

Set Up and Manage Your Virtual Private Server

Making System Administration
Accessible to Professionals

Jon Westfall

Apress®

SET UP AND MANAGE YOUR VIRTUAL PRIVATE SERVER

MAKING SYSTEM ADMINISTRATION
ACCESSIBLE TO PROFESSIONALS

Jon Westfall

Apress®

Set Up and Manage Your Virtual Private Server: Making System Administration Accessible to Professionals

Jon Westfall
Cleveland, MS, USA

ISBN-13 (pbk): 978-1-4842-6965-7

ISBN-13 (electronic): 978-1-4842-6966-4

<https://doi.org/10.1007/978-1-4842-6966-4>

Copyright © 2021 by Jon Westfall

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr

Acquisitions Editor: Shiva Ramachandran

Development Editor: Matthew Moodie

Coordinating Editor: Nancy Chen

Cover designed by eStudioCalamar

Distributed to the book trade worldwide by Springer Science+Business Media New York, 1 New York Plaza, New York, NY 100043. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at <http://www.apress.com/bulk-sales>.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/9781484269657. For more detailed information, please visit <http://www.apress.com/source-code>.

Printed on acid-free paper

*Dedicated to my parents, Alan and Dianne,
who bought me my first computer.*

Contents

About the Author	vii
About the Technical Reviewer	ix
Acknowledgments	xi
Introduction	xiii
Chapter 1: What Is a Virtual Private Server?	I
Chapter 2: Selecting a VPS Provider	17
Chapter 3: Basic Linux Administration via the Command Line	43
Chapter 4: Basic Linux Administration via GUI (Webmin)	77
Chapter 5: Basics of Linux Security	111
Chapter 6: Basics of Backups and Version Control	133
Chapter 7: Basics of Server Software Administration	153
Chapter 8: Installing an Open Source Software Product: WordPress	179
Chapter 9: Administrator's Weekly Checklist	199
Chapter 10: Open Source Application Possibilities	217
Index	241

About the Author



Jon Westfall is an award-winning professor, author, and practicing cognitive scientist. He teaches a variety of courses in psychology, from introduction to psychology to graduate seminars. His current research focuses on the variables that influence economic and consumer finance decisions, as well as retention and persistence of college students. With applications to psychology, information technology, and marketing, his work finds an intersection between basic and applied science. His current appointments include Associate Professor of Psychology, Coordinator of the First Year Seminar program, and Coordinator of the Psychology program at Delta State University. Prior to joining the faculty at Delta State in 2014, he was a Visiting Assistant

Professor at Centenary College of Louisiana and the Associate Director for Research and Technology at the Center for Decision Sciences, a center within Columbia Business School at Columbia University in New York City. He now maintains a role with Columbia Business School as a research affiliate/variable hours officer of administration and technology consultant.

In addition to his research, Dr. Westfall also has career ties in information technology, where he has worked as a consultant since 1997, founding his own firm, Bug Jr. Systems. As a consultant, he has developed custom software solutions (including native Windows 32 applications, Windows .NET applications, Windows Phone 7 and Android mobile applications, as well as ASP, ASP.NET, and PHP web applications). He has also served as a senior network and systems architect and administrator and been recognized as a Microsoft Most Valuable Professional (MVP) 2008–2012. He currently is the owner and managing partner of Secure Research Services, LLC. He has authored several fiction and nonfiction books and presented at academic as well as technology conferences and gatherings. A native of Ohio, in his spare time he enjoys knitting, crocheting, creative writing with the Delta Writers Group, and a variety of other hobbies.

For more, visit jonwestfall.com, listen to him weekly on the MobileViews podcast (mobileviews.com), or follow him on Twitter: [@jonwestfall](https://twitter.com/jonwestfall).

About the Technical Reviewer



Dr. Evan Hill is an associate professor of psychology and researcher in the field of comparative psychoacoustics. The courses he instructs at the University of Nebraska at Kearney cover various aspects of the biological basis of behavior and neural functioning. His research focuses primarily on species-level differences in hearing ability, with an emphasis on identifying structural, physiological, and ultimately functional explanations for these differences. Because his research frequently involves the use of novel species not commonly used in laboratory settings, almost everything is a DIY project. In any given study, he often must operate as a carpenter, electrical engineer, computer programmer, biologist, and veterinarian to get results.

Acknowledgments

Writing a book is never an easy task and is seldom the task of just one individual, even in a sole-author work. I am indebted to my wife, Karey, for her support throughout this process. I'm very lucky to have a soul mate who "gets" why I'm passionate about crazy topics like Virtual Private Servers (VPSs)!

