

May 2017

AWS VS DEDICATED SERVERS

A GUIDE TO CHOOSING THE
BEST HOSTING OPTION



SERVERPRONTO

VOLUME

1



TABLE OF CONTENTS

INTRODUCTION	02
CHAPTER OVERVIEW.	04
ASSESSING YOUR HOSTING NEEDS.	05
A CHECKLIST FOR COMPARING HOSTING OPTIONS.	10
AN IN-DEPTH LOOK AT AWS.	14
AN IN-DEPTH LOOK AT DEDICATED SERVERS.	22
VPS SHARED HOSTING AND MORE	30
MANAGED VS UNMANAGED HOSTING. . .	38
COMPARING TOP HOSTS.	45
CONTINUALLY EVALUATING NEEDS,	48
CONCLUSION.	52

Introduction

This is the most extensive and comprehensive look at hosting options including AWS, Dedicated Servers, VPS, and more. You won't find such an in-depth guide anywhere else.

Why We Wrote This Guide

Amazon Web Services exploded on to the scene in 2006. By 2007, it had gained over 160,000 users and by now it's grown to include more than 70 services and gain a market share 8x larger than Google, Microsoft and IBM combined.

At no point was Amazon's domination of the hosting industry clearer than when their largest data center went down earlier this year. A simple human error affected several of the largest websites and caused many to question the reliability of the cloud.

AWS is a massive player in the industry, but it's not the only solution, and not always the best solution. Many are beginning to realize that dedicated servers, one of the oldest approaches to hosting, can be cheaper, have higher performance, and be just as reliable as AWS. Still, the instant scalability of AWS can be a major consideration when choosing a hosting option. We wrote this eBook to provide a framework for comparing AWS to dedicated servers (and other hosting options) so that you can choose your best hosting option.

Who This Guide is For

We wrote this guide for a wide audience including bloggers, entrepreneurs, IT professionals, developers, web masters, and more. Simply put, if you have reason to use hosting, you have reason to read this guide. We want you to walk away from this guide confident that you'll be able to choose the best hosting option both now, and in the future.

How Much of this Guide Should You Read?

This guide is designed to be read cover to cover. Each new chapter builds upon the previous one and addresses an important consideration when choosing your hosting. A core idea that we wanted to reinforce is that each hosting option should be evaluated holistically. Reading this guide from start to finish will help you understand your needs, predict future needs, and choose the hardware, software and services that will best suit your needs now, and later.

Chapter Overview

Chapter 1: Assessing Your Hosting Needs

One of the biggest mistakes individuals and businesses make when it comes to choosing a hosting option is not properly understanding their own needs. Many rely on the sales guy to inform them of what type of power they'll need from a server, but there can be a conflict of interest there. This chapter will provide you with the tools and knowledge needed to assess your hosting needs.

Chapter 2: Building a Checklist for Comparing Hosting Options

Once you understand your hosting needs, you need to know how to identify which hosting packages will best meet those needs. In this chapter, we'll provide a template to follow when comparing hosts and hosting options. The template will help you cut through the marketing and sales mumbo jumbo and make a thorough comparison.

Chapter 3: An In-Depth Look at AWS

In this chapter, we'll dive deep into AWS looking at the costs, performance, features and more of this hosting option. We'll also discuss reliability, which companies should consider AWS and more. After reading this chapter, you should have a complete understanding of what AWS is, how much it costs, and whether it's your ideal hosting setup.

Chapter 4: An In-Depth Look at Dedicated Servers

Next, we'll look at dedicated servers, the tried and true method of hosting. Dedicated servers have been around for decades, powering many websites, apps, services and more. This chapter will provide you with a detailed look at dedicated servers, how they function, what their pros and cons are, and whether they are your ideal hosting setup.

Chapter 5: VPS, Shared Hosting, and Other Hosting Options

In this chapter, we'll provide an overview of some of the other common hosting options you may be considering. These include VPS, Shared Hosting and more. These hosting options each have their nuances and after reading this chapter, you'll know both what those nuances are and whether or not these hosting options are right for you.

Chapter 6: Managed vs Unmanaged Hosting

One important aspect of hosting, that can often be overlooked is whether you require managed hosting or unmanaged hosting. If you don't know what either of these are, that's okay. After reading this chapter you'll understand what each of them entail and which one you should choose when picking your hosting package.

Chapter 7: Comparing and Identifying Top Server Hosts

After you've read chapters 1 – 6, you may know what your ideal hosting setup is, but not who your ideal host is. Each server hosting company is different and focuses on a different aspect of hosting. Some may boast excellent customer service, others the absolute lowest price. This chapter will help you to find, and compare top hosts.

Chapter 8: How to Continually Evaluate Your Hosting Needs

Once you've found the perfect host and hosting package, you're still not done. You need to be able to continually evaluate your hosting needs. Searching for signs that you're outgrowing a package, or even host can be critical to preventing downtime and outages. This chapter will guide you through continually analyzing your hosting needs.

Chapter 9: Conclusion

In this chapter, we'll break everything down into a few actionable points, and provide you with a special offer.

1

ASSESSING YOUR HOSTING NEEDS

“What are your hosting needs?” This is the very first question you need to answer when choosing a server host/hosting package. There are 3 major consequences to not first assessing your hosting needs:

First off, you’re leaving it to be answered by a salesperson. Leaving the answer up to a salesperson presents an issue because there’s a conflict of interest. The salesperson is incentivized to sell you a more expensive package than you may need, and as such they may not accurately represent your needs.

Second you may end up “shooting yourself in the foot” by looking for a server without understanding your needs as you’ll only judge based off price-resulting in an underperforming server.

The third and last situation is one in which you don’t estimate your needs so you choose AWS or a similar cloud host so that you don’t need to worry. After all, you’ll only pay for what you need and if you use more it won’t matter. However, this is a surefire way to spend more money than you need as cloud hosting options can be as much as 3x more expensive than similar dedicated servers. (Though we’ll cover that more in later chapters.)

You don’t want to be in either of those situations, so the very first thing you need to do when choosing a host is assess your hosting needs. If you’re already using a server, checking what your needs are is as simple as asking the following question: “is your server meeting your needs?”

