not recommend receiving email.

The Exploits Block List (XBL) a list of hijacked computers infected by third party exploits and viruses.

The Policy Block List (PBL) a list of addresses that should not be sending unauthenticated email at all.

These lists are combined to form the Zen list.

The following instructions will enable use of these lists on our example domain **my-brilliant-site.com**.

1. Connect to your machine using FileZilla
2. On the remote directory tree, navigate to `/srv/my-brilliant-site.com/config/`.
3. In this directory, create another directory called `blacklists/`. This is done by clicking the right mouse button on the `config/` directory, and selecting Create directory from the menu that pops up.
4. On your local machine create a file called `zen.spamhaus.org`. This is just an empty file.
5. Once this is done, navigate to the `blacklists` directory on the remote file system, and select `zen.spamhaus.org` from the local file system, and upload it. Make sure that the remote file has the correct name, i.e. no extra `.txt` extension.

That is all that is needed to start using the Spamhaus Zen blacklist. If you'd rather use a combination of lists create one or more of the following files:

- `sbl.spamhaus.org` to enable the SBL list
- `xbl.spamhaus.org` to enable the XBL list
- `pbl.spamhaus.org` to enable the PBL list
- `sbl-xbl.spamhaus.org` to enable the combined SBL and XBL list
- `zen.spamhaus.org` to enable the combined SBL, XBL, and PBL list

16.14 Manually blocking incoming mail from specific sources

While publicly maintained blacklists like spamhaus are much easier to rely on and lower maintenance, at some point you might find occasion to block specific email senders. Symbiosis allows blocking based on these criteria:

- Hostname of sender, which is matched against the reverse DNS of the sender's IP. Example entry: `*.bad-domain.com`
- IP of sender, which can be a single IP or a range specified in CIDR notation (be wary of blocking too much if you use this option). Example entry: `192.168.0.1`
- Address of sender. This option may specify wildcard records, eg `"*@example.com"` will catch all emails from that domain. Please note this works on the "envelope from" rather than the "from" address. Example entry: `bad_sender@example.com`.

To block with one of these criteria, you can use:

- `/etc/exim4/blacklist/by_hostname` for eg `.bad-domain.com`
- `/etc/exim4/blacklist/by_ip` for `192.168.0.1`
- `/etc/exim4/blacklist/by_sender` for `bad_sender@example.com`

Each entry to these files should be on a new line.

It is also possible to explicitly allow email from senders that would otherwise be blacklisted by adding entries in similarly named files under `/etc/exim4/whitelist`.

16.15 Installing Roundcube webmail

Roundcube is another webmail client which can be used with Symbiosis. To install it, simply run:

- `sudo apt-get install symbiosis-webmail-roundcube`

This will install the roundcube packages and set them up. MySQL credentials will be required to complete the installation, and you should provide these to `dpkg` when prompted.

After the install has completed, Roundcube will be the default webmail client, accessible at `my-brilliant-site.com/webmail` and all other domains hosted on the machine.

16.16 Configuration layout

Here is an example configuration layout for the domain `my-brilliant-site.com`. All the following files are kept in `/srv/my-brilliant-site.com/`.

mailboxes/ This is where individual mailboxes are defined. If this directory does not exist, then mail will not be accepted for `my-brilliant-site.com`, unless a default forwarding address or filter has been set up.

mailboxes/bob/ Mail will be accepted for the email address `bob@my-brilliant-site.com`.

mailboxes/bob/Maildir/ This is where the email for `bob@my-brilliant-site.com` will be delivered. It will be **created automatically** upon receipt of the first message to that address.

mailboxes/bob/password File containing the password for `bob@my-brilliant-site.com` allowing him to collect his email over IMAP/POP3, and relay email using SMTP. His username is the same as his email address. See Section 16.4 for more information.

mailboxes/bob/quota File containing the quota for a user. The quota should a number of bytes. This can be followed by one of `k`, `M`, or `G` to specify kibibytes, mebibytes, or gibibytes respectively. For example `100M` would be 100 mebibytes, or 104857600 bytes. See Section 16.6 for more information.

mailboxes/bob/forward File containing either a comma-separated list of addresses, or an Exim filter. All mail addressed to `bob@my-brilliant-site.com` will be forwarded to the list of addresses, or processed by the filter. See Section 16.8 for more information.

mailboxes/bob/vacation File containing a vacation message for Bob. See Section 16.9 for more information.

mailboxes/bob/sieve File containing a Sieve filter. This can be edited by the user without shell access to the server. See Section 16.7 for more information.

config/aliases This file contains a list of aliases for a domain. The format is the local username followed by one or more spaces, and then comma separated list of email addresses which should receive the mail. See Section 16.10 for more information.

config/default_forward File containing either a comma-separated list of addresses, or an Exim filter. All mail addressed to the domain `my-brilliant-site.com` for local parts without directories under `mailboxes` will be forwarded to this address or processed by this filter. See Section 16.8 for more information.

config/bytemark-antispam This file indicates use of Bytemark's wholesale anti-spam service, which is no longer provided.

config/antispam If this file is present, then all email for the domain my-brilliant-site.com will be scanned by [SpamAssassin](#) to determine whether it is spam. If it is spam, it will be rejected. If that file begins with the word tag, mail will never be rejected, just tagged as usual. See the section called “[Scanning email to prevent spam and viruses](#)” more information.

config/mailbox-quota If this file is present, then all mailboxes for this domain will have their quota determined by this file. The quota should a number of bytes. This can be followed by one of k, M, or G to specify kilobytes, megabytes, or gigabytes respectively. For example 100M would be 100 megabytes, or 100,000,000 bytes. See Section 16.6 for more information.

config/antivirus If this file is present, then all email for the domain my-brilliant-site.com will be scanned for viruses by [ClamAV](#). If a message is determined to contain a virus, it will be rejected. If that file begins with the word tag, mail will never be rejected, just tagged. See the section called “[Scanning email to prevent spam and viruses](#)” for more information.

config/blacklists/sbl.spamhaus.org Reject mail for this domain if the sending machine's IP is listed in the [Spamhaus Block List](#).

config/blacklists/xbl.spamhaus.org Reject mail for this domain if the sending machine's IP is listed in the [Spamhaus Exploits Block List](#).

config/blacklists/pbl.spamhaus.org Reject mail for this domain if the sending machine's IP is listed in the [Spamhaus Policy Block List](#).

config/blacklists/sbl-xbl.spamhaus.org Reject mail for this domain if the sending machine's IP is listed in either the Spamhaus or the Exploits block lists.

config/blacklists/zen.spamhaus.org Reject mail for this domain if the sending machine's IP is listed in the [Spamhaus Zen Block List](#), which is a combination of the Spamhaus, Exploits, and Policy block lists.

Chapter 17

XMPP Reference

XMPP is a protocol that supports both private instant messages, and group instant messages. The server also supports features such as roster management, for keeping track of contacts and showing who is and is not online. Here is a broad overview of what the `symbiosis-xmpp` package supports:

- Federation - this is where users of your XMPP server may communicate with users of any other correctly configured XMPP server, with their own, locally hosted account. This will also allow a user to connect to multiple user chats (see below) on the local server or remote servers, should they wish to do so.
- Roster (contact list) management - before receiving messages from a new contact, each user must add the other to their contact list. The server will then remember contacts, such that they will be known on all a user's XMPP clients.
- Private messages - a user is able to speak to any of their online contacts
- Multiple user chat (MUC) - this feature enables users to communicate in groups, rather than one on one. These chats will often be named after the intended subject of discussion, eg "office" or "managers". If you wish, you can host a MUC that anyone else can connect to and use to chat. Think of this as being a cross between an instant message and a mailing list.

The server that underlies the `symbiosis-xmpp` package is called Prosody. It's website is <http://prosody.im/> .

Chapter 18

Setting up domain based multi-user FTP/FTPS Access



Before you start this chapter

1. Connect to your server over [SFTP](#) using FileZilla (see [Chapter 3](#)).
2. Set up a website (see [Chapter 4](#)).

This section describes how to set up distinct per-domain FTP/FTPS users. Where with the old setup you may have logged in with a user like:

```
my-brilliant-site.com
```

With this method you can have many users, of the form:

```
bob@my-brilliant-site.com
alice@my-brilliant-site.com
charlie@my-brilliant-site.com
```

Users can be limited to specific directories, and quotas imposed on each, so **bob@my-brilliant-site.com** may be limited to eg, **htdocs**, which works out as **/srv/my-brilliant-site.com/public/htdocs**. These domain-based users do not have SSH/SFTP access.

Each user is defined using four fields, added to **/srv/my-brilliant-site.com/config/ftp-users**:

```
username
password
directory (limited to directories in /srv/my-brilliant-site.com/public)
quota
```

Configuration for giving **bob** access and a quota of 10 megabytes will look like so:

```
bob:your-password-here:htdocs:10M
```

This can be done for any number of users - all are specified in the same ftp-users file.



Warning

Please be aware that despite being limited to the `public/` directory when logging in over FTP/FTPS, it is trivial for the user to read files elsewhere on the filesystem, for example by using specially written PHP scripts. It is advisable to grant access only to trusted people.



Note

Please be aware that the FTP quota will include all log data from the web-server in `public/logs/`, as well as the automatically generated statistics in `public/htdocs/stats/`.
In addition

Chapter 19

Firewall Reference

The firewall component of the Symbiosis system serves to protect the system by controlling its inbound and outbound connections. It comprises of a set of rules, and automatic whitelist and blacklist generation.

The firewall should be configured over SFTP as the **admin** user, and any changes made will take affect immediately.

19.1 Allowing and denying access to services

All usual firewall configuration can be carried out by creating and deleting files in `/etc/symbiosis/firewall/`. In this directory there are a number of subdirectories. Permissions for inbound connections are stored in `/etc/symbiosis/firewall/incoming.d/`, and outbound connections in `/etc/symbiosis/firewall/outgoing.d/`.

These files are all of the format `number-name`. The number determines the position of the rule in the firewall, the name is the name of the service that we wish to permit. These names are stored in `/etc/services`. There are also names that do not correspond to services, which are documented in the next section.

Additionally if the name is not known then the file format can be `number-number` where the first number specifies the position of the rule in the firewall, and the second number is the port that should be opened. For example, the files `10-http` and `10-80` achieve the same effect.

Finally, each file can contain a list of hostnames or IP addresses to which that rule will apply, one per line. For example, if addresses were added to an incoming rule, named `incoming.d/10-accept`, all connections **from** those addresses would be **accepted**. If a file were added named `outgoing.d/20-reject` and address added to that file, then outgoing connections **to** those addresses would be **rejected**.

For example, to allow an incoming connection to arrive at your machine, and be accepted, on port 22, you would create the file `/etc/symbiosis/firewall/incoming.d/10-ssh`. The firewall will update as soon as the file has been created, so no commands are needed to be run.

If you were wishing to ensure that your host would only accept incoming SSH requests from your office you might create the same file with the contents `office.my-brilliant-site.com`.

This would ensure that when the firewall was generated incoming connections on the SSH port would be accepted from the host `office.my-brilliant-site.com` but not from anywhere else.



Note

If hostnames, rather than IP addresses are used, then they are translated to IP addresses at the time the firewall is generated using DNS. If the IP address of a hostname changes, then the firewall may not function as intended until any cached DNS entries have expired, and the firewall has been regenerated.

19.2 Predefined special rules

There are a number of rules that don't naturally fit the convention described above. This list describes rules that have been written specially for Symbiosis to cope with these situations. Each rule described below can be used in both `incoming.d/` and `outgoing.d/`, and for both IPv4 and IPv6 addresses, unless otherwise specified.

These rules are used in the same way as those described in the previous chapter. Files are added in the `incoming.d/` or `outgoing.d/` directory with the name prefixed by a number giving the position of the rule. The files can contain addresses or hostnames, one per line, against which the rule should be applied.

accept Accept all connections. Uses the iptables `ACCEPT` target.

allow Alias of **accept**.

blacklist Alias of **reject**.

collector Permit TCP connections on port 1919.

dns Accept incoming TCP and UDP connections from port 53 to high-numbered, unprivileged ports. Designed to allow replies to DNS queries. This rule can be removed in favour of **related**. This is for **incoming** connections only.

drop Drop all connections. Uses the iptables `DROP` target.

essential-icmpv6 Accept ICMPv6 packets that are essential for IPv6 networking to operate. Without this rule the machine IPv6 networking **will not work**. It permits ICMPv6 types destination-unreachable, packet-too-big, parameter-problem, router-solicitation, router-advertisement, neighbor-solicitation, and neighbor-advertisement. This is **IPv6** only.

established Permit connections that are already established. Uses the iptables `ESTABLISHED` target.

ftp Permit TCP connections on both ports 20 and 21, i.e. ftp and ftp-data.

icmp Permit all ICMP connections. This **IPv4** only.

icmpv6 Permit all ICMP6 connections. This is **IPv6** only.

imager Permit TCP connections on port 5000.

new Permit new connections.

ping Permit ICMP types echo-request, echo-reply, and ttl-exceeded, for allowing the machine to be pinged, and show up on traceroutes.

reject Reject all connections. Uses the iptables `REJECT` target. For TCP connections a TCP reset is sent. Otherwise it returns port unreachable.

reject-www-data Rejects all outgoing connections that are started by the **www-data** user. **N.B.** if addresses are put in this file, they will be **not** rejected, which is the reverse of how every other rule functions. See Section 19.4 for more information.

related Accept new connections, but only if they are associated with an existing one, for example DNS queries, or FTP data transfer.

whitelist Alias of **accept**.