I'm also thankful to my parents, Alan and Dianne, who instilled a love of reading and learning in me early in my life. Writing is only possible after spending a ton of time reading, whether it be fiction or nonfiction. I'm thankful to my friends who have listened to me talk about this project and provided feedback, including Steve Jocke, Jason Dunn, Matt Rozema, and my longtime podcasting partner Todd Ogasawara. Other friends who have supported me without knowing it (by providing inspiration for projects or stories in this book) include Christy Riddle, Tricia Killebrew, Kristen Land, Darla Poole, Kesha Pates, Jontil Coleman, Elise Mallette, Jackie Goldman, Andrés García-Penagos, Sally Zengaro, and many others in the extended Delta State family. I'd also like to acknowledge my students, for whom many of these applications originally were designed, for inspiring me to continue creating. I also am lucky to have the support of those at Apress, especially Nancy Chen, Shiva Ramachandran, and Matt Moodie.

On a personal level, I'd also like to thank the Delta Writers Group (Michael Koehler, Katy Koehler, Jason Hair, and Dick Denny) for providing a ground for sharpening my writing and critique skills regularly. I also gain so much support from friends and family, including Nate and Kristen Toney, Sarah Speelman, Maggie Ditto, Heather Hudgins, Ashley Newman, Dan, Sue, Scott, Emily, Greg, Janet, Mark, and Brenda Himmel, Margaret Lee, Christine and Carl Morris, Don Sorcinelli, Tony Rylow, Trella Williams, Eric Johnson, Elke Weber, Karen Fosheim, Carol Beard, Maria Gaglio, Hope Hanks, Tom Brady, and many others.

Introduction

“Just build it yourself” is a very ambitious statement, no matter what your art form. Another ambitious statement, “If you want it done right, do it yourself,” seems to make even the most difficult task seem trivial enough that anyone could accomplish it. Is it really that easy?

When it comes to having a professional web page or software to manage your customer list and contacts or a collaborative effort like a wiki, doing it yourself can often seem daunting even to someone proficient in using a computer. People tell themselves “it’s too complex” or “it’s too expensive” or both. However, it is easier and more inexpensive today to turn these services into a DIY project, as long as you have a road map. That’s the purpose of this book – to take you from someone who can use a computer to someone who can safely and effectively run their own virtual private server. Not only is this likely to be cheaper than commercial alternatives, it also provides a lot of peace of mind, knowing exactly who owns and stores your data.

If your initial thought was along the lines of cost and time, you are correct that there was a time when running your own server was a very expensive proposition that required a full-time IT staff. Today, thanks to virtualization, or the ability to host multiple self-contained servers on one large physical server, the only barrier to running your own server is your own knowledge. While formidable to learn everything you need to know, we’re also living in a time where resources are plentiful. This book not only gives you the basic “lingo” you need in order to get your own server up and running, it also empowers you to find solutions to problems as you grow in your technical knowledge. We follow a logical approach that first tells you what a Virtual Private Server is and how it can be available at such a low price. We then take an entire chapter (Chapter 2) to discuss potential service providers out there and how to “vet” one to find if it’s a good fit. Some cater more toward absolute beginners and charge a bit more for the handholding they do, while others are bargain basement dealers who require you to be at the top of your technical game. After selecting a provider, we then spend two chapters (Chapters 3 and 4) discussing the Linux operating system and how it works a bit differently than a graphical environment you may be familiar with. By the time we hit the middle of the book, we’re going to ease you into topics like security and backup. Finally, we finish by showing off some of the things you can do with a Virtual Private Server, including hosting your own web pages, applications, and other fun stuff.

Along the way, you'll find a lot of examples from over 20 years of my own personal and professional explorations. I started off in IT; however, in the mid-2000s, I pivoted to a decidedly non-IT career: a research psychologist. It is my hope that the explanations that I provide will not only be easy to understand (as I'm aiming to do quite a bit of translation from "tech geek" to "normal person"!), you'll also see the clear applications for each topic – because this stuff isn't useful in the abstract, it gains its real value when you use it to speed up and simplify your life! Before we can get to the cool stuff, though, we'll need to cover the basics – namely, what is a "Virtual Private Server"?

What Is a Virtual Private Server?

Let me share a story with you that illustrates what this book can do for you in terms of personal productivity. In other words, how you can save yourself time, money, stress, and more by learning the skills this book introduces. Join me on a journey into a “psychology professor’s” life for a moment.

Several months ago I had a problem, and my solution was a bit unorthodox. The problem was that people were always asking me for computer help, which was somewhat understandable since I am the Internet Editor for the Society for the Teaching of Psychology (STP). And while I like helping people, I was finding it hard to keep things organized. Who was I working with this week? What requests did they have? What about the associate editors – could I delegate this task? And what about my boss and my collaborators – were they looped in? It occurred to me that a help desk ticket system, similar to what many companies use to track customer support requests, would be useful for our group. It would let me track who I was helping and how quickly they were helped and also allow me to add collaborators easily.