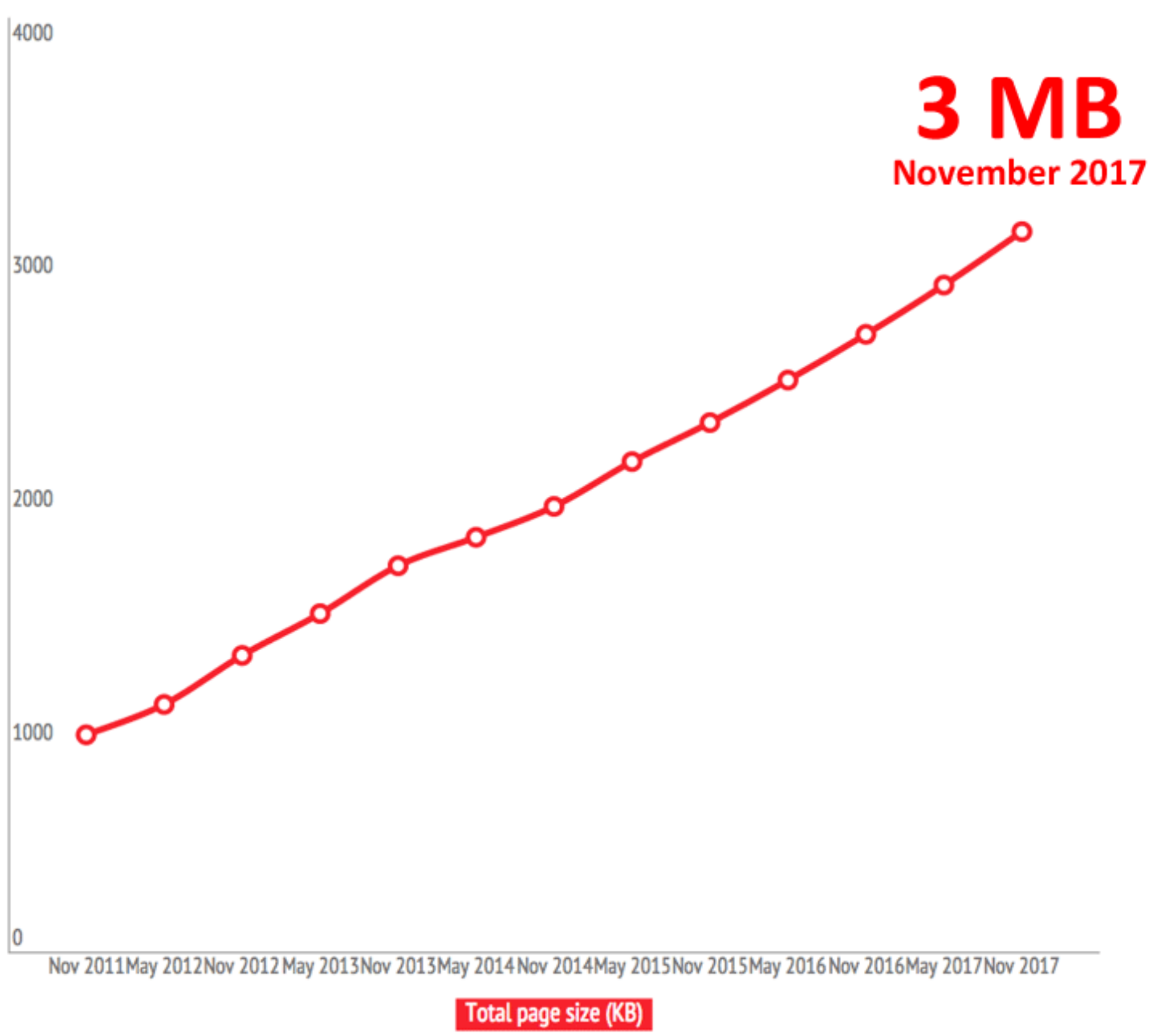
If the answer is yes, you should already have a good idea about what your needs are. You may want to look for something slightly stronger (about a 20% increase in resources) to accommodate any potential load spikes or growth, but you’ll at least have a good starting point.

If you don't already own or lease a server, or you want to be extra thorough, calculating your needs can be a bit trickier. Luckily, we've got a easy to follow method for checking just how many resources you need for your server. Let's start with storage.

Storage

Storage space isn't often given the attention it needs, with many hosts offering 1TB servers as standard. But do you really need all that space or is it just a marketing gimmick? Well, let's look at the following points:

- When a disk becomes full (or close to full) processes and applications will slow down significantly.
- Having more storage space allows you to scale up when needed.
- A server with a large amount of space can fill multiple roles.
- Everything is getting bigger, photos, videos and even web pages have more than doubled in average size. The average web page is estimated to be 3MB in size before the end of this year. (2017)



With that in mind, let's look at some common items and how much space they take up on your server:

- HTML and stylesheets account for about 120KB combined
- Scripts have doubled in the last 4 years and are now 330KB on average
- The average image is 1.3MB
- The average document is 1.5MB
- The average web page is 2.1MB
- The average song is 3.5 MB
- Video is 15MB per minute. (1 hour = 1 GB)

With these numbers in mind, you can begin to estimate how much storage space you need. Simply multiply the values above by however many you anticipate having in the next 2 years (to ensure your server will perform both now and in the future) then add 20% as a buffer zone. With this mentality, let's say you're hosting a few small websites. Each site has only 50 pages, and you've got 10 websites. That's 500 pages total, and at the average of 2.1MB that's just 1.05 GB. Add 20% as a buffer and you've got 1.26 GB of suggested storage space.

That's obviously quite small storage needs and most will need way more than that, but it's a great example of how to calculate your storage needs. Now, it's important to also consider your hard drive's speed as well. 1TB of storage isn't worth much if you only get 1 MB/s of throughput. A good SSD or similar will be more expensive, but have much better throughput. For many, this is a commonly overlooked point and can cause significant headaches as you upgrade memory or processors to no avail. Speaking of processors...

Processors

Server processors can be tricky to compare and really understand. Do you need Intel's latest generation of processors or is an older generation fine? Does .5Ghz really make that big of a deal? Do you need a Quad-Dodeca-Core processor or is that overkill? The questions go on and on.

Luckily, it's relatively easy to break down performance and identify what kind of processor you need. There are a couple of main considerations when choosing a processor:

- Clock speeds (Ghz)
- Cores/threads
- I/O bandwidth
- RAM compatibility
- Whether its server grade or a consumer/desktop model

Most server hosts will only advertise clock speeds and core count/thread count, but if you can find the name/model of the processor they use, you should be able to easily identify all its other characteristics as well.

The main difficulty with knowing what kind of processor you'll need is that it varies so much by what your usage for the server is. For example, a server responsible for video streaming will have much greater CPU requirements than a storage server!

Even though usage varies, we can give a good approximation for your CPU needs by counting the number of active users (the number of users active within a 5-minute window).

- Very Light | 20 or less active users
- Light | 21-100 active users
- Medium | 101-500 active users
- Heavy | 501-2000 active users
- Very Heavy | 5000 active users

Now, active users in this context refers to the peak number of users active within a 5-minute window. You don't want to make this number an average, as if it's an average, you'll experience server issues during usage spikes. Instead look back at your last few months and find you busiest 5 minutes. Then take that peak user number and use your current growth predictions to estimate what your active users will be in the future.

With these levels in mind. We can begin to layout your recommended requirements.

- Very Light | Dual-Core CPU with 1.9 GHz (2c/4t x 1.9 GHz)
- Light | Dual-Core CPU with 3.2 GHz (2c/4t x 3.2 GHz)
- Medium | Quad-Core CPU with 3.3 GHz (4c/8t x 3.3GHz)
- Heavy | Dual Quad-Core CPU with 2.4 GHz (8c/16t x 2.4GHz)
- Very Heavy | Dual Octa-Core CPU at 2.9 GHz (16c/32t x 2.9GHz)

RAM/Memory

Since memory is so closely tied to CPU in terms of needs, you can utilize the same system we laid out above. With this system, you're looking at the following recommended memory:

- Very Light | 2GB RAM
- Light | 3GB RAM
- Medium | 8-16GB RAM-DDR3
- Heavy | 32GB RAM-DDR3
- Very Heavy | 64GB RAM-DDR3

Now remember, these are just estimations based on usage, and if you're running resource intensive scripts or programs, your needs will be higher. It's also important to note that there are more than just 5 levels of servers, and you may find yourself needing much more than 64GB of RAM.

Traffic

Another important consideration when choosing a server hosting package is the amount of traffic provided in said package. It's hard to talk about traffic without getting into the AWS vs dedicated servers debate. With AWS and other cloud hosts, traffic is at a premium, resulting in monthly rates that can be more than 3x as much as a dedicated server.

With dedicated servers, however, substantial traffic is often included with no need for paying extra to increase that number. Even value dedicated server packages may come with some 5TB of data.

2

BUILDING A CHECKLIST FOR COMPARING HOSTING OPTIONS

Great, so you have a better understanding of your needs, but now you need to know how to compare different hosting packages. This easy to follow checklist will help you compare multiple hosting packages quickly and easily

PRICING

- Are there setup fees?
- What are the payment terms?
- How much does it cost?

CPU

- How many cores and threads does the CPU have?
- What model and generation is it?
- How many GHz?

RAM

- How many GBs of RAM?
- Is the RAM DDR3 or DDR4?

HARD DISK

- How much storage space?
- What are the read/write speeds?
- Is it an SSD?

TRAFFIC

- How much traffic is included?
- What port does the server run on?
- If no traffic is included, or additional is needed, how much does it cost?