These rules are all contained in `/usr/share/symbiosis/firewall/rule.d/`. It is perfectly possible to write your own rules based on those in this directory, but they should be kept in `/usr/local/share/symbiosis/firewall/rule.d/`.

19.3 An example firewall

This example should be read in conjunction with the previous sections. A machine has the following firewall rules defined for its incoming connections.

- `incoming.d/00-related`
- `incoming.d/00-established`
- `incoming.d/05-essential-icmpv6`
- `incoming.d/05-ping`
- `incoming.d/07-ssh` which contains `1.2.3.4`, and `2001:41c8:1:dead:beef::/64` on separate lines.
- `incoming.d/10-http`
- `incoming.d/20-25`
- `incoming.d/99-reject`
- `incoming.d/100-666`

This would set up a firewall that would do the following tests, in order:

1. Accepted all packets from **established** connections.
2. Accepted all packets from **related** connections
3. Accepted all ICMPv6 packets required for IPv6 connectivity.
4. Accepted ICMP/ICMPv6 packets required for pings and traceroutes.
5. Accepted new TCP/UDP connections to port 22 (SSH), but only from `1.2.3.4` or addresses in the `2001:41c8:1:dead:beef::` netblock.
6. Accepted new TCP/UDP connections to port 666. Note that this rule comes before `10-http`, even though it is called `100-666`. This is because the order is given by the ASCII rather than numerical value of the filename.
7. Accepted new TCP/UDP connections to port 80 (HTTP).
8. Accepted new TCP/UDP connections to port 25 (SMTP).
9. Rejected anything that had not been accepted yet.

These rules would be installed for IPv4 and IPv6 connections using `iptables` and `ip6tables` respectively. To inspect the firewall rules at any given time, you can run `sudo iptables -L -v -n` which will return the current firewall status. In this example, the rules would look like this.

```
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target    prot opt in  out source            destination
    0     0 ACCEPT    all  --  lo  *  0.0.0.0/0         0.0.0.0/0
   13  1012 whitelist all  --  *  *  0.0.0.0/0         0.0.0.0/0
    0     0 blacklist all  --  *  *  0.0.0.0/0         0.0.0.0/0
    0     0 ACCEPT    all  --  *  *  0.0.0.0/0         0.0.0.0/0 state ESTABLISHED
    0     0 ACCEPT    all  --  *  *  0.0.0.0/0         0.0.0.0/0 state RELATED
    0     0 ACCEPT    icmp --  *  *  0.0.0.0/0         0.0.0.0/0 icmp type 8
```

```

0      0 ACCEPT      icmp -- * * 0.0.0.0/0      0.0.0.0/0 icmp type 0
0      0 ACCEPT      icmp -- * * 0.0.0.0/0      0.0.0.0/0 icmp type 11
0      0 ACCEPT      tcp  -- * * 1.2.3.4         0.0.0.0/0 tcp dpt:22
0      0 ACCEPT      udp  -- * * 1.2.3.4         0.0.0.0/0 udp dpt:22
0      0 ACCEPT      tcp  -- * * 0.0.0.0/0      0.0.0.0/0 tcp dpt:80
0      0 ACCEPT      udp  -- * * 0.0.0.0/0      0.0.0.0/0 udp dpt:80
0      0 ACCEPT      tcp  -- * * 0.0.0.0/0      0.0.0.0/0 tcp dpt:666
0      0 ACCEPT      udp  -- * * 0.0.0.0/0      0.0.0.0/0 udp dpt:666
0      0 ACCEPT      tcp  -- * * 0.0.0.0/0      0.0.0.0/0 tcp dpt:25
0      0 ACCEPT      udp  -- * * 0.0.0.0/0      0.0.0.0/0 udp dpt:25
0      0 REJECT      all  -- * * 0.0.0.0/0      0.0.0.0/0 reject-with icmp- ←
port-unreachable

```

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)

```
pkts bytes target      prot opt in out source      destination
```

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)

```
pkts bytes target      prot opt in out source      destination
0      0 ACCEPT      all  -- * lo 0.0.0.0/0    0.0.0.0/0
7    1388 ACCEPT      all  -- * * 0.0.0.0/0    0.0.0.0/0 state ESTABLISHED
0      0 ACCEPT      all  -- * * 0.0.0.0/0    0.0.0.0/0 state RELATED
0      0 REJECT      all  -- * * 0.0.0.0/0    0.0.0.0/0 owner UID match 33 ←
reject-with icmp-port-unreachable

```

Chain blacklist (1 references)

```
pkts bytes target      prot opt in out source      destination
0      0 REJECT      all  -- * * 71.63.72.4    0.0.0.0/0 reject-with icmp- ←
port-unreachable
0      0 REJECT      all  -- * * 61.145.118.190 0.0.0.0/0 reject-with icmp- ←
port-unreachable

```

Chain whitelist (1 references)

```
pkts bytes target      prot opt in out source      destination
13    1012 ACCEPT      all  -- * * 212.110.163.132 0.0.0.0/0

```

This listing shows how the rules in the files under `/etc/symbiosis/firewall/` are translated into iptables rules. It also shows that by default all connections on the loopback interface `lo` are permitted, and that the whitelist and blacklist tables have references in the INPUT, i.e. incoming, table before the rules defined in `/etc/symbiosis/firewall/incoming.d/` are applied.

IPv6 rules follow the same format, and can be checked by running `sudo ip6tables -L -v -n`.

19.4 Allowing web applications to make remote connections

By default the firewall contains the rule `outgoing.d/50-reject-www-data`, which is designed to reject outgoing connections made by the web server. This prevents many ways of infecting a machine with malicious software following a compromise in a web application.

Following establishing that a web application has security flaws, hackers will attempt to trick the application into downloading their software onto the machine. Once downloaded the software is used in various ways, for example to participate in denial of service attacks, or to access confidential data on the machine. Thus this rule is a basic defence against vulnerable web applications being exploited, and is a good thing to have in place.

However there are legitimate cases when a web application might need to make such a connection. For example, if you have an application which needs to make outgoing HTTP connections to update RSS or Twitter feeds, you will

need to either add permitted addresses to the file, or remove it completely.

For example, to permit access to `search.twitter.com`, add that address to the file `outgoing.d/50-reject-www` to permit it.

Adding addresses to this file will **permit** outgoing connections to those addresses.

19.5 Making custom additions to your firewall

The Symbiosis firewall package should allow you to carry out the most common tasks, simply by creating files named after the services you wish to permit or deny.

However there are times when you might wish to make your own custom additions, and for this purpose the firewall package allows you to run an unlimited number of custom scripts/programs once it has loaded the rules - these scripts may perform arbitrary actions, but will be most typically used to update the firewall rules, via the `iptables` or `ip6tables` commands.

The program `run-parts` is used to execute scripts in `/etc/symbiosis/firewall/local.d/`, after the firewall has finished loading. This means that the scripts have to have to fulfil the naming conditions described in the [run-parts\(8\)](#) manual page. Essentially the script should be marked executable, and only contain alphanumeric characters in its name.



Warning

If any scripts in `local.d/` exit with a non-zero status the firewall will be deemed to have failed in some way, and the firewall will be restored to its prior state.

19.6 Blocking abusive remote hosts

The `symbiosis-firewall-blacklist` tool runs four times an hour, and is designed to scan your server's logfiles for abusive behaviour from malicious remote hosts. Malicious activity which is detected will result in the remote host being denied further access to your server.

Currently we regard malicious activity as:

- Invalid SSH logins.
- Invalid FTP logins.
- Invalid SMTP/POP3/IMAP/ManageSieve logins.

Every 15 minutes various logfiles are scanned for certain patterns to search for new malicious IPs, and the firewall is updated.

These patterns are defined in `/etc/symbiosis/firewall/patterns.d/`. For example, for SSH the following pattern definition is used:

```
#
# The logfile we look for matches within.
#
file = /var/log/auth.log ❶
```

```
#
# Any matches will be denied access to these ports.
#
# Comma-separated values are expected.
#
ports = 22 ❷

#
# Patterns we'll match upon.
#
Failed password for invalid user [^ ]+ from __IP__ port [^ ]+ ssh2 ❸
Failed password for [^ ]+ from __IP__ port [^ ]+ ssh2
```

- ❶ Is the file to search
- ❷ Are the ports to block
- ❸ Are the regular expressions to look for, where `__IP__` is a pre-defined regular expression that matches both IPv4 and IPv6 addresses.

If an IP matches one of those patterns in the period since the last check was made, it is added to the blacklist.

Disabling the firewall completely will disable the blacklisting behaviour, but you might also wish to disable that separately.

To do this, login over SFTP as **admin** and create the file `/etc/symbiosis/firewall/blacklist/disabled`. This will immediately disable and clear the blacklist.



Note

IPv6 addresses are masked to a /64, which is the smallest assignment of addresses recommended for an end site.

19.7 Whitelisting "known-good" IP addresses

The `symbiosis-firewall-whitelist` tool runs once per hour, and is designed to perform the opposite task to the `symbiosis-firewall-blacklist` script - in short it is designed to ensure that any remote host which has successfully connected to your server in the past isn't (accidentally) blacklisted in the future.

Every hour the script will examine the successful logins which have been observed recently. Each IP address which has successfully been the source of a login attempt will be permitted access to the system on a global basis, and will thus not be locked out.

As with the automatic blacklist, IPv6 addresses are masked to a /64, which is the smallest recommended assignment for an end site.

To disable the automatic whitelist, login over SFTP as **admin** and create the file `/etc/symbiosis/firewall/whitelist.d/disabled`. This will immediately clear the whitelist, and prevent further updates.

You can add your own entries to the whitelist, which never expire, by creating entries in the directory `/etc/symbiosis/firewall/whitelist.d/`. Create the file `/etc/symbiosis/firewall/whitelist.d/<ip address>` and the specified IP address will not be blacklisted, or refused access to your server.

19.8 Disabling the firewall

If you wish you may disable the firewall completely, allowing remote users to connect to any service you have running upon your machine.

We'd not recommend that you disable the firewall, because it does provide a increase in system security, but if you wish it is possible by executing the following two commands:

```
touch /etc/symbiosis/firewall/disabled
sudo symbiosis-firewall flush
```

The presence of the disabled rule will not itself clear the firewall, merely prevent further updates to it, which is why the `flush` command is needed.

19.9 Configuration layout

All configuration of the firewall is conducted via the presence or absence of files in a number of directories beneath `/etc/symbiosis/firewall/`. Actions and rules are all kept under `/usr/share/symbiosis/firewall/`.

`/etc/symbiosis/firewall/blacklist.d/` A persistent record of IP addresses which are blacklisted, such that no connections will be permitted from them.

`/etc/symbiosis/firewall/blacklist.d/disabled` If this file is present, then the automatic blacklisting is disabled.

`/etc/symbiosis/firewall/disabled` If this file is present then the firewall will be disabled. However this will not clear the firewall rules. See Section 19.8.

`/etc/symbiosis/firewall/incoming.d/` Settings related to the incoming connections your machine will receive.

`/etc/symbiosis/firewall/local.d/` The place to add local customisations.

`/etc/symbiosis/firewall/outgoing.d/` Settings related to the outgoing connections your machine is permitted to initiate.

`/etc/symbiosis/firewall/patterns.d/` A collection of pattern files use by `symbiosis-firewall-blacklist` to automatically determine addresses to blacklist

`/etc/symbiosis/firewall/whitelist.d/` A persistent record of IP addresses which are always allowed to connect to your server.

`/etc/symbiosis/firewall/whitelist.d/disabled` If this file is present, then the automatic whitelisting is disabled.

`/usr/share/symbiosis/firewall/action.d/` This directory contains the various actions that the `symbiosis-firewall` uses to maintain the firewall. If you wish to write your own actions, or change the ones that come with `symbiosis`, they should go in `/usr/local/share/symbiosis/firewall/action.d/`.

`/usr/share/symbiosis/firewall/rule.d/` This directory contains the various pre-defined rules described in Section 19.2. If you wish to add your own rules, or change the ones provided, they should go in `/usr/local/share/symbiosis/firewall/rule.d/`.

Chapter 20

DNS Hosting

To take full advantage of the Symbiosis system, your domain needs to be configured to have Bytemark's name servers as authority for it.

What follows only applies if our name servers are used; if that is not the case you will need to manage your DNS data outside of the Symbiosis system. Section 20.1 gives a listing of the records needed for the correct functioning of the system.

All domains which are hosted upon a Symbiosis system will have their DNS records automatically uploaded to the Bytemark Content DNS servers.

By default a set of typical records is created for each hosted domain with MX records pointing to the local system, and aliases such as *www.* and *ftp.* for convenience. If you wish you may edit the records to make custom additions or otherwise make changes to those defaults.

For the domain "my-brilliant-site.com" you will find the auto-generated DNS records in `/srv/my-brilliant-site.com/config/dns/my-brilliant-site.com.txt`

The DNS files are uploaded to the Bytemark content DNS service every hour, and allow you to use the full range of available TinyDNS options. These options are documented upon the [Bytemark Website](#) and in the [TinyDNS documentation](#).