So I installed one and we started using it...that day...for free (see Figure 1-1).

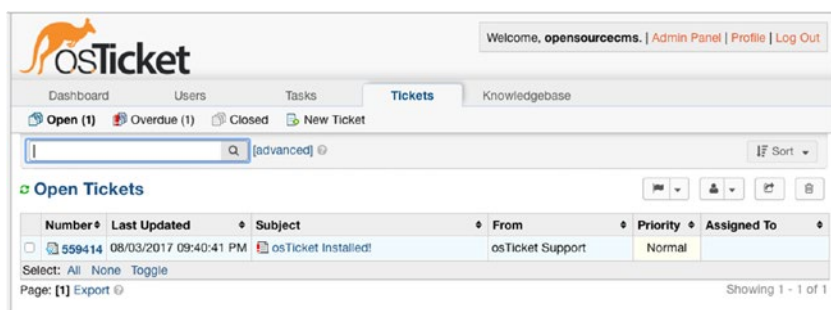


Figure I-1. A demo version of osTicket hosted on OpenSourceCMS.com, the product I installed on my own server

This sounds crazy to most people – after all, surely there must be a lot of steps involved in doing something like this. Setting up a website that can take requests and track them, making email talk to that website, setting up a way to log in and reply to the tickets, and dozens more smaller tasks. Plus you’d need a computer that was always on and connected to the Internet; otherwise, when replies to your tickets came in, how would they get to the ticket software? And to do all of this for free? It may seem like a miracle.

But it isn’t – it’s something that, by the end of this book, you may feel comfortable pulling off yourself. Will it take you less than a day? Probably not. But less than a weekend is certainly a possibility. You can even have time for meals and sleep. This book is about giving you the tools to pull off these tiny miracles and the flexibility to know how to use those tools in unique and novel ways. In this book, we’re going to talk about Virtual Private Servers – what they are, how they work and are managed, and, most importantly, how they can change your professional and personal life for the better. Soon you’ll be setting up your own Internet world!

A Brief Understanding of the Internet and Computers Connected to It

I try not to assume much in this book; however, I am about to make an assumption – you know what a computer is. If by some chance you don’t, then you may be in the wrong section of the bookstore! What you may not know, though, is that the websites that you visit every day, the programs and apps that you use on your phone, and the software that runs your intranet at work all run off of computers that are actually very similar to your desktop or laptop. They aren’t “special” in any way other than perhaps having more storage space or processing power connected to them and the fact that they’re rarely, if ever, turned off.

And the Internet is full of them – millions of computers that all connect to each other. The majority of those computers are *clients* – they are seeking information or services from *servers*, computers that provide information or services. When you open your smartphone and launch the browser, your smartphone becomes a client, likely opening a web page that lives on a server. The lines aren't mutually exclusive though – the server that your web page lives on may need to download information from another server in order to display the page. Thus, to you it is a server, but to the other computer it is a client. Imagine the Internet as merely a series of trillions of client-server interactions happening every day. Client asks for something, and server sends it (see Figure I-2).

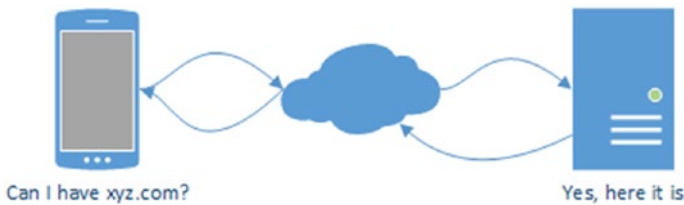


Figure I-2. A flowchart of your computer requesting a web page

Now that we've established what a server is, what is this whole "virtual" stuff that you hear thrown around in places such as the title of this book? Well, let's use another analogy – imagine a shopping mall. Inside we have 100 stores, each with their own entrances, their own staff, their own products, and their own policies for sales and returns. They all live in the same physical structure, and so all of them have to abide by the same global rules – if the mall ownership says that stores must open by 10 AM, then the smaller stores need to open by 10 AM. If the ownership says that you can stay open as late as 9 PM, then stores have permission to close early if they like, but can't go past 9 PM.

Let's call the shopping mall the "physical server" and all of the stores "virtual servers." Each store is unique and separated – the owner of one can't walk into the other without permission. Individual owners can decide to sell whatever they want, but they can't exceed the physical limitations that the mall has stipulated. In the same way, a very powerful computer server can house many less powerful virtual servers. Each virtual server runs with a subset of resources devoted to it and is isolated from the other servers running.