MISC.

- What OS and control panel are included?
- How many IP addresses are included?
- Are any additional services provided?
- Is there an uptime SLA?
- What support options are offered?
- What are the hours in which you can contact support?

3

**AN IN-DEPTH LOOK AT
AWS**

Since 2006, AWS (standing for Amazon Web Services) has steadily taken over the hosting industry. Their market share is so great, that when one of their data centers goes down (like it did earlier this year), it causes an “internet snow day”. So what’s led them to become the most used server host, and are they the right host for you? We’ll answer that and more, but first we need to take a look at the history of AWS.

Before we jump into this in-depth look, it’s important to mention that AWS is just one of many different cloud hosts. They are the most popular cloud host by far, which is why we singled them out over discussing cloud hosting in general. Many of the points here will broadly apply to other cloud hosts as well (like performance, reliability, and pricing), but a few points are amazon specific (like AWS’ proprietary knowledge requirements).

The History of AWS

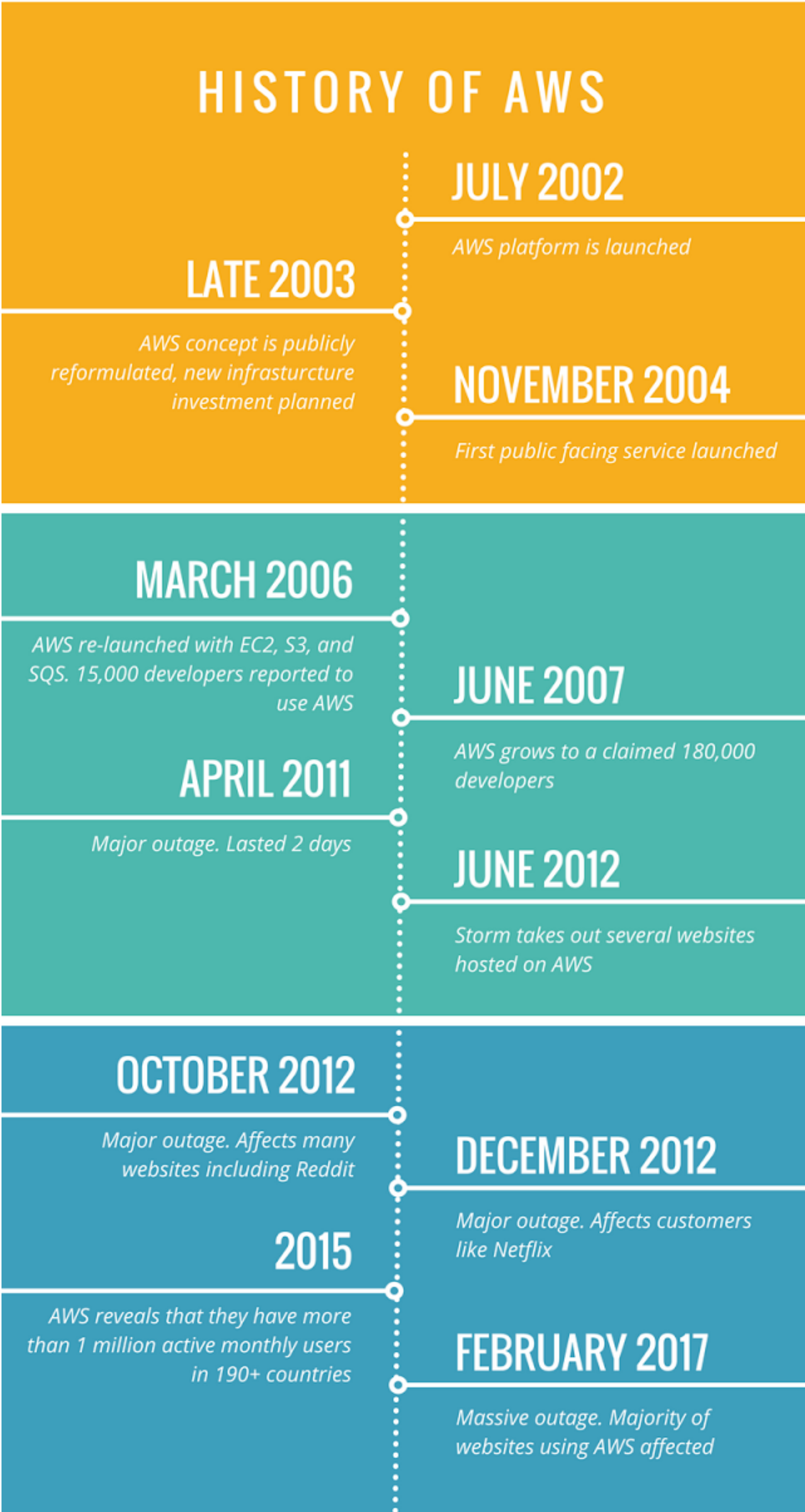
This handy infographic will help you understand the history/timeline of AWS. Take a look on the next page to see it.

THE ILLUSTRATED HISTORY OF AWS

Every major milestone and outage during
AWS' 15 year history.

.....

Presented by: ServerPronto



AWS Pricing and Performance

Finding pricing and performance information on AWS isn't always straightforward, however there are resources and benchmarks out there. Amazon itself only provides information on the number of virtual CPUs, memory and storage. They provide pricing per hour of usage, but that's not ideal when it comes to estimating your monthly bill.

Luckily, there has been some third-party testing, and we can help you calculate an average monthly bill. The following benchmarks come from an m1.large instance. These instances come with 7.5 GiB of memory, 4 EC2 compute units (2 vCPU each) 850 GB of local storage, and a 64-bit platform. Benchmarks were performed using sysbench.

First, we'll show you the raw results, then break them down.

We'll start with CPU:

```
[ec2-user@ip-10-0-0-17 ~]$ sysbench --test=cpu --cpu-max-prime=20000 run
sysbench 0.4.12:  multi-threaded system evaluation benchmark

Running the test with following options:
Number of threads: 1

Doing CPU performance benchmark

Threads started!
Done.

Maximum prime number checked in CPU test: 20000

Test execution summary:
total time:                36.1470s
total number of events:    10000
total time taken by event execution: 36.1343
per-request statistics:
    min:                    3.57ms
    avg:                    3.61ms
    max:                    4.72ms
    approx. 95 percentile: 3.71ms

Threads fairness:
events (avg/stddev):       10000.0000/0.00
execution time (avg/stddev): 36.1343/0.00
```

Next, let's check I/O

```
[ec2-user@ip-10-0-0-17 ~]$ sysbench --test=fileio --file-total-size=1G --file-test-mode=rndrw --init-rng=on --max-time=300 --max-requests=0 run
sysbench 0.4.12: multi-threaded system evaluation benchmark
```

```
Running the test with following options:
Number of threads: 1
Initializing random number generator from timer.
```

```
Extra file open flags: 0
128 files, 8Mb each
1Gb total file size
Block size 16Kb
Number of random requests for random IO: 0
Read/Write ratio for combined random IO test: 1.50
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing random r/w test
```

```
Threads started!
```

```
Time limit exceeded, exiting...
```

```
Done.
```

```
Operations performed: 259620 Read, 173080 Write, 553828 Other = 986528 Total
Read 3.9615Gb Written 2.641Gb Total transferred 6.6025Gb (22.536Mb/sec)
1442.33 Requests/sec executed
```

```
Test execution summary:
total time: 300.0010s
total number of events: 432700
total time taken by event execution: 5.9789
per-request statistics:
    min: 0.01ms
    avg: 0.01ms
    max: 0.16ms
    approx. 95 percentile: 0.02ms
```

```
Threads fairness:
events (avg/stddev): 432700.0000/0.00
execution time (avg/stddev): 5.9789/0.00
```