20.1 Example DNS records

This is an example of the records Symbiosis generates for `my-brilliant-site.com`. They are created automatically and stored in `config/dns/my-brilliant-site.com.txt`.

DNS records example

```
#
# Nameserver records. ❶
#
.my-brilliant-site.com::a.ns.bytemark.co.uk:300
.my-brilliant-site.com::b.ns.bytemark.co.uk:300
.my-brilliant-site.com::c.ns.bytemark.co.uk:300

#
# The domain name itself ❷
#
=my-brilliant-site.com:89.16.174.65:300
```

```

#
# Useful aliases. ❸
#
+ftp.my-brilliant-site.com:89.16.174.65:300
+www.my-brilliant-site.com:89.16.174.65:300
+mail.my-brilliant-site.com:89.16.174.65:300

#
# A record for MX ❹
#
+mx.my-brilliant-site.com:89.16.174.65:300

#
# The domain name itself -- AAAA record and reverse. ❺
#
6my-brilliant-site.com:200141c80001596d00000000000000065:300

#
# Useful aliases -- AAAA records only
#
3ftp.my-brilliant-site.com:200141c80001596d00000000000000065:300
3www.my-brilliant-site.com:200141c80001596d00000000000000065:300
3mail.my-brilliant-site.com:200141c80001596d00000000000000065:300

#
# AAAA record for MX
#
3mx.my-brilliant-site.com:200141c80001596d00000000000000065:300

#
# MX record -- no IP defined, as this is done separately above. ❻
#
@my-brilliant-site.com::mx.my-brilliant-site.com:15:300

```

- ❶ These lines create *NS* and *SOA* records for `my-brilliant-site.com` pointing at `a.ns.bytemark.co.uk`, `b.ns.bytemark.co.uk`, and `c.ns.bytemark.co.uk`. The time-to-live for these records is 300 seconds. Note that the double colons in these records are deliberate as the IP addresses are defined elsewhere by Bytemark.
- ❷ This creates an *A* record pointing `my-brilliant-site.com` to the IP address `89.16.174.65`, and a *PTR* record for the reverse. Again, the TTL is 300 seconds.
- ❸ These three lines add *A* records for expected aliases. Once again, the TTL for these records is 300 seconds.
- ❹ This line adds in an *A* record for the *MX* record defined below.
- ❺ From here the IPv6 equivalents of **2**, **3**, and **4** are specified, using *AAAA* records is used instead of an *A* record. Note that IPv6 addresses are specified in full, without any colons.
- ❻ This last record creates an *MX* record directing mail for `my-brilliant-site.com` to `mx.my-brilliant-site.com`, with a distance of 15. The double colon is deliberate since we defined the *A* record for `+mx.my-brilliant-site.com` in <4>, and an *AAAA* record for the same name in <5>.

20.2 Adding a wild-card hostname record

In addition to these records for each domain, a wild-card *A* record is needed for the hostname such that the `.testing.` prefix works. If your machine is at Bytemark, this has already been setup for your machine's Bytemark alias, for example `example.vm.bytemark.co.uk`.

If your machine is not hosted at Bytemark, or your hostname does not end in `bytemark.co.uk` then you will need to set this alias up. Adding the following line to your DNS file will work, assuming the domain is hosted at Bytemark. This assumes that your machine is called `host.example.com` and that your machine's IP address is `1.2.3.4`.

```
+*.host.example.com:1.2.3.4
```

20.3 Moving domains between machines using the Bytemark content DNS service

If you wish to move your domains between two machines running Symbiosis and using the Bytemark content DNS service, you must contact Bytemark Support to arrange the domain to be moved between content DNS accounts.

This results from the necessity for ensuring that only people with the proper authorisation can change live DNS data. Once a domain has been hosted on our network, a content DNS account will have sole authority for it.

If you purchase a second server and move some of your domains onto it, or purchase a domain from another Bytemark customer you must contact us to move authority for the domain into the correct account.

Until this is done, although the Symbiosis system will be creating and uploading data it will not be to the account with the authority to make the data live.

20.4 Configuring SPF and DKIM records

SPF and *DKIM* are standards that help mail servers decide if email is legitimate, ensuring it is more likely to reach the intended recipient's inbox instead of being rejected or marked as spam. Both these technologies require creation of one or more DNS records.

20.4.1 Adding SPF records

Before adding any records, a policy needs to be decided. The guide at [OpenSPF](#) can help determine what the record should look like. The default policy Symbiosis uses is `v=spf1 +a +mx ?all`.

To create SPF records simply create the file `/srv/my-brilliant-site.com/config/spf`. Nothing more is required if the default policy is adequate. If you have decided on a different policy, then you can just write it to this file.

A task is run hourly to generate the DNS data and upload it to the Bytemark DNS servers, at which point the domain will start benefiting from it. If you wish to speed up this process, run `sudo symbiosis-dns-generate --verbose`.

20.4.2 Adding DKIM records

DKIM is a way of cryptographically signing email headers to provide a level of confidence surrounding the origin of said email. Configuring DKIM requires a private RSA key, and a DNS record specifying the public part of the key, along with a policy dictating how the key should be used. For DKIM to work in Symbiosis two files are required, one contains the private key, and the second contains the selector (or nothing).

1. To generate the private key, run **openssl genrsa -out /srv/my-brilliant-site.com/config/dkim.key 2048 -outform PEM** on your server. This will generate a key that is 2048 bits long.
2. Next create the file `/srv/my-brilliant-site.com/config/dkim`, either as an empty file or with the selector in it. If the file is empty, the selector is left as "default".

Once both files are in place the hourly DNS task will update the DNS records for your domain and upload them as usual. If you wish to speed up this process, run **sudo symbiosis-dns-generate --verbose**.

Chapter 21

Scheduled tasks

Jobs can be scheduled to run on a per-domain basis. This is configured in the same style as the traditional crontab, and is kept in the `config/` directory of a domain. Configuration of the per-domain crontab is covered in Chapter 9.

21.1 Testing the crontab

The crontab can also be tested. To do this you have to SSH to the machine, usually as **admin** to run the command.

For example, to test the **my-brilliant-site.com** crontab navigate to `/srv/my-brilliant-site.com/config/` and run **symbiosis-crontab --test crontab**.

The **my-brilliant-site.com** crontab reads

```
# Send any output to Bob
#
MAILTO=bob@my-brilliant-site.com

#
# run at 18:40 every day
#
40 18 * * *          echo Hello Dave.

#
# run at 9am every Monday - Friday
#
0 9 * * mon-fri wget http://www.my-brilliant-site.com/cron.php

#
# Run once a month
#
@monthly             /usr/local/bin/monthly-job.sh
```

Therefore the output generated is

```
Environment
-----
HOME = /srv
LOGNAME = admin
PATH = /usr/local/bin:/usr/bin:/bin
MAILTO = bob@my-brilliant-site.com
```

```

=====
Jobs next due -- Local time 2010-06-17T17:57:37+01:00
-----
Date                Command
-----
2010-06-17T18:40:00+01:00  echo Hello Dave.
2010-06-18T09:00:00+01:00  wget http://www.my-brilliant-site.com/cron.php
2010-07-01T00:00:00+01:00  /usr/local/bin/monthly-job.sh
=====

```

**Note**

The only environment variables that can be set within your crontab are PATH and MAILTO. All the rest are set automatically, and cannot be altered.

21.2 System scheduled tasks

There are various automated tasks which are executed upon a Symbiosis system. These scheduled tasks are responsible for automating things such as:

- The addition of new IP addresses to your system.
- The generation and upload of DNS data.

The following section documents precisely which jobs are installed by default, along with their purpose.

These are the system tasks which are installed by default:

/etc/cron.d/symbiosis-common This carries the rudimentary password checks on mail and FTP passwords on an hourly and weekly basis.

/etc/cron.d/symbiosis-cron This is responsible for launching any user-scheduled jobs, as described in Chapter 21, and is run every minute.

/etc/cron.d/symbiosis-firewall The jobs here are responsible for checking for new blacklist and whitelist entries, as discussed in Section 19.6. The whitelist and blacklists are regenerated every 15 minutes. The whole firewall is reloaded hourly.

/etc/cron.d/symbiosis-monit This schedules the Symbiosis service monitor, which is described in Chapter 24. This is run every two minutes.

/etc/cron.d/symbiosis-dns This regenerates DNS data for all the domains in `/srv/`, and triggers an upload to the Bytemark DNS server.

/etc/cron.d/symbiosis-updater This script is responsible for fetching and applying pending package updates. It is run on a daily basis.

/etc/cron.daily/symbiosis-httpd-rotate-logs This manages rotation of the webserver log files for each domain.

/etc/cron.hourly/symbiosis-configure-ips This adds IP addresses configured by domains in `config/ip` to the host, as described in Chapter 13

/etc/cron.hourly/symbiosis-httpd-configure This task creates a per-IP Apache configuration file for new IP addresses, and is closely related to the previous task.

Chapter 22

Database configuration

Initially the **root** password for the database is the same as that of the **admin** user used to connect to your machine via SSH or SFTP. To change this you can use the phpMyAdmin interface.

As a general rule, each application should have its own username and access rights, to make sure that there is a degree of separation between all the applications on a server. This can all be done through the [phpMyAdmin interface](#).

22.1 Enabling remote MySQL access

As a security measure, your MySQL server is not opened to the world. However you might wish to access it remotely for performing queries, or allowing other hosts to otherwise communicate with it.

Your MySQL server should be configured already to listen upon your external IP addresses. Therefore only two steps are needed to configure remote access: opening the firewall, and adding a user with remote privileges to the database.

22.1.1 Opening the firewall for MySQL

To open a hole in the firewall to the whole internet, you should create the file `/etc/symbiosis/firewall/incoming.d/55-mysql`. It is a good idea to restrict access to the database to a list of known IP addresses. To do this simply add IP addresses to the above file, one per line.

Chapter 19 gives full details of how the firewall works.

22.1.2 Adding a user with remote privileges

There are two ways to do this, either using the MySQL command line tool, or via phpMyAdmin. This section will cover doing it with the latter.

1. In phpMyAdmin, select the Privileges link from the front page, once you've logged in to it as **root**— see Chapter 8 for details on how to do this.
2. The privileges section will present a **User Overview**, at the bottom of which there is a link to Add a new user.
3. In the **Add a new user** screen, fill out the details in the form as needed, making sure that the Host field is set to **Any host**.

The privileges tick boxes lower down should be selected carefully. Most applications will need at least those in the Data section, and some of those in the Structure section. Check the documentation of the software you're using to see what it requires.

If you want an account with all privileges, select check all.

4. Once you're satisfied with everything, click Go. This will confirm that a user has been created.
5. Now return to the home screen by clicking the phpMyAdmin logo at the top left of the screen.
6. Finally, on the front page click the Reload privileges link to make sure MySQL knows about this new user.

You should now be able to access the MySQL database remotely, using this new username and password.

Chapter 23

Backup Reference

The Symbiosis system includes a component designed to handle backups, using the flexible [backup2l software](#).

backup2l was selected due to its simplicity and flexibility, which allows it to be used easily. By default the backup software executes once per day and archives the contents of significant directories to a local directory.

23.1 Configuration

In Symbiosis the Backup2l configuration is generated from the snippets in `/etc/symbiosis/backup.d/conf.d/`.

- The local directories to backup (`/etc/`, `/srv/`, etc).
- The destination to which the backups should be stored (`/var/backups/localhost/`)
- The number of backups to keep.

23.2 Advanced Configuration

Additionally we've configured the backup software to easily execute a number of scripts before and after the backup is performed:

`/etc/symbiosis/backup.d/pre-backup.d/` Any executable script located in this directory is executed, prior to a backup execution.

`/etc/symbiosis/backup.d/post-backup.d/` Any executable script located in this directory is executed after a backup has completed.

23.3 Listing Backup Contents

To list the contents of your backup area you need to run backup2l with the "-l" flag:

```

all.1: /etc/.pwd.lock
all.1: /etc/GeoIP.conf.default
all.1: /etc/X11/Xresources/x11-common
all.1: /etc/X11/Xsession
all.1: /etc/X11/Xsession.d/20x11-common_process-args
all.1: /etc/X11/Xsession.d/30x11-common_xresources
all.1: /etc/X11/Xsession.d/40x11-common_xsessionrc
all.1: /etc/X11/Xsession.d/50x11-common_determine-startup
...

```

Here you will see the contents of the `/etc/` directory which have been archived.

If you'd like to restrict this view you can apply a regular expression to filter the results. For example we can list the files which match the pattern `passwd` with this command:

```

~$ sudo backup2l -l passwd
Listing locations...
all.1: /etc/exim4/passwd.client
all.1: /etc/passwd
all.1: /etc/passwd-
all.1: /etc/phpmyadmin/htpasswd.setup
all.1: /etc/pure-ftpd/pureftpd.passwd
...

```

23.4 Restoring From Backup

To illustrate how this works, an example is used. We're looking for a backup of the file `/etc/passwd`.