Translating all of this together, a virtual private server is a computer connected to the Internet that is cheaper to "rent" than a physical server, because multiple people can rent space on the same physical system. The owners of those systems, the hosting company or service, set the parameters of what each virtual server can do: how much processing power it gets, how much

storage space, and how much bandwidth – think foot traffic in our mall scenario – is allowed. As long as you pay your bill and follow the rules, the owner of the physical server is happy to have you as a client (see Figure 1-3).

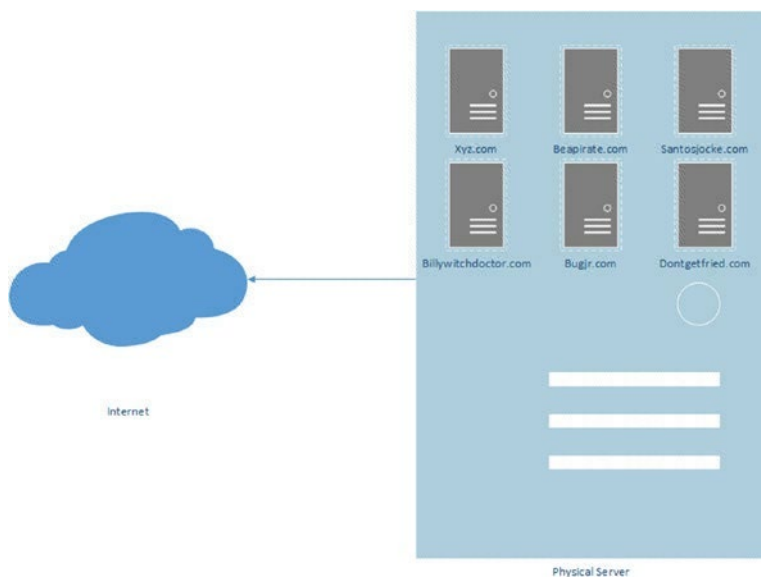


Figure 1-3. A physical server hosting multiple virtual private servers

Now that you know what a virtual private server is (smaller computer nested inside a much larger computer!), I’m going to discuss what exactly happens when you decide to open a web page, using our client-server terminology and imagining that the web page is hosted on a virtual private server.

Opening a Web Page: How the Magic Happens

If you’ve been using the Internet for the majority of your life, you probably don’t think too much about how it works because you’re used to it...well...just working! However, those of us who remember the “good old days” of the Internet (which weren’t that good, but were old) may remember the lag involved in loading pages prior to broadband and wondered “Why does it take so long?” Let’s dig into what happens when you try to view a web page, and then we can talk about the parts of it that you, as the owner of the web page, can control or tweak.

Imagine for a moment that you’re interested in visiting a fictitious website after a vivid daydream of quitting your day job, *beapirate.com* (for the pirate enthusiasts of the world). You open your web browser, and you enter the

address in the address bar at the top. You press Enter or click the Go button, and within a few minutes, you're being enticed to join a pirate ship and sail the open seas. But what occurred between the browser address bar and the sales pitch? Let's dig in.

First: Where Does beapirate.com Live?

The first thing that your computer did was to ask a Domain Name Service (DNS) server if it knew the numerical Internet Protocol (IP) address that beapirate.com lives at. Roughly analogous to looking up an address in a phone book, the DNS server took a look at the DNS records for beapirate.com (see Figure 1-4) that are held with various registrars around the world and found the DNS server that has authority over beapirate.com. Once it found that server, it was able to ask it directly for the address, and hopefully that server gave it that address without fail.

Assuming that the DNS settings are correctly configured, your computer got the numerical address it needed, and it moved on to the next step. However, there are ways that this can go wrong.

Records

Last updated 8/25/2020 10:03 AM

Type	Name	Value	TTL
A	@	Parked	600 seconds
CNAME	e	email.secureserver.net	1 Hour
CNAME	email	email.secureserver.net	1 Hour
CNAME	ftp	@	1 Hour
CNAME	imap	imap.secureserver.net	1 Hour
CNAME	mail	pop.secureserver.net	1 Hour
CNAME	mobilemail	mobilemail-v01.prod.mesa1.secureserver.net	1 Hour
CNAME	pda	mobilemail-v01.prod.mesa1.secureserver.net	1 Hour
CNAME	pop	pop.secureserver.net	1 Hour
CNAME	smtp	smtp.secureserver.net	1 Hour
CNAME	webmail	webmail.secureserver.net	1 Hour
CNAME	www	ghs.google.com	1 Hour
CNAME	_domainconnect	_domainconnect.gd.domaincontrol.com	1 Hour
MX	@	smtp.secureserver.net (Priority: 0)	1 Hour

Figure 1-4. The DNS records for beapirate.com

First, the domain name, `beapirate.com`, might not actually exist. Imagine someone typing in the wrong domain name or perhaps a domain name that used to exist but doesn't now. What should happen in this case is that your computer gets back a “no records exist” message from the DNS server and shows you an error message saying as much. However, today it's pretty fashionable for your Internet provider, whether it be cable, DSL, or through your phone, to instead send you to a page of possible search results with ads (see Figure 1-5). Not very user friendly from a technical standpoint, but perhaps it helps some people find what they were looking for. It can be confusing though – because you legitimately made an error and never find out about it.