And finally, let's look at memory:

```
[ec2-user@ip-10-0-0-17 ~]$ sysbench --test=memory --memory-block-size=1M --memory-total-size=7G run
sysbench 0.4.12: multi-threaded system evaluation benchmark

Running the test with following options:
Number of threads: 1

Doing memory operations speed test
Memory block size: 1024K

Memory transfer size: 7168M

Memory operations type: write
Memory scope type: global
Threads started!
Done.

Operations performed: 7168 ( 3649.63 ops/sec)

7168.00 MB transferred (3649.63 MB/sec)

Test execution summary:
total time: 1.9640s
total number of events: 7168
total time taken by event execution: 1.9552
per-request statistics:
    min: 0.27ms
    avg: 0.27ms
    max: 0.47ms
    approx. 95 percentile: 0.29ms

Threads fairness:
events (avg/stddev): 7168.0000/0.00
execution time (avg/stddev): 1.9552/0.00
```

Now, let's interpret the results. First off, the CPU isn't bad! It's just shy of a Quad-Core 2.53 GHz in terms of performance, and isn't anything like the vCPUs of old. I/O is also surprisingly good with 1400 IOPS and throughput of 22.5 MB/s. This is better than many SATA drives, but isn't as fast as a SSD which can be had in many dedicated servers today. Unfortunately, the results for memory were a bit disappointing and is about 30% slower than the DDR3 RAM commonly found in dedicated systems.

So, performance is good overall, but nobody doubts AWS' performance. What will put a damper on things is AWS' price. The server resources alone cost \$256.20, but that's not where the real expense is. The real expense lies in bandwidth. For 5TB of outgoing data transfer, you'd be looking at \$460.71. This would bring your total monthly pricing up to somewhere in the ballpark of \$715.65. Meanwhile, even a \$29 value dedicated server comes with 5TB of bandwidth included.

Reliability

Because each of AWS outages is so high profile, you're likely already aware of a few of them. They've had 5 major outages since 2006, which have been listed below.

October 2012. An outage occurred in which a latent memory leak bug in an operational data collection agent) affected many sites such as Reddit, Foursquare, Pinterest, and more. December 2012 outage which affected customers in the northeastern united states.

September 2015. An outage caused by a power outage and inadequate failover procedures occurs. The outage affects a number of related Amazon services include Simple Queue Service, EC2 autoscaling, Amazon CloudWatch, and the online AWS console. A number of customers are negatively affected, including Netflix.

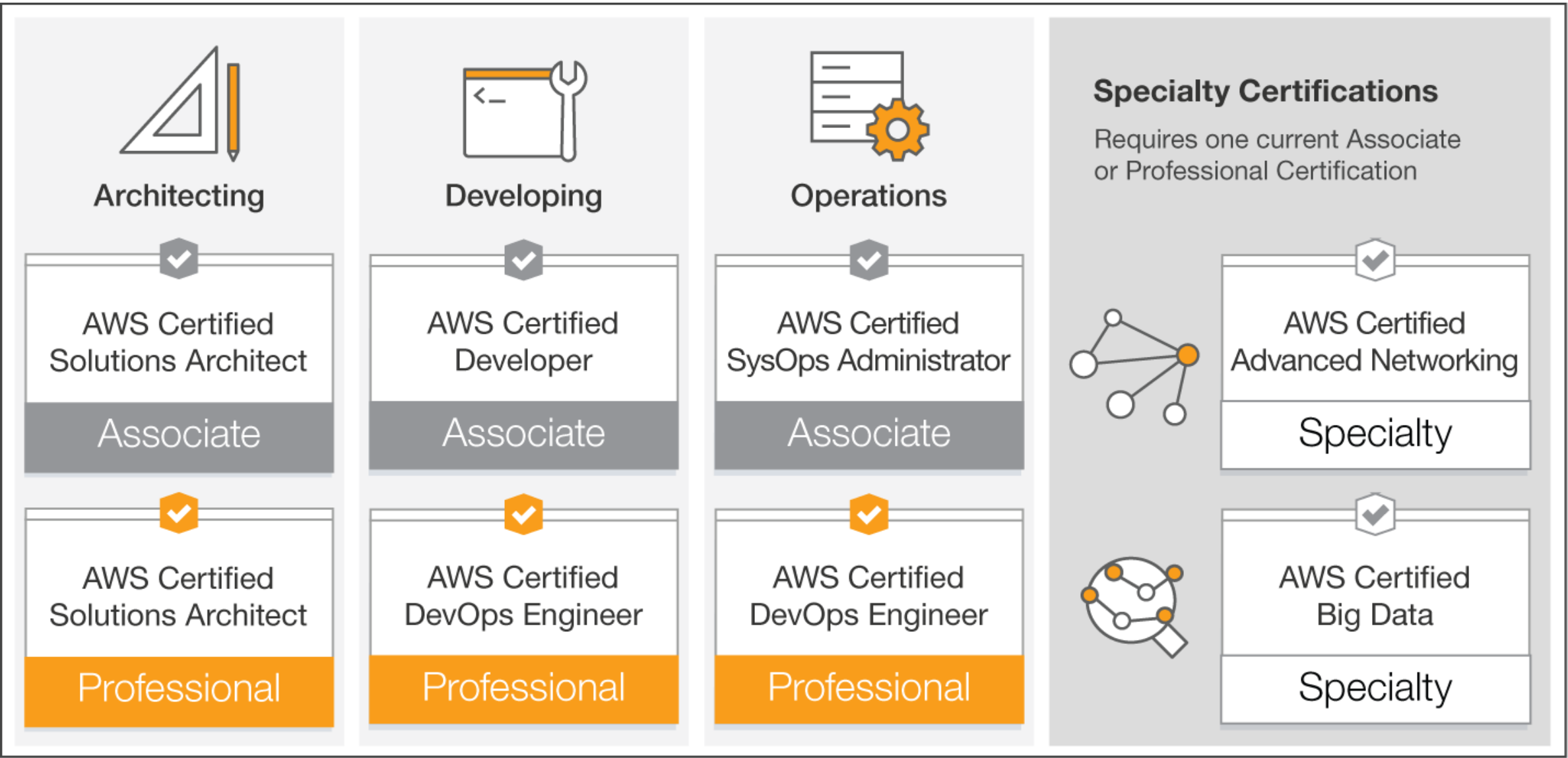
June 2016. This outage, which affected AWS Sydney, was caused by severe thunderstorms in the region which caused a power outage in the data centers.

February 2017. An outage affecting S3 in us-east-1 occurs. Other services in us-east-1 fail as well including Cloudformation, autoscaling, Elastic MapReduce, Simple Email Service, and Service Workflow Service. The AWS status dashboard also fails to report the outage due to a dependency on S3. Websites and services including Medium, Slack, Imgur, Trello, and more are affected. It was later revealed that the outage was caused by human error.

Now, this may seem like a lot of outages, especially in recent years at a rate of 1 major outage per year, but most users still experience one of the highest uptimes of any cloud hosting provider.

Let’s Talk Proprietary Knowledge

One of the often-overlooked aspects about Amazon Web Services is the fact that it requires proprietary knowledge to use. Businesses that wish to use AWS must have their developers learn architecting, developing, and operations from the ground up. In fact, the knowledge required to use AWS has led Amazon to create 6 different certifications in 3 different categories for its web hosting services.



These certifications aren’t free either! The associate certification has a \$150 registration fee while the professional exam fee is \$300. In addition to exam fees, you’ll need to pay around \$25/mo per developer/system admin to train them for their certifications and keep them up to date afterwards. There are also the AWS Global Summits, which while free, still involves the cost of travel, hotel and food. The summits are highly recommended for AWS users as they educate customers on technical changes and aid in companies being more successful with AWS.

Assuming 6 months of training are needed, a practice exam is taken, and the developer is attaining the professional certification, you could be looking at a cost of \$475 per developer just to ensure you and/or your developers have the knowledge necessary to use the platform!

No other hosting solution has ever required users to learn proprietary knowledge.