1. First log in to your machine over SSH (see Chapter 12) as **admin**.
2. To find the available versions of the file, run **sudo backup2l -l '/etc/passwd\$'**. The dollar sign is there to show that you want an exact match of `/etc/passwd`. The first time you run **sudo** you will be prompted for the **admin** password. The following output will be generated by backup2l.

```

backup2l v1.5 by Gundolf Kiefer

Active files in <all.1101>: 1
  found in all.1101:      0  (    1 left)
  found in all.11:       1  (    0 left)

Listing locations...
all.11: /etc/passwd

```

This shows the latest available version of the file

3. To recover it you should run **sudo backup2l -r '/etc/passwd\$'**. The following output will be generated

```

backup2l v1.5 by Gundolf Kiefer

Active files in <all.1101>: 1
  found in all.1101:      0  (    1 left)
  found in all.11:       1  (    0 left)

Restoring files...
  all.11.tar.gz: 1 file(s) using 'DRIVER_TAR_GZ'

```


That has restored the file to `etc/passwd` in the current directory. It is **not recommended** to run this program in the `/` directory, as any existing files will get overwritten.

23.5 Recovery From Earlier Backups

It is also possible to pick which version of a file you wish to restore.

1. First login to your machine over SSH as **admin**
2. Then, to show all available versions of a file, run **sudo backup2l -a '/etc/passwd\$'**. Again, the first time you run **sudo** you will be prompted for a password. The following output is generated.

```
backup2l v1.5 by Gundolf Kiefer

Listing available files...
all.101  -      1067 06/18/08 13:59:47 0000.0000 0644 /etc/passwd
all.101  +      1118 06/19/08 11:29:10 0000.0000 0644 /etc/passwd
all.108  -      1118 06/19/08 11:29:10 0000.0000 0644 /etc/passwd
all.108  +      1153 08/27/08 10:25:45 0000.0000 0644 /etc/passwd
all.11   -      1067 06/18/08 13:59:47 0000.0000 0644 /etc/passwd
all.11   +      1153 08/27/08 10:25:45 0000.0000 0644 /etc/passwd
all.1    +      1067 06/18/08 13:59:47 0000.0000 0644 /etc/passwd
```

Note that the versions are not shown in date order, and that the dates are in the US `mm/dd/yy` format. In that list the `+` indicates that the file is new and thus contained in the archive file. A `-` indicates that the file has been removed (or replaced). Choose which backup you wish to recover from.

3. To recover the file dated 19th June 2008, you need backup number 101—remember the `+` indicates that it is present in that archive. To recover that file, run **sudo backup2l -t 101 -r '/etc/passwd\$'**

```
backup2l v1.5 by Gundolf Kiefer

Active files in <all.101>: 1
  found in all.101:      1  (    0 left)

Restoring files...
  all.101.tar.gz: 1 file(s) using 'DRIVER_TAR_GZ'
```

Notice the `-t 101` argument which specifies which backup we want to restore from.

We have now successfully restored the file to `etc/passwd` in the current directory. We can check by running **ls -la etc/**

```
total 16
drwxr-xr-x  2 root  root  4096 2008-09-09 09:56 .
drwxr-xr-x 14 root  root  4096 2008-09-09 09:51 ..
-rw-r--r--  1 root  root  1118 2008-06-19 11:29 passwd
```

23.6 Offsite backup storage

The Symbiosis system assumes that it has access to an associated external storage area. It will try and use `rsync` to upload the backups to this area, via a script named `/etc/symbiosis/backup.d/post-backup.d/99-upload-backup`.

If the host is on Bytemark's network, this script can establish the backup space name automatically. Otherwise you can specify it manually by setting the full `rsync` path in `/etc/symbiosis/dns.d/backup.name`.

23.7 Recovering from the offsite backup storage

Before each backup a second script will synchronise the remote backup space locally, ensuring that a complete set of backups are held in both places. This means that if disaster strikes your machine, it is straightforward to recover your backups. This is done by running `/etc/symbiosis/backup.d/pre-backup.d/00-download-backup`.

This also helps to maintain the integrity of the differential backups provided by `backup2l` by replacing any files accidentally removed from the local backup directory before the backup starts.

23.8 Trimming the size of the local backups.

It is possible to reduce the size of the backups stored locally. The first thing to do is check the current status of the backups by running `sudo backup2l -s`. This will present a summary of the current backups. For example:

```
backup2l v1.5 by Gundolf Kiefer
```

```
Summary
```

```
=====
```

Backup	Date	Time	Size	Skipped	Files+D	New	Obs.	Err.
all.1	2010-08-10	02:52	41.7M	0	3836	3836	0	0
all.11	2010-11-01	04:45	38.1M	0	3935	1517	1418	0
all.12	2011-01-21	04:27	39.7M	0	3985	561	511	0
all.121	2011-01-30	04:38	10.5M	0	4001	137	121	0
all.122	2011-02-08	03:54	1.5M	0	4029	129	101	0
all.123	2011-09-07	05:08	33.8M	0	3892	1437	1574	0
all.124	2011-09-16	05:07	1.3M	0	4791	956	57	0
all.125	2011-09-25	04:45	868K	0	5676	928	43	0
all.126	2011-10-04	05:15	11.3M	0	6559	990	107	0
all.127	2011-10-13	04:29	894K	0	7444	928	43	0
all.128	2011-10-22	04:59	345K	0	8329	935	50	0
all.13	2011-10-31	05:03	45.7M	0	9218	6833	1600	0

```
Filesystem      Size  Used Avail Use% Mounted on
/dev/vda         10G  1.9G  7.6G  20% /
```

From here it is possible to see which levels of backups that can be pruned. In the above example the third-level backups `all.121` to `all.128` can be pruned, as there has been a subsequent second level backup, `all.13`. The downside of this is that any changes contained in those backups will be lost, and only changes from the `all.12` will be available.

To prune these backups run `sudo backup2l -p 121`. This will then show `Backup2l` removing `all.121` and all its dependent backups.

```
backup2l v1.5 by Gundolf Kiefer
```

```
Purging <121>...
```

```
removing <all.121>
removing <all.122>
removing <all.123>
removing <all.124>
removing <all.125>
removing <all.126>
removing <all.127>
```

```
removing <all.128>
```

Finally we need to make sure these deletions are synchronised to the remote backup space, to ensure that our deleted files do not mysteriously return again prior to the next backup run.

```
sudo /etc/symbiosis/backup.d/post-backup.d/99-upload-backup
```

Which will provide output similar to that shown below.

```
Sending backups to example.backup.bytemark.co.uk::example/example.vm.bytemark.co. ←→
  uk...
building file list ... done
deleting localhost/all.lock
deleting localhost/all.128.tar.gz
....
deleting localhost/all.121.error.gz
deleting localhost/all.121.check
localhost/

sent 2.95K bytes  received 22 bytes  1.98K bytes/sec
total size is 400.59M  speedup is 134742.36
```

Those level three backups will no longer exist.

Chapter 24

Service Monitoring

The Symbiosis system is comprised of several distinct components, which we've documented throughout the course of this reference:

- The MySQL database server.
- Exim & Dovecot servers for handling email.
- Apache for serving websites.
- The FTP server, proftpd
- The inotify cron daemon, incron.

Each of these services runs in an independent fashion, and it is possible under certain circumstances that these services might fail, or stop themselves.

To handle the case of services failing to execute normally we've included an automated service checker as part of the Symbiosis system. The service checker will check upon the health of your system, by default once every two minutes, and it will automatically restart any services which have failed.

The 'symbiosis-monit' command is responsible for testing each of the available services, and restarting the failed ones. By default it is executed every two minutes, such that it may respond quickly to failures. It will also stop services that are not required. For example if the machine is not configured to scan any domains' mail, then SpamAssassin will be stopped.

At any time you wish to check upon the health of your system you may launch it manually, when connected to your server via SSH.

```
admin@example:~$ sudo symbiosis-monit
= service test report =====

* Host: kvm4.vm.bytemark.co.uk
* Tests started at: Fri, 11 Jun 2010 15:55:45 +0100

* apache2: PASS
* clamav-daemon: PASS
* clamav-freshclam: PASS
* cron: PASS
* dovecot: PASS
* exim4: PASS
* mysqld: PASS
```

```
* pure-authd: PASS
* pure-ftpd: PASS
* spamassassin: PASS
* sshd: PASS

* 11/11 tests passed on first attempt.
* Tests finished at: Fri, 11 Jun 2010 15:55:46 +0100

= End of service test report =====
```

In this case all services are working correctly, so "PASS" was reported instead of the failing "FAIL". The possible output status are:

FAIL The service failed.

PASS The service appears to be running correctly.

Part III

Support Guide

Chapter 25

Troubleshooting Symbiosis

We're happy to accept bug reports via our usual support system. But you can make it easier for us to assist you if you check the common things first.

We have produced an FAQ which might answer the questions you're asking.

If none of the suggestions on this help it would aid us if you were very specific about the problem you're experiencing.

25.1 Database problems?

If you already had a password configured for the MySQL database prior to installing our packages it will be unchanged. The password for the root user is only changed if it is unset when the packages are installed or updated for the first time.

The Debian MySQL packages create a local user for automated use, so if you're unsure of your MySQL password you may use this login to reset it. You may find details of the Debian login contained in the file `/etc/mysql/debian.cnf`.

25.2 Firewall problems?

You, or a customer with FTP or SFTP/SSH access, may become locked out of the machine if repeated attempts are made to access the machine incorrectly

If you believe you've become locked out, via the firewall, it is possible to fix this if you have another means of connecting to your server.

**Note**

Users who have their hosting with Bytemark will be able to use the [Console Shell](#) to gain access to their machine, even if the network is disabled, or the firewall is refusing direct connections.

1. Using your fall-back connection method connect to your server.
2. Navigate to the `/etc/symbiosis/firewall/blacklist.d` directory with the command **`cd /etc/symbiosis/firewall/blacklist.d`**
3. Check the contents of the directory with the command **`ls /etc/symbiosis/firewall/blacklist.d`**; the presence of the file `<ip address>.auto` confirms the problem.

4. remove the file, `rm /etc/symbiosis/firewall/blacklist.d/<ip address>.auto` and restart the firewall with the command `firewall`

**Tip**

You can whitelist an IP address to ensure it is never blocked by the Symbiosis firewall. Create the directory `/etc/symbiosis/firewall/whitelist.d/` and the file `/etc/symbiosis/firewall/whitelist.d/<ip address>`. Note that you do not add ".auto" to that filename.

25.3 Package problems?

Every evening your system will be configured to update itself. This ensures that you'll have any Debian-provided security updates applied to your system. It will also update your system to the latest available collection of the Bytemark Symbiosis packages.

If your system fails to update you may correct this by running, as root:

```
apt-get update
apt-get dist-upgrade
apt-get -f install
```

25.4 Permission problems?

The mail-server and FTP-server we're running will refuse to work with directories which are owned by the root user.

If you find that you've added a new site/mailbox to your system and it doesn't work but existing ones do then this is most likely the source of the problem.

Unless you're handling ownership in a special way you may reset the permissions to avoid this problem by running the following command:

```
chown -R admin.admin /srv/
```

25.5 SSL problems?

If you run into problems with the configuration of SSL-based sites please get in touch, this support is still very new and there might be a couple of kinks to work out of the process.

The most common problem is that you need to install a "bundle" or keychain file.

If the missing keychain/bundle is the problem you'll see this logged in the Apache SSL error file beneath `/var/log/apache2`. Each generated SSL-site will use its own logfile - rather than the global `access.log` and `error.log` file. So taking a look at that could be useful with the shell command:

```
tail /var/log/apache2/*ssl*.log
```

You should be able to validate the combined SSL private key and certificate via the use of the `openssl` tool; run the shell command:


```
openssl verify /srv/my-brilliant-site.com/config/ssl.key
..
error 18 at 0 depth lookup:self signed certificate
OK
```

You're looking for the "OK" at the end, rather than the error message which is harmless.

Chapter 26

FAQ

Q: *My new website shows only the "Bytemark Unconfigured Host" page.*

A: Simply upload a file to the root of your website directory, i.e. `/srv/my-brilliant-site.com/public/htdocs/`, with the name of `index.html`, or `index.php`.

Your new index will override our default one.

Q: *I want <http://my-brilliant-site.com/> and <http://www.my-brilliant-site.com/> to show different content.*

A: When we've created websites so far we've created directories without the **www** prefix, for example `/srv/my-brilliant-site.com`. These directories are served when clients request both <http://www.my-brilliant-site.com> and <http://my-brilliant-site.com>. If you'd like different content simply create a new directory with the **www** prefix .

Q: *What is the password for the admin user?*

A: When the packages are installed a new local admin user is created. The password for that account will be same as for your existing **root** account. For all work with Symbiosis we recommend you connect and login as user **admin**.

Q: *How do I redirect one domain to another?*

A: We'll use a redirection of **my-brilliant-site.com** to **example.com**, so that when users go to **www.my-brilliant-site.com**, they are redirected to **www.example.com**.

1. Firstly, create the `my-brilliant-site.com` `htdocs` directory as usual, i.e. `/srv/my-brilliant-site.com/public/htdocs/`.
2. Now inside that directory, create a file `.htaccess` with the following line inside:

```
Redirect 301 / http://www.example.com/
```

and the redirection should start working.

See the [Apache documentation](#) for a full description of the `Redirect` directive.

Q: *Help! My browser issues a big warning when I try to connect to my Webmail.*

A: This is because the SSL certificate is self-signed. We've provided the following step-by-step guides to accepting this certificate permanently in some of the more popular browsers.