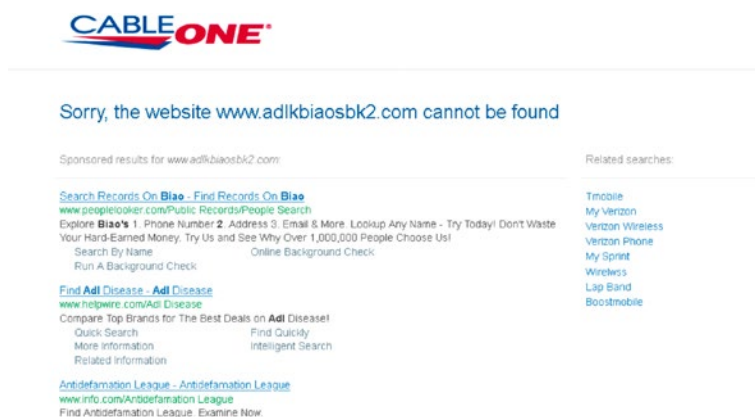


Figure 1-5. A page of search results returned when a domain name does not exist

But what if the domain name does exist but you still can't get to the page? It could be that whoever has control over `beapirate.com`'s DNS server didn't enter the right records. In essence, they screwed up when they put the address in the address book.

And finally, it's possible that somewhere someone is messing with your DNS replies. Perhaps your ISP (Internet service provider) or work network has decided that they'd rather you not visit `beapirate.com`, so instead of telling you the correct answer, they “poisoned” the results – they told you it didn't exist when it really did.

We'll talk more about DNS in future chapters, but for now, let's assume that you typed in `beapirate.com` and got the correct address. Now what happens?

Second: Let's Knock on the (Web Server) Door

Once your computer has the address of the computer that houses beapirate.com's website, it can then connect directly to it. When it does that, it makes a connection to a specific port number on that other computer. What's a port? Well, it's a bit like a door. Imagine that the other computer is an apartment building, which just happens to have 65,535 apartments. Web pages usually always live in the same apartment – when it comes to nonsecure pages, it's apartment 80. Secure web pages live in apartment 443. When your computer goes to the “apartment building” for beapirate.com, it knows to knock on door/port 80 for nonsecure web pages and 443 for secure web pages. It's also worth noting that 65,535 is not an arbitrary number; it is 2 to the power of 16, minus 1 – the highest number that can be represented by a 16-bit binary number, which the original designers of the IP decided to use when creating port numbers.

So if your computer connects to the computer that houses beapirate.com and knocks on door 80, who answers? Well, someone listening for a knock on that door. In this case, a piece of server software that's listening for requests, which then sends the web page data to your computer. Your computer then displays it to you.

There are a few things that can go wrong here too. First, the computer that you connect to might not be running the software that listens for the knock at the door – the web server. Just like trying to read your email before opening up your inbox, it's impossible for you to view a web page if no one answers the door. Second, it's possible that the web server isn't configured properly – maybe you're knocking on the “secure” door (port 443) but the server only answers the “nonsecure” door (port 80). Finally, it's possible that the server is configured properly but there is an issue with the web page.

Perhaps that web page needs to use a database to display information, and the database isn't responding or set up.

You may be wondering why I've just gone into detail on how this happens, especially so early in this book. Well, the point to be made is twofold: First, there are a lot of “moving parts” when it comes to having your own server that does even the simplest task – displaying a web page. But second, and perhaps more importantly, there are ways to deal with the times when the moving parts break. The goal of this book is to not only introduce you to how the system works but also to talk about ways in which it can fail. Ways in which you might get frustrated and ways in which you can troubleshoot on your own before having to admit defeat. My goal by the end of this book isn't to turn you into a certifiable system administrator ready for an entry-level job in IT – because you likely already have a job that you enjoy but would just like to add these skills to your repertoire. But before we get to skill learning,

we need to address some of the big questions many people have regarding purchasing, configuring, and ultimately running their own server. In this next section, we will talk about who provides services that you can purchase, how much they cost, and your responsibility for them both in time and in skill.

An Introduction to Information Technology Cloud Services

If you're like most people, you've heard the term "cloud" in referring to computers. People will say "Put that in the cloud" or "I store that in the cloud." Obviously they aren't talking about clouds in the sky, but what are they talking about?