Why do you think Amazon does this? It's simple. To lock users into using AWS for the foreseeable future. If you've already invested the time and money to train your developers to use AWS you don't want to throw that away to switch to another platform.

It also incentivizes developers to push their CEOs to use AWS over other hosting options. This is because developers with certifications and experience using AWS will have more job opportunities and can be paid higher than those without AWS experience. Just how much higher are AWS certified developers paid? Here's a recent look at average salaries:

- AWS Certified Developer - Associate Level \$137,825
- AWS Certified Solutions Architect - Professional Level \$117,434
- AWS Certified Solutions Architect - Associate Level \$114,935
- AWS Certified SysOps Administrator - Associate Level \$108,046

For comparison, here are the salaries for non-AWS certified positions:

- Developer - \$75,441
- Solutions Architect - \$70,550
- SysOps Administrator - \$66,735

Who should use AWS?

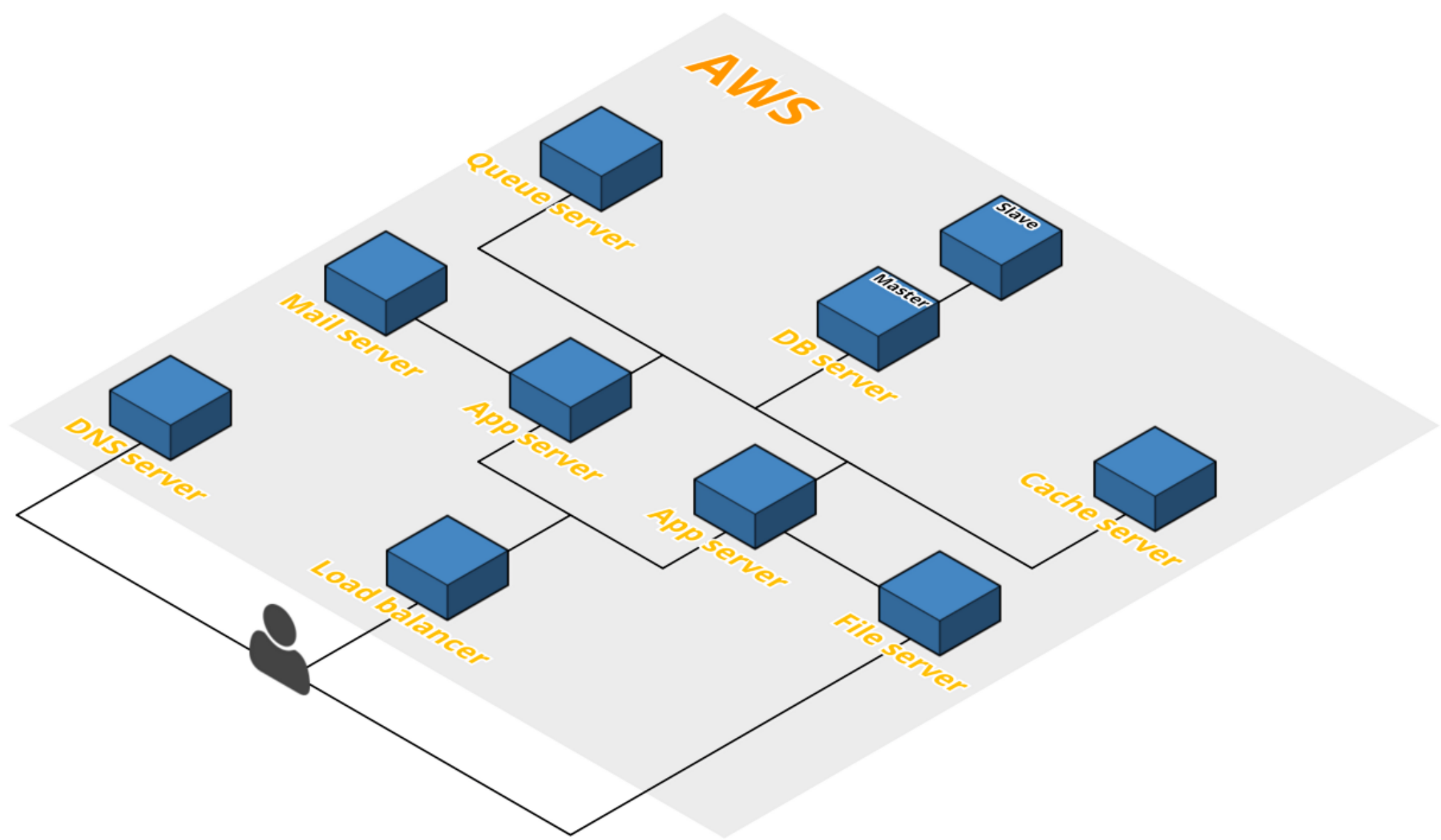
If you're looking just at the spec sheet, AWS is a solid hosting choice. However, when you factor in price, AWS starts to look less appealing. The cost of AWS is nearly 4x that of a comparable dedicated server. For most companies, that simply won't do. This means that AWS is only the best choice in two cases.

First, if your usage is going to be extremely unpredictable with large spikes followed by periods with little to no usage. In situations like this, Amazon's rapid scalability and payment system (you only pay for the resources you use) will make sense for you. However, if your usage is predictable, it's probably best to consider a dedicated server capable of handling those spikes in usage.

The next person who would likely use AWS is someone with which money is no object. Maybe you just raised millions in your round of funding, or are simply a large enough organization. In these cases, your monthly bill for hosting really doesn't matter as it won't make a big dent, however there is still something to be said for saving money on the hosting bill and using it elsewhere.

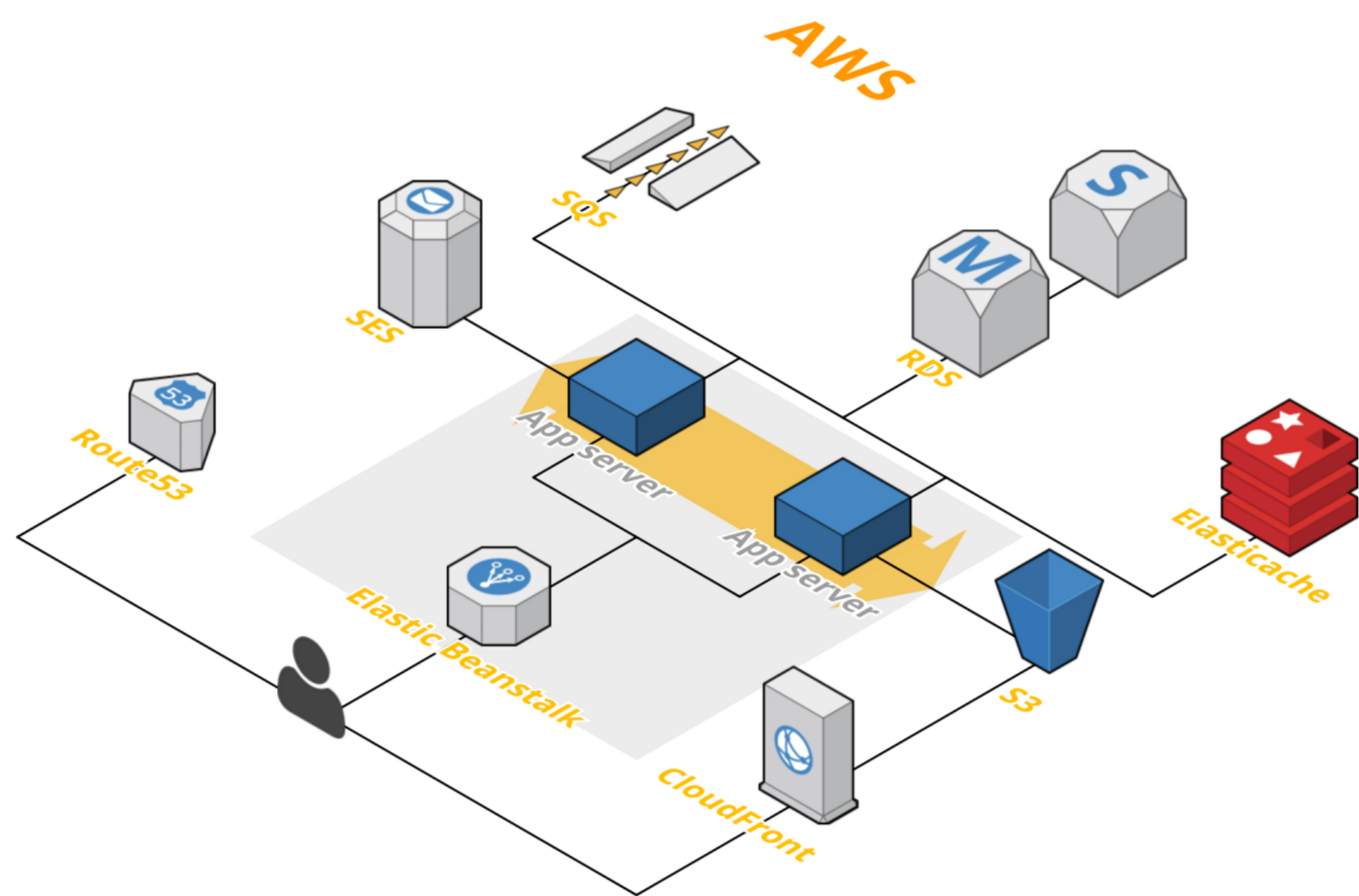
What does an AWS setup look like?