- Apple Safari
- Mozilla Firefox (version 2)

- Mozilla Firefox (version 3)
- MS Internet Explorer (version 6)
- MS Internet Explorer (version 7)

Q: *I've set up the ftp-password as instructed, but I can't log in*

A: The first thing to check is permissions. These need only be checked if you've created domains with the root user.

By default the Bytemark Symbiosis package installs a new user **admin** which owns the `/srv` directory, and can create domains inside that directory. If you've set up the `/srv` directory by hand, it is probably owned by **root**, and any domains inside that directory will also be owned by **root**. This will prevent both the email services and FTP services from running correctly.

To fix this, you need to create a user to own all the `/srv/` directories. It is suggested that you create a user called `admin`, as this will fit into the Bytemark scheme.

```
adduser --home /srv --no-create-home admin
```

This command will prompt you for all sorts of information, including a password, and it will create a group called **admin** too. Once you've created this user, you will need to change the ownership of the `/srv` directory.

```
chown -R admin.admin /srv
```

Q: *How do I modify the firewall, where is it located?*

A: Please see Chapter [19](#).

Q: *How do I enable remote access to MySQL?*

A: Please see Section [22.1](#).

Q: *I have a PHP script that sends emails or tries to make an external connection via http and it is not responding.*

A: This has fallen foul of the firewall which says that web servers cannot make outbound connections. Please see Section [19.4](#).

Chapter 27

Reporting issues

The Symbiosis project, and its documentation both have issue trackers. Before these are explained, however, we have a few tips on how to help us help you.

Firstly, make sure that no-one else has reported the same problem. The issue trackers are public, and there is a search box to help you through them. The issue tracker is located at <https://projects.bytemark.co.uk/projects/symbiosis/issues>.

Bugs can also now be reported with Debian's "reportbug" utility.

Secondly, please use the following guidelines to make sure we have as much information as we need to diagnose, and hopefully fix the problem. We'd like to know the following:

Tracker There are two broad types of issue, each with its own tracker.

- Feature: a request for new functionality
- Bug: a problem with existing functionality

Subject It is very important to make this as descriptive, yet concise as possible. The following are examples of bad subjects

- help
- cannot login
- bug
- new feature Good subjects include
- Expected firewall update didn't occur
- Backups are failing to take place
- Add functionality to automatically white-list IPs in the firewall

Description A description of your problem; this should include copy and pasted log snippets, an explanation of what you were trying to do and what you expected to happen, and anything else you can think of that would be helpful. Always err towards lots of information rather than none: remember, there's a chance we'll only know as much about your problem as you tell us.

Part IV

Appendices

Appendix A

Email client setup

This chapter covers setting up email collection and delivery under Thunderbird 10.0, Apple Mail and Windows Live Mail.

A.1 Generic client configuration.

The following details might be needed when setting up a mail client to use an email account. The user of bob@my-brilliant-site.com on the machine `example.vm.bytemark.co.uk` has been chosen for these worked examples.

It is recommended that all communication with the mail server is conducted over encrypted connections, either using [SSL](#), or [TLS](#).

**Note**

By default a self-signed certificate is used during the secure transactions. Some mail clients may warn you about this certificate being invalid. If you are at all unsure about the validity of the certificate it would be prudent to double-check it.

Incoming email can be collected using either the [IMAP](#) or [POP3](#) protocols. IMAP is generally recommended over POP3 as it can handle folders, push notification, can selectively download message parts, and the email remains on the server enabling back-ups to be made.

Outgoing email is sent using [SMTP](#). It is good practice to send any outgoing email via the Symbiosis server, rather than any relay service provided by your ISP.

Managing sieve filters is done using the [ManageSieve](#) protocol.

For both sending and receiving email, the following login information would be used.

Username bob@my-brilliant-site.com

Password (contents of `/srv/my-brilliant-site.com/mailboxes/bob/password`)

Server name `mail.my-brilliant-site.com`

The default ports are used for all protocols. For further details see Section [16.1](#).

**Tip**

It is common for Internet service providers to block the standard outgoing email port, i.e. port 25. If your email client complains that it cannot connect to your server on this port, then port 587 is provided as an alternative.

A.2 Configuring Mozilla Thunderbird 10.0.

[Mozilla Thunderbird](#) is a popular open source mail client, that runs on a variety of platforms. In this program we can use the Create a new account link on the home screen to create our new account. Alternatively select File New Existing mail account. . . . In this worked example, Bob Example, who has the email address bob@my-brilliant-site.com will set up his email account.

1. Mail Account Setup: Enter your name as you would like it to appear in the From: header in the Your name box. Enter your email address in the Email address box. Bob uses bob@my-brilliant-site.com, and finally enter your password in the Password box. Click Continue.

Mail Account Setup

Your name: Your name, as shown to others

Email address:

Password:

Remember password

2. Thunderbird will then attempt to guess the configuration details using common server names. Once found, it is best to use IMAP to collect email as it allows storage of email in folders on the remote host, meaning that it will be displayed identically in webmail. Click Create account.

Mail Account Setup

Your name: Your name, as shown to others

Email address:

Password:

Remember password

Configuration found by trying common server names

IMAP (remote folders) POP3 (keep mail on your computer)


Incoming: IMAP, mail.my-brilliant-site.com, STARTTLS

Outgoing: SMTP, mail.my-brilliant-site.com, SSL (Warning: Could not verify server)

Username: bob

3. At this point Thunderbird might display a warning. As described in the previous chapter, this is due to the SSL certificate being self-signed, i.e. locally generated. This does not effect the security of the connections. Check I understand the risks, and then Create account

Mail Account Setup

 **Warning!**

Outgoing settings: mail.my-brilliant-site.com does not use a trusted certificate.

▶ Technical Details

Thunderbird can allow you to get to your mail using the provided configurations. However, you should contact your administrator or email provider regarding these improper connections. See the Thunderbird FAQ for more information.

I understand the risks.

4. Finally a further warning is displayed about the SSL certificate. Check Permanently store this exception, and then Confirm security exception.



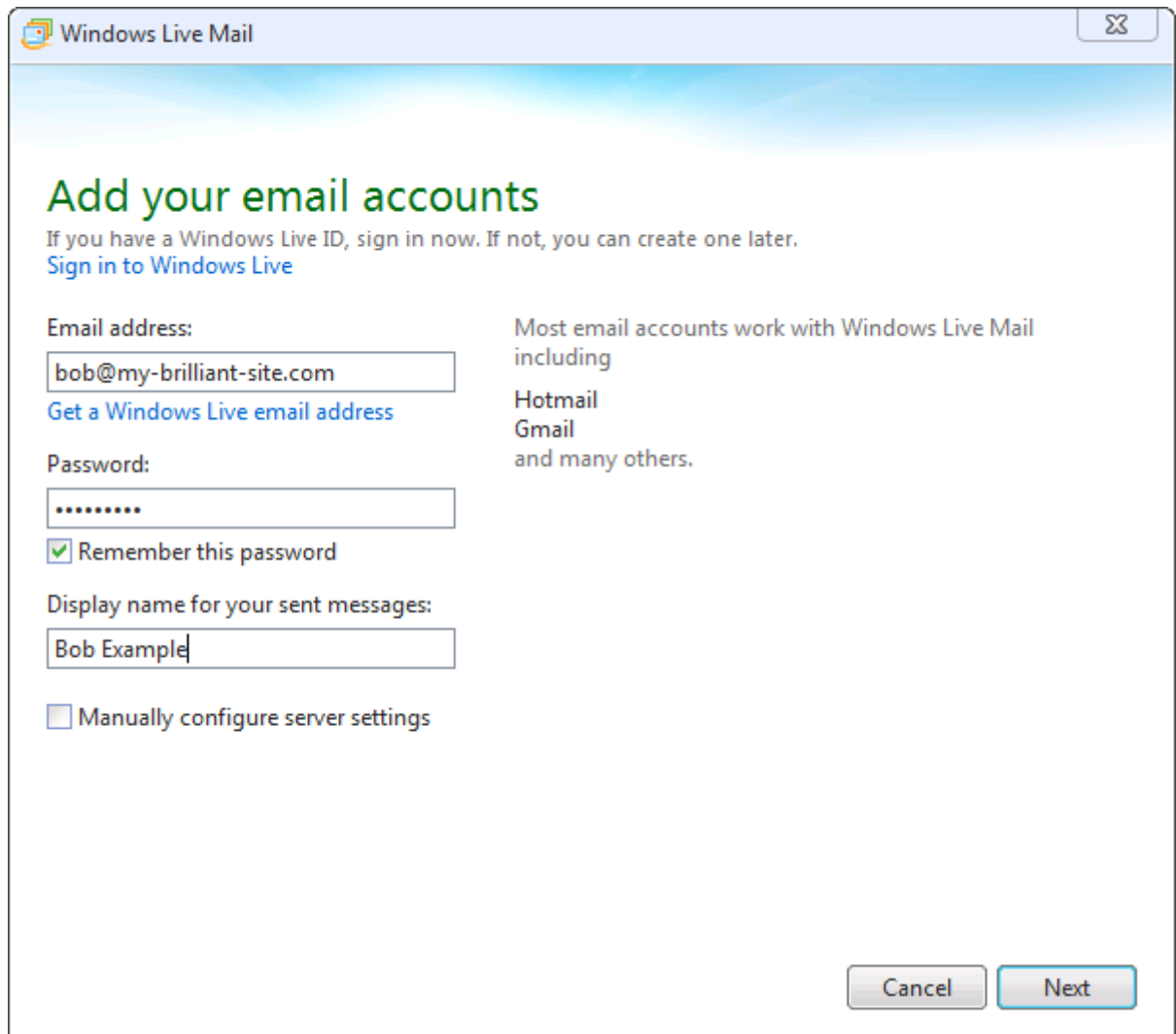
5. Your mail account is now ready to use.

A.3 Configuring Windows Live Mail

Firstly a new account needs to be added. If you've not used Windows Live Mail before, a wizard will automatically appear when you start the program for the first time. If you have used it before, click the Accounts tab and click Email.

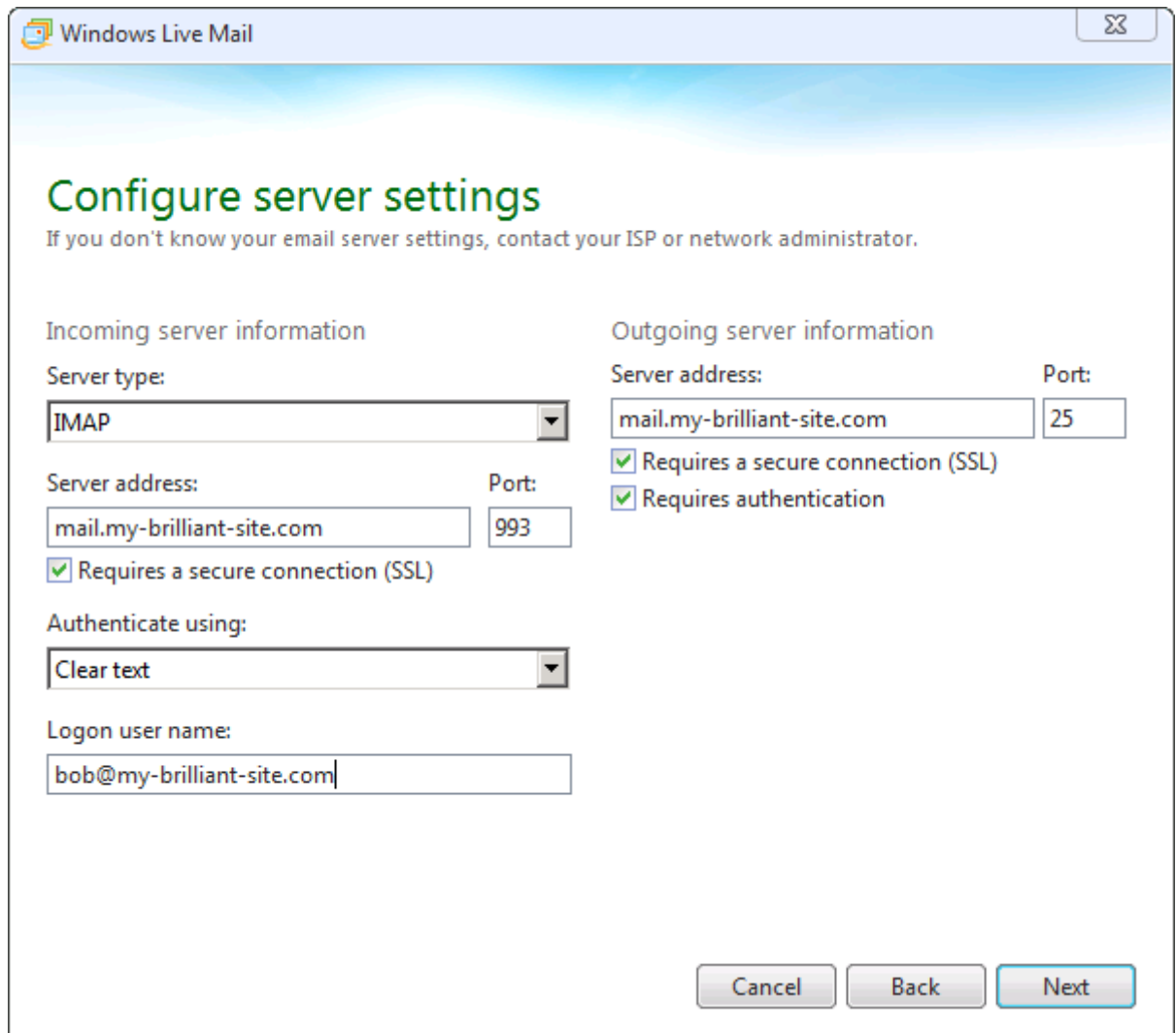
This walk-through uses the example person Bob Example, with the email address bob@my-brilliant-site.com.