Loosely defined, the "cloud" they speak of is a network of computers. Nothing more complex than that – leading some to say "The cloud is just someone else's computer," which is an accurate albeit slightly cynical way of putting things! Clouds exist so that you can offer services to users on the Internet without having to go the "old-fashioned" route. Let me introduce you to "the bad old days."

Physical Infrastructure

Twenty years ago, if you had wanted to run your own server, this is what you would have done. First, you would have purchased computers. Likely several of them, and as they were server-grade, they typically would cost about two to three times more than your computer at home. They'd come in special form factors, typically, to be mounted in a rack in a data center. For now, imagine that you'd just spent \$10,000 on four servers (see Figure 1-6); a router that you could plug them into for network capability; operating systems for them (if you're not using an open source operating system); various wires and cables; a keyboard, a mouse, and a small monitor (as well as something called a KVM switch that would let you switch those three items between all four servers without having to unplug and replug in everything; see Figure 1-7); and mounting hardware. It's all sitting in the back of your car or, more likely, van.



Figure I-6. Four 1U rack-mount servers. We spared no expense on labels



Figure I-7. A monitor, keyboard, and mouse connected to a KVM

Now that you have your hardware, you can't simply plug it into your home Internet connection. In theory you could, but your Internet service provider (ISP) probably won't like you running a small business from your home Internet connection. Most home connections give you a ton of bandwidth/transfer speed to download items from the Internet, but relatively small bandwidth/transfer speed to upload items to the Internet, so a home connection likely wouldn't work anyway because your web pages would load far too slow for people's tastes. So if you can't jack all of this into your cable modem at home, where do you put it? The answer is a data center, also known as a large room filled with a lot of computers!

Technically a data center is a bit more than that – typically these are fairly large rooms or series of rooms that have a raised floor to route cables and rows of metal racks that one can mount server hardware in. Data centers feature high-capacity cooling systems to keep the temperature safe for microprocessors (thousands of running computers can make things quite hot!), as well as redundant power systems and fire suppression systems. Given the computing power running at any given time, they also tend to be pretty loud! Large companies may run their own data centers for their servers, and several companies exist in the world that run data centers for clients to rent space in, typically called “co-location.” You'll find these nestled in office buildings, industrial parks, and other random places. Given the fact that companies typically store pretty sensitive data on their servers, you'll seldom see these places highly advertised or visible from the road, and security is typically tightly controlled. Finally, there are two types of co-location: manned, where technicians from the company are present all the time in case something goes wrong, and unmanned, where you might just be given a key card to get in during off hours. Manned data centers typically cost more because as the owner of a rack, you can typically request them to perform simple procedures for you – eject a USB hard drive, unplug and replug in something, and so on. If you're at an unmanned data center and it's 11 PM at night when the server goes down and it needs you to physically hold the power button to reboot, then you're driving in to do it. Something I wish I could say I've never done before!

So assuming you have your computer equipment and a signed lease with a data center, which will run you about \$1,000–2,000 a month on the low end, you now get the joy of driving up to the data center and installing your hardware. You then set it up to connect to the network and hopefully set it up so that you can access and administer it remotely (unless you like spending time in 65-degree temperatures while millions of fans whirl around you). You now have everything you need in order to run your own server. As you may have noticed though, you've racked up a ton of costs in both money and time. Around 2005, the landscape started changing – for small business owners and individuals – it became much improved.

Web Hosting vs. Virtual Private Servers

For many years, we'd had individuals described earlier, who had just put in massive amounts of money into hardware and co-location space, offer "shared web hosting" on their servers to defray costs. A web host would typically offer you an amount of space on their server, perhaps other services such as email and database, and in turn they would manage the hardware and the software associated with it. Using a shopping mall analogy, imagine a store that has booths you can rent out to sell your own wares – in this case, the store owner rents space for the entire store, and you rent space from the store owner. The mall controls the space, then the store owner, and then finally you. In shared web hosting, you have enough control to put a basic web page together, but since you were sharing the server with hundreds of other clients, you couldn't make drastic changes or install custom software (e.g., you can't control the level of the lights in the store, since all share the same ceiling!). For many people, this model is sufficient. However, as you grow, you find yourself bumping up against the limitations of the "walled garden" you're put in. Thankfully around 2005, virtualization software became much more affordable and robust, meaning that individuals who had their own physical servers could install software to create "virtual" servers on them. You could have your own storefront, instead of living in someone else's.