It’s easiest to answer this question with a visual. Each blue square in the image below represents an Amazon instance.



POWERED BY
CLOUDCRAFT.CO

Another, more in depth image can be seen below. This image shows some of the specific AWS services used.



POWERED BY
CLOUDCRAFT.CO

Keep in mind, you or one of your IT staff will be responsible for setting all this up. This is why the proprietary knowledge is so important. You and your team will either need to learn or pay to hire someone who knows AWS to set this up for you. Most dedicated servers are significantly easier to set up as they use common knowledge and require less advanced infrastructure.

4

**AN IN-DEPTH LOOK
AT DEDICATED
SERVERS**

Dedicated Servers are the “OG” when it comes to hosting. The most basic definition of a dedicated server is as follows:

A dedicated server is a reliable and powerful computer in which the full hardware resources are solely dedicated to your server tasks and performances. There is no shared computing power or memory usage and you have full control over the server.

This differs from AWS in which the resources of the physical server are being shared by many customers. It also provides more control than AWS, but we’ll get to that later.

Now it’s important to note that dedicated servers are a type of hosting option rather than a specific host. Unlike AWS in which we singled out a single host due to their domination of the cloud market, there are plenty of dedicated server providers with no 1 single host dominating the market. This chapter discusses dedicated servers only, but we’ll discuss how to choose your specific host in a later chapter.

Let’s get into it, starting with an detailed description of what dedicated servers are, and what they typically come with:

What is a Dedicated Server?

Dedicated servers are stand-alone physical computers typically consisting of enterprise or datacenter hardware resources. In other words, they are not your average desktop computer. They consist of more powerful and more reliable hardware than consumer computers and are built to power business-critical applications.

With dedicated servers, you don’t share hardware resources. The server is “dedicated” to one client. Because you don’t share computing power or connectivity, you’ll have the highest speeds possible when performing server tasks.

Dedicated servers typically come with at least one public IP address, full root/administrator access, an operating system of your choice, and monthly bandwidth. The actual server hardware can consist of many different combinations – and some hosts will let you build/customize your dedicated server's physical components.

Because you have full control over your server, the uses for dedicated servers are endless. They can be used as web servers, email hosting, reseller hosting and more! No matter the usage of your server, there are a few considerations you need to make:

Operating Systems

The first thing you need to look at when choosing a dedicated server is the operating system it will run. There are dozens of different OS, but it boils down to two main systems: Linux and Windows. The main choice for server operating systems is Linux due to its low price point, decades long track record of reliability, and plethora of options. Windows Server has been gaining market share as of late, but Linux remains the top choice.

Let's take a look at a few of the most popular Linux OS for dedicated servers, along with their pros and cons.

CentOS

CentOS is the most popular OS choice for dedicated servers, and with good reason. CentOS is based on the Red Hat distribution.

Pros:

- Widely distributed enterprise solution
- Based on a commercial solution
- Very Stable
- Extensively tested

Cons:

- Long period between releases
- latest features not always available

Debian

Debian is one of the oldest distros out there, and it's still going strong.

Pros:

- Stable releases with few bugs
- Support for many architectures
- Large, helpful community

Cons:

- Complex installation process
- Long wait for releases

Ubuntu

Ubuntu is normally thought of as a desktop distribution, but it is also one of the most powerful Linux server OS out there!

Pros:

- Ease of use
- Plenty of documentation
- Steady release dates
- Official support

Cons:

- Some software will not run properly
- Proprietary software is used
- Features don't support power users

CPUs

In addition to the Operating System, you need to consider your Processor. Unlike cloud hosting which utilizes vCPUs, a dedicated server gives you the choice over which CPU you use. This enables you to choose the best processor for your needs, but if you're a little lost with all the industry jargon, no worries. Here's what you need to know:

Server Grade vs Consumer Grade

Let's address the topic of server grade CPUs vs consumer grade CPUs. (This applies to other components such as RAM as well!) Simply put, server grade CPUs undergo more rigorous testing, resulting in a more dependable and reliable component. They also come with server-grade features like error code correction (ECC) and higher frequencies (which processes more data). Consumer grade CPUs are cheaper, but lack server-grade features and may suffer from decreased dependability and lower resistance to high temperatures.

So in the end, is it okay to use a consumer grade processor in a server? Sure, but you will miss important server features like self-correction. Plus, if you're looking at a server host who's using a consumer grade CPU, it's a sign that they could be cutting corners where they shouldn't.

Processor Generations

CPU manufacturers come out with a new generation roughly every 2 years. These generations usually feature new or improved architecture which allows for increases in efficiency and performance. Generally, you can assume each new generation is roughly 10% more powerful than the last. However, occasionally you have a generation that makes as much as a 50% increase in powerful or as little as 3% more powerful.

So, it's a good idea to look for the newest generations of server processors.

RAM

Choosing RAM for your server is easy. You'll want to look for at least DDR3 RAM, with DDR4 being a plus. (A big plus!) The amount of RAM you pick will depend on your hosting needs as we discussed earlier:

- Very Light | 2GB RAM
- Light | 3GB RAM
- Medium | 8-16GB RAM-DDR3

- Heavy | 32GB RAM-DDR3
- Very Heavy | 64GB RAM-DDR3

Hard Disk

Many people fail to realize how crucial the hard disk is to server speeds. A slow read/write speed can hinder the performance of a server, and many inexperienced server owners will incorrectly assume the RAM or CPU needs upgrading.

Let's remember that a server's primary purpose is to serve files. If the drive that's storing said files is slow, it doesn't matter how fast your CPU and RAM are. You won't be able to serve the files at a reasonable speed!

This is actually one of the cloud's biggest weaknesses as its hard disk speeds very rarely match that of a true dedicated server. This is especially true when using a dedicated server with solid state drive (SSD). SSDs offer increased performance and are becoming available in almost all dedicated servers. A few hosts even offer packages with SSDs standard. If you want the best performance, you'll need to choose an SSD, and you'll benefit greatly from sticking with a dedicated server instead of a cloud hosting option like AWS.

Traffic

As we mentioned earlier, traffic is one of the main categories in which dedicated servers stand apart from cloud hosting solutions such as Amazon's AWS. Most dedicated servers come with a large amount of traffic included, which'll save you from headaches down the line (not having enough traffic) and save you a great deal of money both now and in the future.