1. Add your email accounts: fill in your email address, password, and display name. For this example, Bob would put bob@my-brilliant-site.com, his password and Bob Example



The screenshot shows the 'Add your email accounts' window in Windows Live Mail. The window title is 'Windows Live Mail'. The main heading is 'Add your email accounts' in green. Below it, there is a sub-heading: 'If you have a Windows Live ID, sign in now. If not, you can create one later.' and a link 'Sign in to Windows Live'. The form contains several fields: 'Email address:' with the value 'bob@my-brilliant-site.com' and a link 'Get a Windows Live email address'; 'Password:' with a masked password '*****' and a checked checkbox 'Remember this password'; 'Display name for your sent messages:' with the value 'Bob Example'; and an unchecked checkbox 'Manually configure server settings'. To the right of the form, there is text: 'Most email accounts work with Windows Live Mail including Hotmail Gmail and many others.' At the bottom right, there are 'Cancel' and 'Next' buttons.

2. Configure sever settings: Windows Live Mail can make guesses as to which server names to use for both incoming and outgoing servers. In this example `mail.my-brilliant-site.com` was automatically filled in. Make sure that Requires a secure connection (SSL) is ticked for both the incoming and outgoing servers. The incoming server should be set to authenticate using Clear text. The Logon user name is set to Bob's email address, i.e. `bob@my-brilliant-site.com`. Finally ensure that the Requires authentication box is checked for the outgoing server.



The screenshot shows the 'Configure server settings' window in Windows Live Mail. The window title is 'Windows Live Mail'. The main heading is 'Configure server settings' in green, with a subtitle: 'If you don't know your email server settings, contact your ISP or network administrator.'

The window is divided into two columns: 'Incoming server information' and 'Outgoing server information'.

Incoming server information:

- Server type: IMAP (dropdown menu)
- Server address: mail.my-brilliant-site.com (text box)
- Port: 993 (text box)
- Requires a secure connection (SSL)
- Authenticate using: Clear text (dropdown menu)
- Logon user name: bob@my-brilliant-site.com (text box)

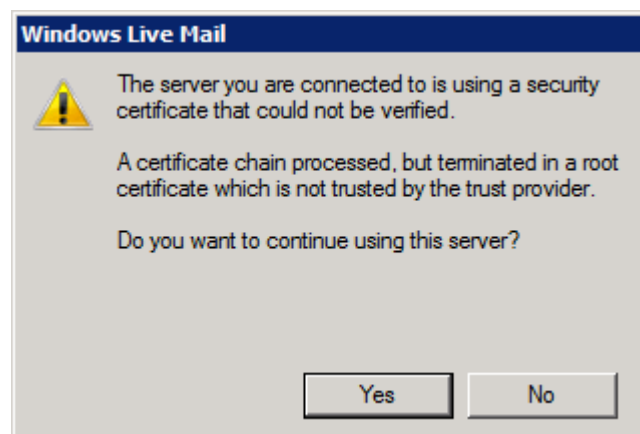
Outgoing server information:

- Server address: mail.my-brilliant-site.com (text box)
- Port: 25 (text box)
- Requires a secure connection (SSL)
- Requires authentication

At the bottom right, there are three buttons: 'Cancel', 'Back', and 'Next'.

3. Your email account was added: Click Finish to create the account.

When a connection is first made to the account, a warning box may pop up warning about the SSL certificate.



It is safe to answer Yes when asked if you wish to continue using the server.

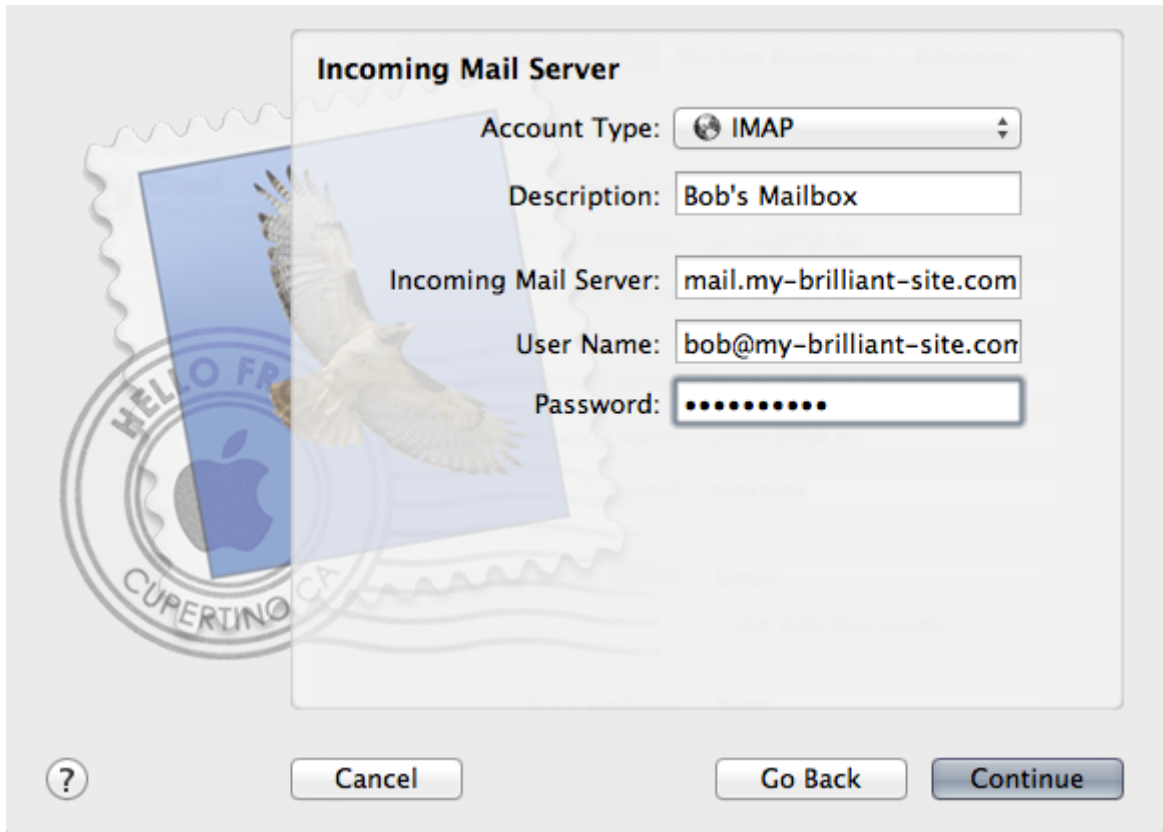
A.4 Configuring Apple Mail

Apple Mail is the standard email client that comes with an Apple's Mac OS X. This walk through uses Bob Example, who has the bob@my-brilliant-site.com email account.

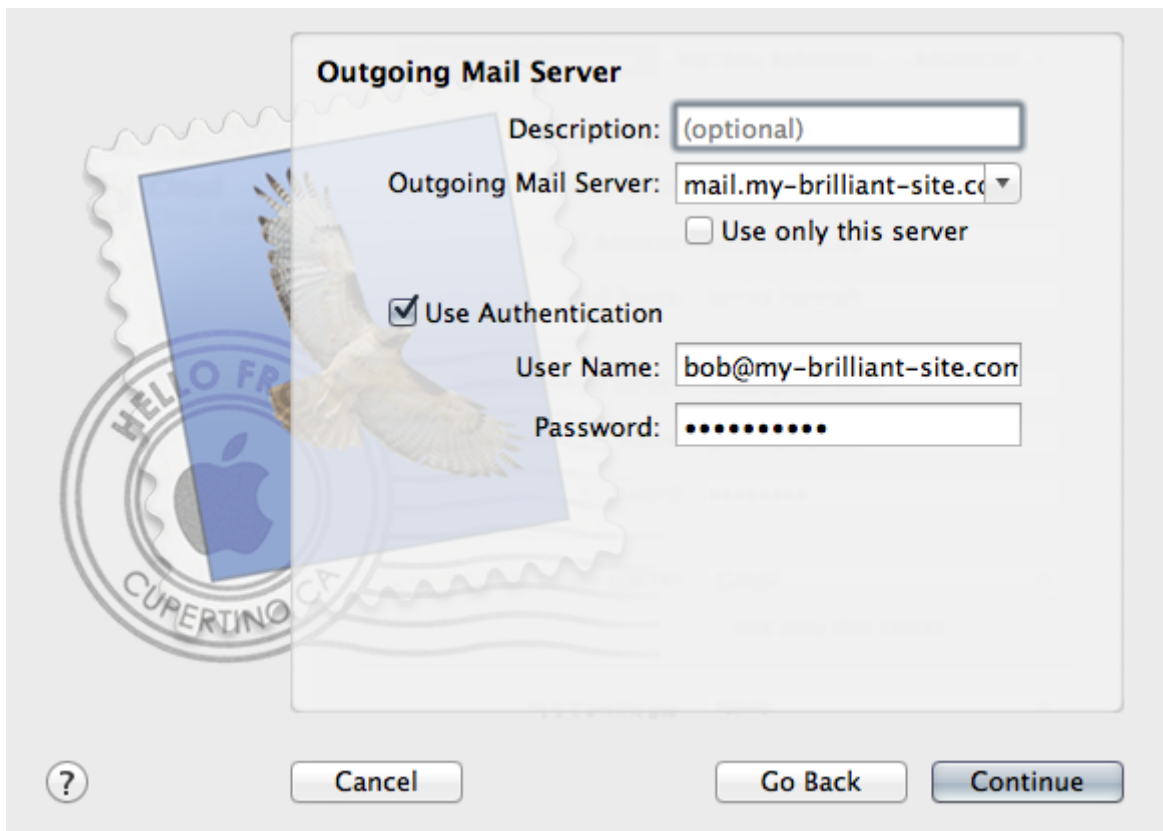
1. Add account: enter your name as you would like it displayed, your email address and password.



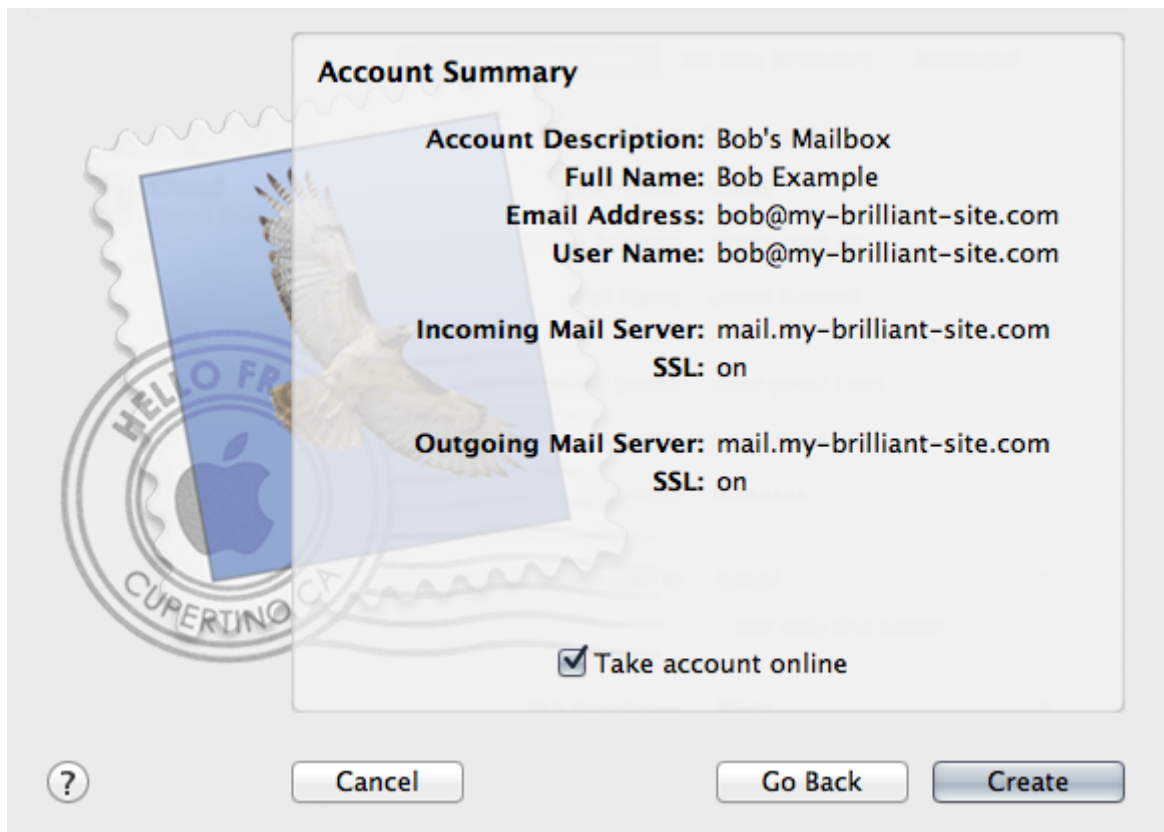
2. Incoming Mail Server: Set the account type to IMAP, the incoming mail server name, and the username, which is the same as your email address. In Bob's case, the incoming server name is `mail.my-brilliant-site.com`.