A virtual server differs from web hosting in that it appears to be its own computer. You get to control what software you install on it, and the owner of the physical server can allot a portion of disk space, memory usage, and CPU usage to your virtual server. A separate computer inside a bigger computer, in essence. To you, the owner of the virtual server, it appears that you have your own system – a system you can rent for a small monthly charge instead of having to invest thousands of dollars in a physical infrastructure. The downside, however, is that you are responsible for most everything with it. While a web hosting contract might stipulate that the owner of the physical server provide support in terms of fixing software issues, backing up data, or ensuring uptime, a virtual server owner assumes these responsibilities. It really is much like the analogy I used at the start of this chapter – you're renting a store in the "mall," and it's up to you to furnish it, keep it functioning, and secure it. In return, you get a very affordable rental rate.

How Much Does It Cost?

As I mentioned before, the cost difference is what makes a Virtual Private Server attractive. To see the difference, I'm going to compare pricing at four of the top web hosting companies – HostGator, DreamHost, I&I IONOS, and GoDaddy – for the following three options: a shared web hosting plan, a virtual private server, and a dedicated physical server. While many options apply, I chose to look at the most comparable and realistic options for

someone starting out. Shared web hosting is the cheapest, averaging around \$4.63 a month. A basic Virtual Private Server runs approximately \$17.49 a month, and a basic dedicated server is (somewhat intimidating) \$107.25 a month! So if you're anything more than a basic user running relatively low-traffic websites, such as those for your local business or your personal blog, a VPS is 85% cheaper than a dedicated server!

Cost is relative though, because I'll freely admit to you that there are even cheaper options out there than the "big" web hosting providers. For one, there are services like Amazon Web Services (AWS) which offers a staggering amount of cloud services that you can combine to fit your needs. A basic Virtual Private Server through AWS Elastic Compute Cloud (EC2) is free for the first year and only billed by the hour after that. This means that you could spin up your VPS for just a few minutes a day if that was all you needed. Also, many of those people who did invest in \$10,000 of hardware and co-location contracts run their own small cloud services and charge a fraction of what the big players do. On one site, lowendbox.com, you can regularly find deals on so-called "bargain basement" VPS providers. On the day I wrote this chapter in late 2020, the first news post was advertising a reasonably powerful VPS package for \$2.50 a month, or \$7.50 a quarter. Around "Black Friday" each year, it's not uncommon to find VPSs for \$10–20 per year. Locking into one of these options is quite a bit cheaper and more flexible than shared hosting. However, one must remember that there is no such thing as a free lunch...

What's the Catch?

So what is the catch? Responsibility. I run several VPSs. They host my personal web page, they store collaborative data, and they run scripts and other services that I use on a daily basis. Imagine what would happen tomorrow if you woke up and the blog that you generated your income from were to disappear. What would you do?

Likely you'd be on the phone with...someone...to fix it. If it were a web hosting company, you'd call their tech support and ask them to fix the problem. If you hired a technician, you'd be calling them asking them to drop everything and get your blog back up. If you worked for a company with a robust IT department, you might be on the phone with them asking them to check why the server was down. However, if you run your own VPS, all of those roads point back to the person in the mirror: you.

Let me give you an example: The company that I rent several VPSs through, which will remain nameless, has been relatively awesome over the past 2–3 years. My websites are available nearly 99.999% of the time (an uptime referred to as the "five 9s" – a term applied to Service Level Agreements [SLAs], a contract that some providers offer their clients guaranteeing availability), and anytime I have an issue related to availability or processing power, I get an

answer within 3–4 hours of submitting a help ticket. However, they weren't always this good – when they first started out, there were situations where a mismanaged physical server didn't provide reliable service for some clients or didn't enforce the rules on all clients such that some VPSs on the physical server were able to hog resources. It was growing pains, and thankfully they grew out of them. Other companies I'd worked with weren't so reliable. At least three to four times in the past 10 years, I've had to move my VPSs from one provider to another, and each time it was a process very much like packing up and moving your house – a hassle.

And even with a reliable company, there is only so much support they can give me. VPS providers tend to give you as much control and privacy as possible. If you don't break their rules on what content you can host, they don't look into your virtual private server. Thus, if I install a piece of software that breaks my VPS or my server contracts a virus, the best my VPS provider can do is “reimage” or “refresh” the VPS – basically restore it to the state it was the day I bought it. All of my data and customizations are gone and I have to rebuild. Some providers offer backup and restore capability, but not all.

What this means is that you need to be comfortable running your own server. From setup to deletion, backup and restore, security to usability, it's your responsibility. It's not particularly challenging if given the tools, but it is a risk that one must take – to trust in their own skills and resources. We'll talk about those skills and resources in the next section, and we'll also talk about who you can turn to for help in some situations. But I'll be honest with you now – if you work a job that requires 60–80 hours a week and you already have to outsource many components that individuals do themselves (e.g., you have a lawn service and a cleaning service, order delivery frequently, etc.), then you might not have the time for running your own server.