Traffic can quickly become a major issue if you don't have enough allocated or if you aren't on a fast enough port. To ensure your dedicated server is the best possible, get one that has sufficient amounts of traffic and consider looking for a server with a gigabit port.

5

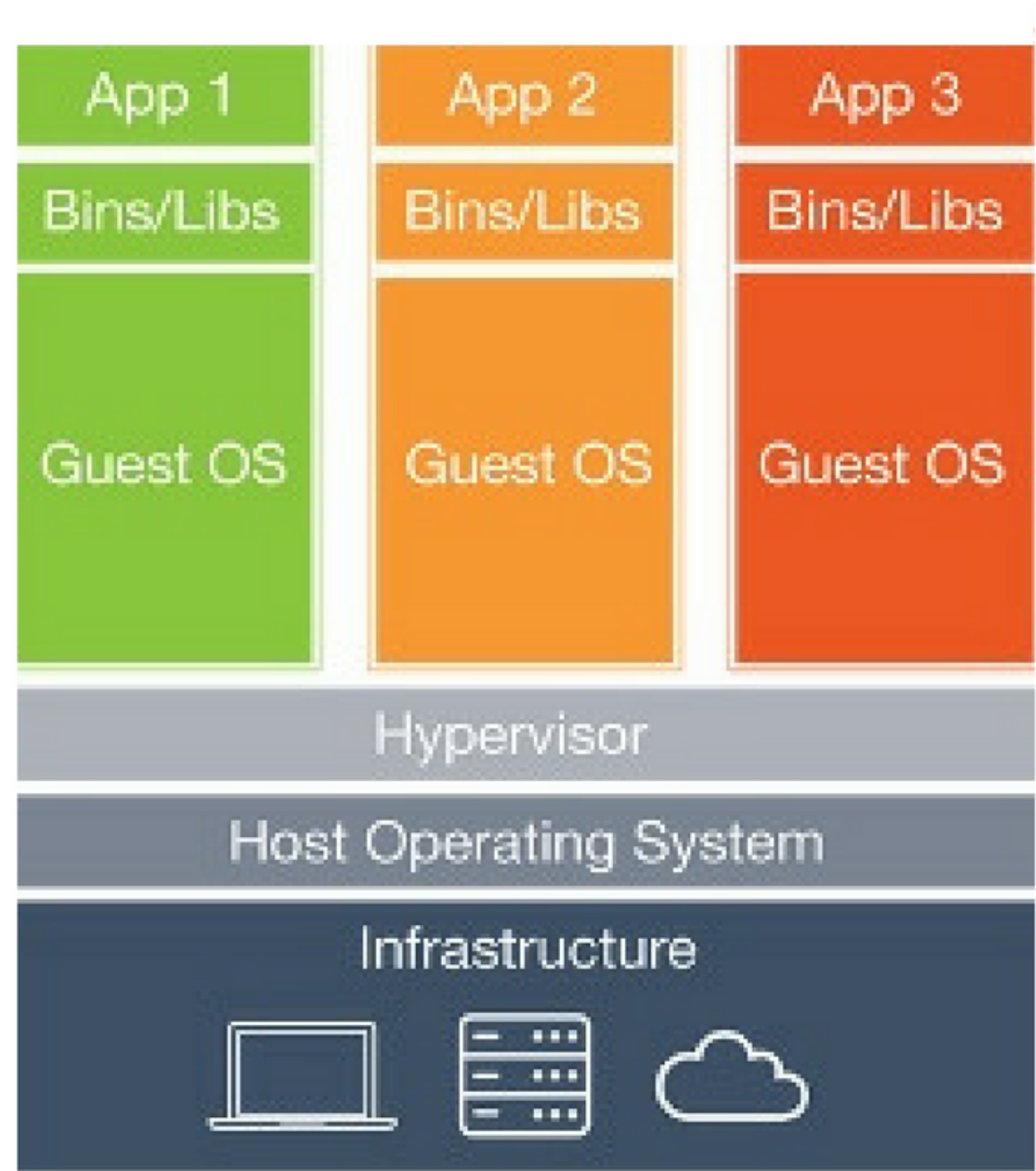
VPS, SHARED
HOSTING, AND OTHER
HOSTING OPTIONS

It's important to both remember and discuss the other options you have when choosing a hosting solution. There are a variety of different solutions on the market including:

- Public Cloud Hosting (AWS)
- Dedicated Servers
- Virtual Private Servers (VPS)
- Shared Hosting
- Private Cloud Hosting

While this guide primarily serves to compare AWS and similar public clouds to dedicated hosting, there are cases where you may not wish to use either! Let's go over some of these other hosting options and look at when you may want to consider one of these over a dedicated server or public cloud hosting setup.

Virtual Private Servers



Virtual private servers are as close to a middle ground between shared hosting and a dedicated server as possible. A VPS is still a dedicated server being shared by multiple users, but it has been setup in a way that it feels like a dedicated server. Let's take a look at how this works:

You start by taking a run of the mill dedicated server and dividing it into several equal parts. These parts each have a percentage of the server's resources and storage allocated to them. Ultimately, users still share a server's resources, so a spike in resource usage by someone else could cause your VPS to run poorly. Still, VPS provide a good deal of control and are typically cheaper than dedicated hosting solutions. This makes them a go to choice for many first time server owners and for individuals and corporations which can't quite make the investment in dedicated servers.

If you do choose to utilize VPS hosting, here are a few points to consider:

Choose a VPS that offers SSD Storage. This will result in significantly better performance. Next, ensure that the amount of included traffic is sufficient for your needs. VPS solutions may not come with as much traffic as a dedicated server. Ask the hosts you consider what the performance of their CPUs and RAM are like. When it comes to VPS, many just list the quantity without providing any real performance information. This could result in you receiving a less than stellar experience with outdated RAM and/or a slow CPU.

Shared Hosting

Shared hosting is a server hosting solution in which multiple customers utilize the same server. It is common as a web hosting server solution where multiple websites are based on one server. Shared hosting has a few big pros and cons that'll set it apart from the rest of hosting solutions and make it clear if it's the setup for you.

Let's start with the pros:

- It's significantly cheaper than other hosting solutions.
- Typically, the server host will maintain the server and provide technical support (which is great if you aren't technically inclined).