3. Outgoing Mail Server: Set the server name to the same as the incoming one was, and tick the Use Authentication box. The username and password should be the same as for your incoming mail. Bob sets his outgoing server to mail.my-brilliant-site.com.



4. Account Summary: This shows the settings that will be used for your account. Note that SSL is enabled for both incoming and outgoing servers, keeping your mail secure against prying eyes.



When you first connect to your server, you may get a certificate warning, similar to the one shown below.



It is safe to click Connect as the SSL certificate is self-signed.

Appendix B

XMPP client setup

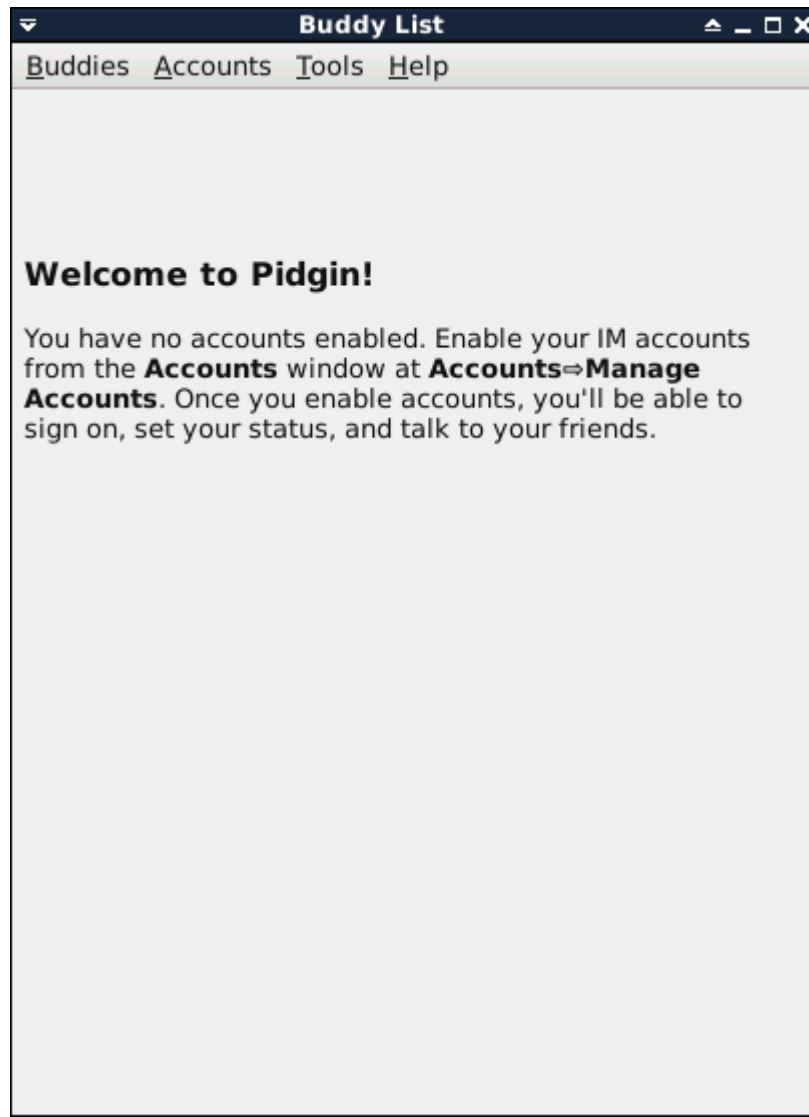
This section covers using Pidgin on Windows, Linux and OSX.

B.1 Pidgin XMPP client configuration.

To start using XMPP that Symbiosis provides, you'll need an XMPP client. Pidgin is one such client, and we'll be showing you how to add an account and a contact for this, complete with illustrative screen shots.

Pidgin is available from <https://pidgin.im/> and can be used on Linux, Windows, and OSX. Install it now if you haven't already, as that's what we'll be using in this guide. If you already have a preferred XMPP client it shouldn't be too hard to infer the settings from what follows.

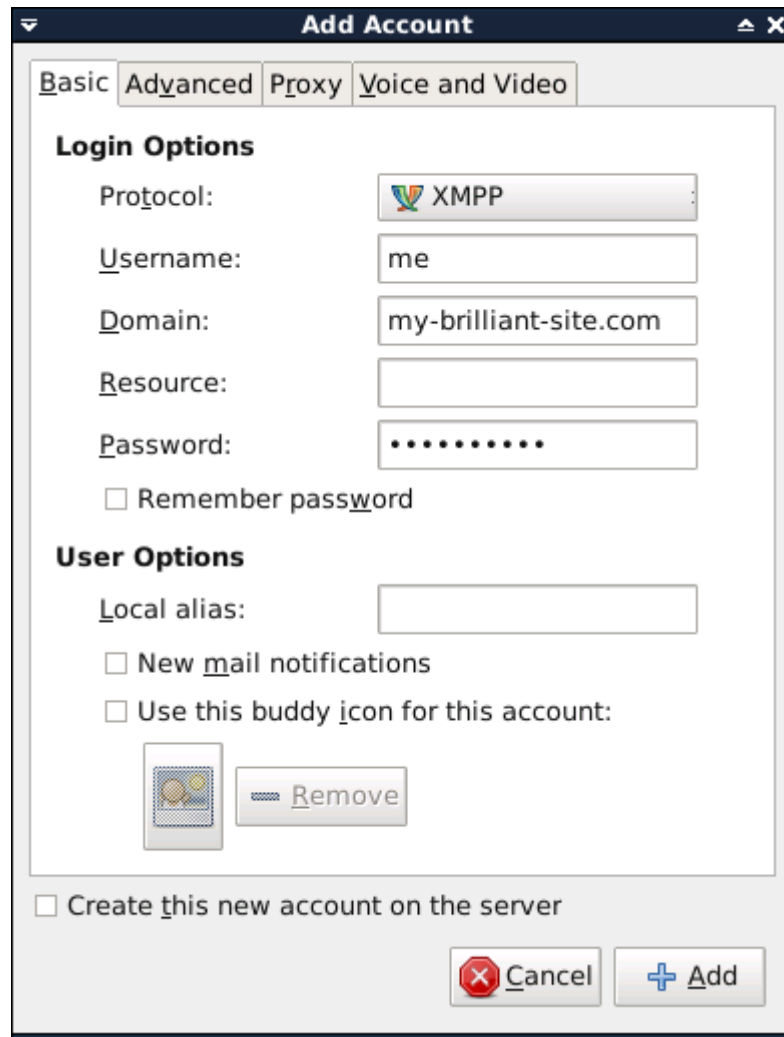
1. After starting Pidgin, go to the Accounts menu shown below. Select the Manage Accounts button from the dropdown.



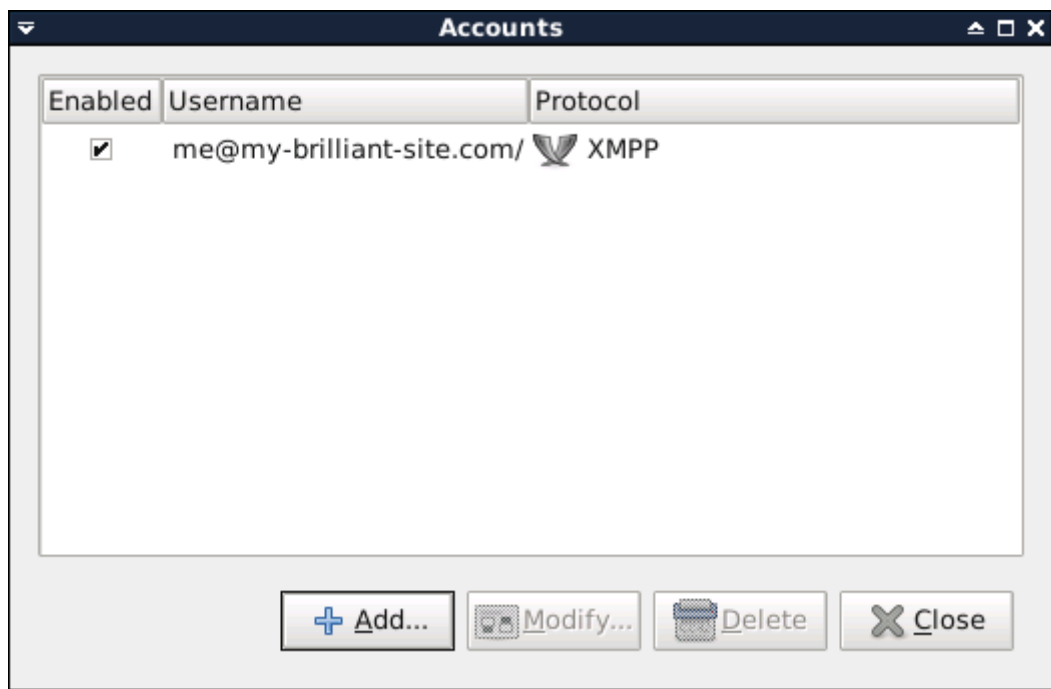
2. Manage Accounts should show a dialogue with a welcome screenshot, and an Add... button. Select this.



3. From the Protocol dropdown, select XMPP. Your Username will be the same as the part of your email address preceding the @ symbol, your Domain will be your domain name, and your Password will be the same as your email address password. Resource may be left blank; this identifies your client and Pidgin will set it for you.



4. Select Add once you've entered your details, and you'll subsequently be returned to the Manage Accounts dialogue. This will show the new account. If the tickbox in the Enabled column isn't ticked, do this.



5. Setup of your XMPP account is now finished - if you're presented with an SSL certificate warning, you can select `Accept`. The warning is because the default SSL certificate is self signed.

Appendix C

GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 [Free Software Foundation, Inc.](#)

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document “free” in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document’s overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical

connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The “Invariant Sections” are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not “Transparent” is called “Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

The “publisher” means any person or entity that distributes copies of the Document to the public.

A section “Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “Acknowledgements”, “Dedications”, “Endorsements”, or “History”.) To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.

- I. Preserve the section Entitled “History”, Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled “History” in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the “History” section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled “Acknowledgements” or “Dedications”, Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled “Endorsements”. Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled “Endorsements” or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version’s license notice. These titles must be distinct from any other section titles.

You may add a section Entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties — for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled “History” in the various original documents, forming one section Entitled “History”; likewise combine any sections Entitled “Acknowledgements”, and any sections Entitled “Dedications”. You must delete all sections Entitled “Endorsements”.

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled “Acknowledgements”, “Dedications”, or “History”, the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See [Copyleft](#).

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version” applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy’s public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

“Massive Multiauthor Collaboration Site” (or “MMC Site”) means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A “Massive Multiauthor Collaboration” (or “MMC”) contained in the site means any set of copyrightable works thus published on the MMC site.

“CC-BY-SA” means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

“Incorporate” means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is “eligible for relicensing” if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright © YEAR YOUR NAME
```

```
Permission is granted to copy, distribute and/or modify this document under the
terms of the GNU Free Documentation License, Version 1.3 or any later version
published by the Free Software Foundation; with no Invariant Sections, no
Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in
the section entitled “GNU Free Documentation License”.
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with... Texts.” line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts
being LIST, and with the Back-Cover Texts being LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Part V

Glossary

BSD, Berkeley System Distribution

A family of Unix versions developed by Bill Joy and others at the University of California at Berkeley, originally for the DEC VAX and PDP-11 computers, and subsequently ported to almost all modern general-purpose computers. BSD Unix incorporates paged virtual memory, TCP/IP networking enhancements and many other features [\[FOLDOC\]](#).

DKIM, Domain Keys Identified Mail

This adds a DKIM signature to each outbound email message on a system which can then be verified by recipients. Recipient SMTP servers will look up the DKIM selector of the mail, and verify that the key the mail is signed with matches the public key in DNS.

DNS, Domain Name System

This system is used to convert IP Addresses into hostnames. It is also used to determine where mail should be routed for a domain.

FTP, File Transfer Protocol

FTP used to be used to transfer large files over the internet. It is an archaic protocol.

FTPS, File Transfer Protocol Secure

FTPS is an extension to FTP that allows encryption using TLS or SSL. It is not to be confused with SFTP, which is a subsystem of SSH.

HTML, Hypertext Markup Language

A system to mark up documents. It is the most common format used for documents on the world-wide web, and is the format that web browsers display.

HTTP, Hypertext Transfer Protocol

This protocol was originally used to transfer HTML documents between machines connected to the internet. It has become the standard protocol for transferring all types of documents over the world-wide web.

IMAP, Internet Message Access Protocol

The Internet Message Access Protocol (IMAP) is one of the two most prevalent Internet standard protocols for e-mail retrieval, the other being the Post Office Protocol (POP). Virtually all modern e-mail clients and mail servers support both protocols as a means of transferring e-mail messages from a server.