Skills and Resources

One might wonder what resources you need to be successful in running your own VPS. In this section, I'm going to go over what I assume about you already, as the reader of this book, what I plan on covering in this book, and what you'll need to continue doing for “professional development” over the next several years that you run your own VPS!

Assumptions

I've purposefully tried to assume very little in writing this book. I believe the goal of running your own VPS is attainable for many people, but there are some aspects that I cannot give you if you're already lacking them. Here's what I expect you are aware of:

1. Basic Operation of a Computer – How to turn one on, how to start a program, and how to navigate your computer’s operating system and a web browser. An example command I might write would be “Open your Internet browser and navigate to www.debian.org.”
2. Basic Understanding of How Computer Networks Work – The idea that computers talk to each other over the Internet and follow certain rules and guidelines while doing such. An example statement I might make would be “The server isn’t responding because software isn’t running on it.”
3. Realization That Computers Are Usually Fixable, but That This Takes Work – Often I’ll hear someone tell me “It just broke” or “They make it stop working when the new version comes out.” Both of these are inaccurate. However, if you treat a computer as a magical black box that you cannot understand the inner workings of, you’ll be very confused when I say “Copy the error message into Google and see if other people have had the same issue” while troubleshooting.

Skills Taught Here

Now that we have the assumptions out of the way, let’s talk about what I will cover here, chapter by chapter.

Chapter	Title	Skills Taught
2	Selecting a VPS Provider	We’ll go in depth on how to select a VPS provider, the different options out there, and what some of the basic terms used mean. We’ll also talk about resources needed based upon what you’d like to do with your server. By the end of the chapter, you should have picked out and signed up with a VPS provider and have your own machine to “play” with!
3	Basic Linux Administration via the Command Line	We’ll be using the Linux operating system in this book, and for many, this means the first time (or first time in a long time) using a command-line-based environment. Those old enough to remember DOS will likely feel at home, while others will be introduced to basic commands to work with files (e.g., <code>mv</code> , <code>cp</code> , <code>rm</code>), commands to start and stop services, and commands used to install new software.

(continued)

Chapter	Title	Skills Taught
4	Basic Linux Administration via GUI (Webmin)	For some, the command line is just too frustrating for everyday use. However, there is a product out there that can help – Webmin, an open source product that has been used on Linux servers for over 20 years. It allows you to configure your server through a web browser. We'll discuss pros and cons of using it and how to set it up.
5	Basics of Linux Security	Firewalls, ban lists, password hashes, and more can be daunting for first-time administrators to learn. In this chapter, we'll talk about the basics of security, from file and directory permissions to auditing what your users are doing and to making sure that only the services you want to be available actually are available to the public.
6	Basics of Backups and Version Control	Everyone needs a disaster recovery plan (DRP), and in this chapter, we'll talk about how to back up and restore your files and how to keep versions of important files so that you can always “roll back” to a previous known good version. Nothing is more frustrating than making a change to your system that you feel is small and insignificant and then finding it trashes everything!
7	Basics of Server Software Administration	I've already talked about three types of servers – DNS, web, and database. However, there are several others that I'll give a basic overview of in this chapter. You'll learn what they are, how to get started with them, and where to find more resources for each as you get yourself up and running!
8	Installing an Open Source Software Product: WordPress	WordPress is one of the most popular content management systems (CMSs) in the world and is used by millions to create web pages. While it's extremely powerful, it's also complex and can be a bit difficult to get up and running in a safe and secure manner. We'll not only cover getting started with it, we'll also talk about how to keep it running properly and out of the grasp of bad actors.
9	Administrator's Weekly Checklist	Once you have your server up and running, you'll want to take care of it regularly. I'll walk you through the tasks that you can do weekly to make sure your server stays happy and secure.
10	Open Source Application Possibilities	WordPress is one popular open source application, but there are hundreds more that are free for you to use as a Linux server administrator. I'll discuss several of these in this chapter to give you a few ideas about how you can use your server to enhance multiple areas of your work and personal life!

Professional Development

The work of a server administrator is never done, even if you're doing your weekly checklists. You also need to keep up to date on things as they change and evolve. As someone who has been working with Linux for over 20 years, there isn't a day that goes by that I don't learn of some new change or tweak that affects some part of the operating system or the operation of a server. We'll talk throughout the book about how you can stay ahead of these changes, so that you're not overwhelmed when new versions come out or new best practices are adopted.

Conclusion

Well, if you've made it this far, then I'll assume you're ready to learn about running your own virtual private server. As I mentioned, the next chapter covers finding one of them and how to get up and running with it. It's an exciting journey that we'll take one step at a time!