IP, Internet Protocol

The network layer for the TCP/IP protocol suite widely used on Ethernet networks, defined in STD 5, RFC 791. IP is a connectionless, best-effort packet switching protocol. It provides packet routing, fragmentation and re-assembly through the data link layer.

+ IPv4 is the version in widespread use and IPv6 was just beginning to come into use in 2000 but was still not widespread by 2008 [\[FOLDOC\]](#).

IP Address

IP addresses come in two flavours, reflecting the two versions of *IP* used.

+ An IPv4 address is a 32 bit number generally represented as a dotted quad e.g. 10.20.30.40. There is a limit of just under 4.3 billion IPv4 addresses, which is slowly being reached, which necessitated the invention of IPv6.

+ An IPv6 address is a 128 bit number, generally represented as a hexadecimal number, split into nibbles of up to four digits, separated by colons, e.g. 2001:41c8:12::34. There are up to 2^{128} or 3×10^{38} addresses available in IPv6.

ISP, Internet Service Provider

A company which provides other companies or individuals with access to, or presence on, the Internet. Most ISPs are also Internet Access Providers; extra services include help with design, creation and administration of World-Wide Web sites, training and administration of intranets and domain name registration [\[FOLDOC\]](#).

ManageSieve

ManageSieve is a protocol that allows [Sieve](#) filters to be managed remotely, testing any filters before allowing them to be used.

MTA, Mail Transfer Agent

A mail transfer agent is a computer process or software agent that transfers electronic mail messages from one computer to another, in single hop application-level transactions. A MTA implements both the client (sending) and server (receiving) portions of the Simple Mail Transfer Protocol.

MUC, Multi User Chat

A Multi User Chat is a feature of XMPP allowing many users to converse in the same window. This is often used to ease communication between groups in different offices, and for the sake of ease can be thought of as the point at which mailing lists and instant messages meet.

NTP, Network Time Protocol

A protocol built on top of TCP/IP that assures accurate local timekeeping with reference to radio, atomic or other clocks located on the Internet. This protocol is capable of synchronising distributed clocks within milliseconds over long time periods.[\[FOLDOC\]](#).

PHP

PHP is a widely-used general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. [\[PHPNET\]](#)

POP3, Post Office Protocol 3

Version 3 of the Post Office Protocol. POP3 is defined in RFC 1081, written in November 1988 by Marshall Rose, which is based on RFC 918 (since revised as RFC 937). POP3 allows a client computer to retrieve electronic mail from a POP3 server via a (temporary) TCP/IP or other[?] connection. It does not provide for sending mail, which is assumed to be done via SMTP or some other method [\[FOLDOC\]](#).

Secure File Transfer Protocol, SFTP

SFTP is a file transfer protocol which involves using an SSH server to manage the file uploads. It is secure in the sense that file contents are encrypted during transfer, and that plain-text passwords are never sent over the internet. SFTP is the logical successor to FTP, which is less secure, and more complex to firewall.

Sieve

Sieve is a language that can be used to filter email messages. It is a powerful language that provides a safe environment for filtering to occur during mail delivery, allowing messages to be delivered directly into mailboxes configured by the user.

SMTP, Simple Mail Transfer Protocol

A protocol defined in STD 10, RFC 821, used to transfer electronic mail between computers, usually over Ethernet. It is a server to server protocol, so other protocols are used to access the messages [\[FOLDOC\]](#).

SPF, Sender Policy Framework

An anti-spam measure designed to let domain administrators choose how mail sent on their domain's behalf will be treated by recipients, which can help send spoofed mail to spam and protect your domain's reputation.

SSH, Secure Shell

A Unix shell program for logging into, and executing commands on, a remote computer. ssh is intended to replace rlogin and rsh, and provide secure encrypted communications between two untrusted hosts over an insecure network. X11 connections and arbitrary TCP/IP ports can also be forwarded over the secure channel [\[FOLDOC\]](#).

SSL, Secure Sockets Layer

A protocol designed by Netscape Communications Corporation to provide secure communications over the Internet using asymmetric key encryption. SSL is layered beneath application protocols such as HTTP, SMTP, Telnet, FTP, Gopher and NNTP and is layered above the connection protocol TCP/IP. It is used by the HTTPS access method [\[FOLDOC\]](#).

TCP, Transmission Control Protocol

The most common transport layer protocol used on Ethernet and the Internet. It was developed by DARPA.

TCP is the connection-oriented protocol built on top of Internet Protocol (IP) and is nearly always seen in the combination TCP/IP (TCP over IP). It adds reliable communication and flow-control and provides full-duplex, process-to-process connections.

TCP is defined in STD 7 and RFC 793 [\[FOLDOC\]](#).

TLS, Transport Layer Security

A protocol designed to allow client/server applications to communicate over the Internet without eavesdropping, tampering, or message forgery.

TLS is defined in RFC 2246 [\[FOLDOC\]](#).

UDP, User Datagram Protocol

Internet standard network layer, transport layer and session layer protocols which provide simple but unreliable datagram services. UDP is defined in STD 6, RFC 768. It adds a checksum and additional process-to-process addressing information [to what?]. UDP is a connectionless protocol which, like TCP, is layered on top of IP.

UDP neither guarantees delivery nor does it require a connection. As a result it is lightweight and efficient, but all error processing and retransmission must be taken care of by the application program [\[FOLDOC\]](#).

URL, Uniform Resource Locator

A Uniform Resource Locator (URL) is a Uniform Resource Identifier (URI) that specifies where an identified resource is available and the mechanism for retrieving it. In popular usage and in many technical documents and verbal discussions it is often incorrectly used as a synonym for URI. The best-known example of a URL is the "address" of a web page e.g. <http://www.example.com> [\[WIKIPEDIA_URL\]](#).

XMPP, Extensible Messaging and Presence Protocol

A protocol enabling instant messaging, contact list maintenance, and presence information. Addresses usually take the same form as an email address, eg, user@domain.tld. Various common extensions exist, including file transfer, voice and video (*Jingle*), service discovery, and multi user chat. Federation is another key feature of XMPP, which allows any user of XMPP to contact any other user, provided they are able to connect that user's XMPP server.

Part VI

Bibliography

27.1 Bibliography

- [1] [FOLDOC] Denis Howe (ed). 'The Free On-line Dictionary of Computing', <http://foldoc.org/>
- [2] [PHPNET] The PHP Group. 'PHP: Hypertext Preprocessor', <http://php.net/>
- [3] [WIKIPEDIA_URL] Wikipedia contributors. 'Uniform Resource Locator', Wikipedia, The Free Encyclopedia. http://en.wikipedia.org/w/index.php?title=Uniform_Resource_Locator&oldid=367676813 (downloaded 2010-06-10).

Chapter 28

Index

B

- Backups, [43](#)
 - Offsite storage, [44](#)
 - upgrade related change, [58](#)

C

- Chat
 - configuration, [34](#)
- config/
 - aliases, [70](#), [73](#)
 - antispam, [31](#), [74](#)
 - antivirus, [31](#)
 - blacklists/
 - pbl.spamhaus.org, [74](#)
 - sbl-xbl.spamhaus.org, [74](#)
 - sbl.spamhaus.org, [74](#)
 - xbl.spamhaus.org, [74](#)
 - zen.spamhaus.org, [74](#)
 - bytemark-antispam, [73](#)
 - crontab, [41](#)
 - default_forward, [68](#), [69](#), [73](#)
 - dkim, [88](#)
 - dkim.key, [88](#)
 - dns/
 - my-brilliant-site.com.txt, [85](#)
 - ftp-password, [35](#)
 - ftp-quota, [38](#)
 - ip, [52](#)
 - mailbox-quota, [69](#), [74](#)
 - no-stats, [63](#), [66](#)
 - spf, [87](#)
 - ssl-only, [54](#), [66](#)
 - ssl.bundle, [53](#)
 - ssl.crt, [53](#), [65](#)
 - ssl.csr, [52](#)
 - ssl.key, [52](#)
 - webalizer.conf, [63](#), [66](#)
 - xmpp, [34](#)

Connecting

- FileZilla, [5](#)
- SFTP, [5](#)
- SSH, [47](#)

Crontab

- format, [41](#)
- Output, [42](#)
- Testing, [89](#)
- using, [41](#)

D

Database

- adding a remote user, [91](#)
- enabling remote access, [91](#)
- opening the firewall for, [91](#)
- root password, [91](#)

DKIM

- setup, [87](#)

DNS records

- DKIM, [87](#)
- example, [85](#)
- hostname wild-card, [87](#)
- SPF, [87](#)

Domains

- adding, [25](#)
- moving between machines, [87](#)
- testing, [26](#)
- wildcard, [26](#)

E

Email

- accepting, [67](#)
 - unix, [68](#)
- aliases, [70](#)
- catching all, [67](#)
- client configuration, [108](#)
 - Apple Mail, [114](#)
 - Mozilla Thunderbird, [109](#)

- Windows Live Mail, [111](#)
- configuration layout, [73](#)
- creating a new mailbox, [27](#)
- encrypting passwords, [68](#)
- filtering using Sieve, [30](#)
- forwarding, [30](#), [69](#)
 - a whole domain, [69](#)
 - keeping a copy, [70](#)
- mailboxes
 - creating, [27](#)
- Manually defined blocklists, [72](#)
- not accepting any, [68](#)
- port numbers, [67](#)
- quotas, [69](#)
- scanning for spam and viruses, [30](#)
- spam headers, [71](#)
- Spamhaus blocklists, [71](#)
- using suffixes, [69](#)
- vacation messages, [70](#)

F

FileZilla

- Common recipes, [7](#)
- Connecting using, [5](#)
- Creating a remote directory, [11](#)
- Creating a remote file, [14](#)
- Deleting files and directories, [18](#)
- Navigating local and remote filesystems, [8](#)

Firewall

- accessing services, [78](#)
- adding custom rules, [82](#)
- automatic blacklist, [82](#)
- automatic whitelist, [83](#)
- blacklist, [45](#)
- configuration layout, [84](#)
- disabling, [84](#)
- example configuration, [80](#)
- patterns used for blacklisting IPs, [82](#)
- predefined rules, [79](#)

FTP access

- setting a quota, [38](#)
- setting the password, [35](#)
- testing, [36](#)

H

HTTPS

- force redirection to, [54](#)

I

IMAP

- new dovecot, [58](#)

IP addresses

- adding, [51](#)

J

Jabber

- configuration, [34](#)

Mmailboxes/, [27](#), [73](#)

- user/, [73](#)
 - forward, [30](#), [69](#), [73](#)
 - Maildir/, [73](#)
 - password, [28](#), [73](#)
 - quota, [69](#), [73](#)
 - sieve, [69](#), [73](#)
 - sieve.d, [69](#)
 - vacation, [70](#), [73](#)
 - vacation.db, [70](#)
 - vacation.log, [70](#)

MySQL

- adding a remote user, [91](#)
- enabling remote access, [91](#)
- managing, [39](#)
- opening the firewall for, [91](#)
- root password, [91](#)

P

phpMyAdmin

- connecting to MySQL, [39](#)

public/, [35](#)

- cgi-bin/, [63](#), [66](#)
- htdocs/, [23](#), [25](#), [66](#)
 - stats/, [63](#), [66](#)
- logs/
 - access.log, [66](#)
 - error.log, [66](#)
 - ssl_access.log, [66](#)
 - ssl_error.log, [66](#)

PuTTY

- connecting to your machine using, [47](#)

Q

Quotas

- email, [69](#)
- FTP, [38](#)

Rroot user, [60](#)**S**Security, [45](#)

- Avoiding weak passwords, [45](#)

- Check system email notices, [45](#)
- Keep your software current, [46](#)
- SpamAssassin, [71](#)
- Spamhaus, [71](#)
- SPF
 - adding records, [87](#)
- SquirrelMail
 - logging in, [31](#)
- SSH
 - connecting to your machine using PuTTY, [47](#)
- SSL
 - making mandatory, [54](#)
- SSL certificates
 - certificate request generation, [52](#)
 - configuring hosting, [51](#)
 - CSR generation, [52](#)
 - encryption using, [64](#)
 - key generation, [52](#)
 - purchasing, [53](#), [64](#)
 - self-signed, [64](#)
 - verifying, [65](#)
 - trusting, [64](#)
 - uploading, [53](#)
- Symbiosis
 - Command-line access, [2](#)
 - components, [59](#)
 - Features, [2](#)
 - Free software, [3](#)
 - Included software, [3](#)
 - installing, [56](#)
 - Open Source, [3](#)
 - upgrading, [56](#)
 - Web interface, [2](#)
 - wheezy release, [58](#)
- Sysadmin, [59](#)
- W**
- Web pages
 - default, [22](#)
 - testing, [26](#)
 - uploading, [22](#)
- Webmail
 - logging in to, [31](#)
- Website
 - access logs, [65](#)
 - CGI scripts, [62](#)
 - Configuration, [62](#)
 - configuration layout, [66](#)
 - Custom configuration, [64](#)
 - error logs, [65](#)
 - redirecting to a preferred hostname, [64](#)
 - statistics, [63](#)
 - customising, [63](#)
 - disabling, [63](#)
 - testing new sites, [63](#)
- X**
- XMPP
 - client configuration, [117](#)
 - configuration, [34](#)