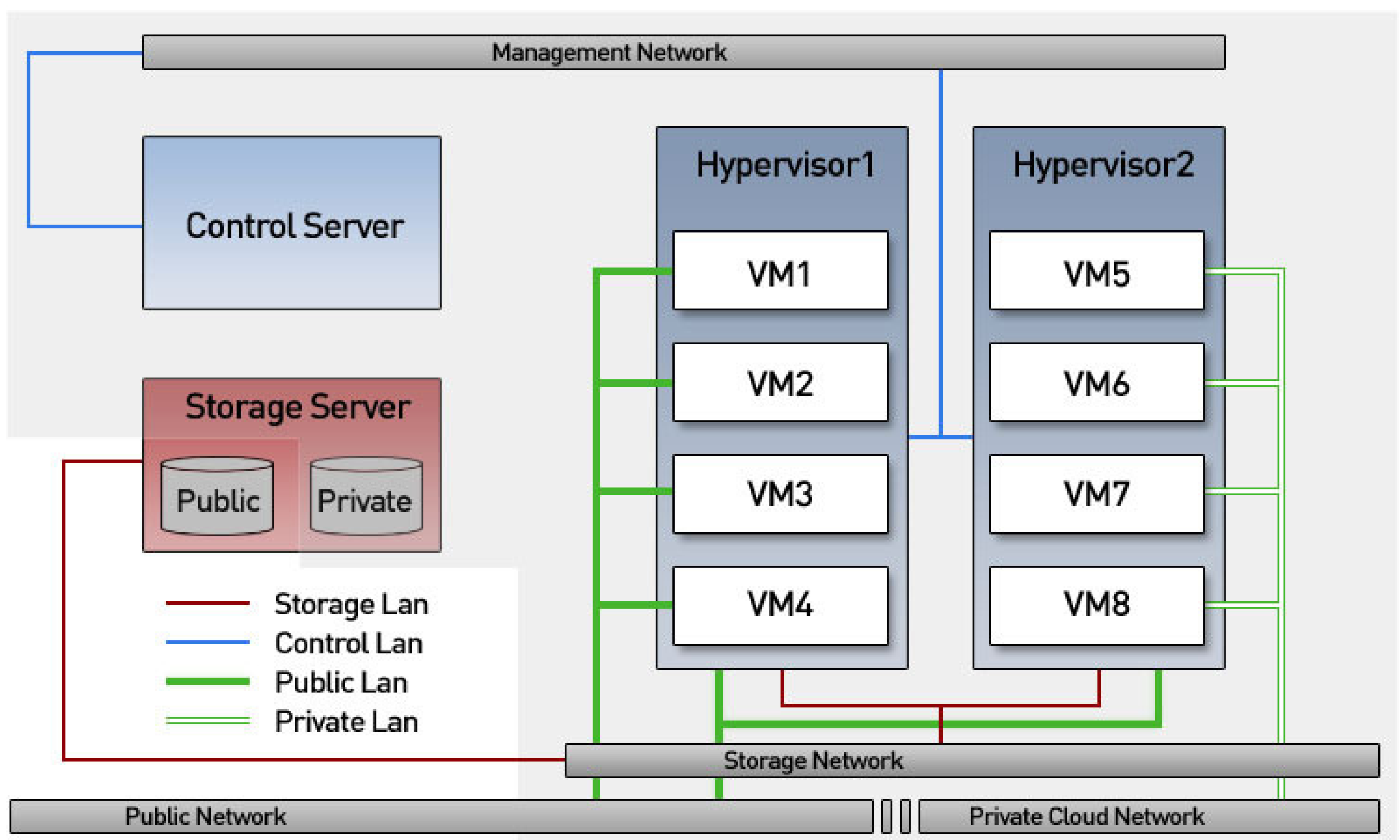
Now, let's look at some of the cons to going with a shared hosting setup:

- Performance of the server is highly variable and your website may experience slow speeds or interruptions based on other's usage.
- The lack of control makes this hosting method less than ideal for many hosting applications.
- The shared environment may present a security risk.

Ultimately, this method may be okay to use for small to medium sized web hosting applications, but for most use cases you'll be better off with a different hosting option. Even those with a small website will likely want to invest in a budget VPS server.

Private Cloud Hosting

Private cloud hosting is a server solution in which you operate your own OpenStack, Hyper-V, or OnApp Private Cloud. The cloud itself will be distributed over multiple dedicated servers (hypervisors) and includes a cloud management system. This type of setup will provide you with the best security and scalability as possible. You may also be surprised to hear that it may be as much as 3x as cheap as a public AWS instance. Here's a visualization of this setup:



Let's compare one of ServerPronto's private cloud offerings with a comparable AWS server:

ServerPronto's private cloud offers:

- 1x Control Panel Server Intel Xeon Quad-Core 8GB RAM-DDR3 2 x 500GB HDD SATA2
- 2x Hypervisor Servers 2x Xeon Quad-Core 32GB RAM-DDR2 4x 1TB HDD SATA2
- 1x Backup Server Intel Xeon Quad-Core 8GB RAM-DDR3 2x2TB HDD SATA2
- Cloud Control Panel
- 16 Public IP Addresses
- 1 Gb/s Uplink
- 40TB Traffic
- \$899/mo

In other words, you get 16 cores, 64GB memory, and 2TB storage for \$899. AWS on the other hand offers this:

- M4.4xlarge EC2 Instance 64GB Memory 16 vCPU
- 2000 GB of Amazon EBS Storage
- 40TB Traffic
- \$2762.81/mo

A nearly 3x as expensive solution for a setup that you ultimately have less control over. Additional benefits of private cloud hosting include:

- Increased security.
- Compliance with industry standards such as HIPAA, PCI, and SOC.
- Increased control and customizability.
- Hybrid deployments.
- Increased compatibility with legacy systems.
- Often include fully managed services from the hosting company.

If you do decided to use a cloud hosting solution, and you're looking for a significant amount of resources, a private/dedicated cloud will almost always be the best choice.

6

MANAGED VS UNMANAGED HOSTING

If you were thrown off by our mention of fully managed services in the chapter before, no worries. After this chapter, you'll have a complete understanding of managed hosting also referred to as managed services or even a managed server. Let's start with the basics:

What is Managed Hosting?

A single, specific definition of managed hosting is elusive, because each server host will include different services in its definition. However, we can define it as a form of server hosting in which the server's host also provides management and maintenance of the servers which it leases. Unmanaged hosting is simply when a client manages and maintains their own server.

Managed hosting may include any number of the following services:

- Server monitoring
- Proactive support
- Security
- Backup and storage management
- Configuration and maintenance
- Performing updates
- Anti-spam
- DDoS protection
- Database administration
- Load balancing
- Disaster recovery
- DNS management
- Performance tuning
- User management
- Consultation
- And in some cases, even programming.

Typically, a server host will break their managed services into different tiers. The norm for the industry is as follows:

- Fully managed, which includes monitoring, software updates, reboots, security patches, OS upgrades, and more. With this, customers can usually be completely hands-off.
- Managed, which includes a medium level of, monitoring, updates, and support. Customers may have to perform specific tasks, but most day-to-day details will be taken care of.
- Self-managed, which may include monitoring, maintenance, and support. Customers are responsible for managing and maintaining a server in this case.
- Unmanaged, which includes no services for the server provider. With this hosting option, a customer is completely on their own and must deal with any problems that may arise.

Not all server hosts will offer services that fit neatly into those four categories, but it's a good generalization to keep in mind. Now that we've got a firm understanding of what managed services are, we can begin to discuss the benefits of managed services:

Do you need managed services?

Managed services come with a number of benefits and a couple of drawbacks. One of the biggest benefits to using managed services is that you no longer need to have strong technical knowledge on your end. (If you're utilizing a fully managed server.) Managed services allow those unfamiliar with how to manage a server to use a server. This isn't to discredit the need to learn some server operating basics, but it will enable you to learn at a more leisurely pace. Better to learn at your own pace, then to need to know the information on a day that something's wrong with your server.

Another benefit of using managed services is the time savings factor. Even if you have IT staff who are knowledgeable and completely capable of running your server, you'll want to look into managed services. Why? Because with managed services, your IT staff can focus on more important work than managing your server.

Then there's the factor of knowledge. Your server host's staff should have more knowledge than you and/or your IT staff when it comes to your server. If it's a good host, their staff will be specialized and be able to detect and fix potential issues sooner than your team could. They should also be more up to date on industry news such as a recent new exploit that needs to be fixed. This all combines to provide you with a more efficient, better protected server.

How does managed services play into the AWS vs dedicated servers debate?

Simple, AWS doesn't offer any managed services. And, to take things a step further, it'll require specialized knowledge which you likely don't have. As we mentioned in chapter 3, AWS requires proprietary knowledge and even has its own certifications in order to teach you said knowledge. This creates a system where you're on your own, and either need to take the time to learn the system yourself or hire someone (which is likely a near 100K a year expense) in order to effectively manage and maintain your AWS hosting setup.

Meanwhile, dedicated servers, VPS, private cloud hosting, and pretty much every other form of hosting offers at least some level of managed services. Even when paying for a fully managed dedicated server, you're going to be paying a significantly smaller amount than you would be paying for an unmanaged AWS instance. In fact, when you factor in the cost of an AWS certified employee, or the cost of learning yourself, you'll be paying even more for AWS than any other option.

Even a private cloud will come fully configured and with 24x7 support. This factor alone puts AWS at a huge disadvantage.

7

COMPARING AND IDENTIFYING TOP SERVER HOSTS

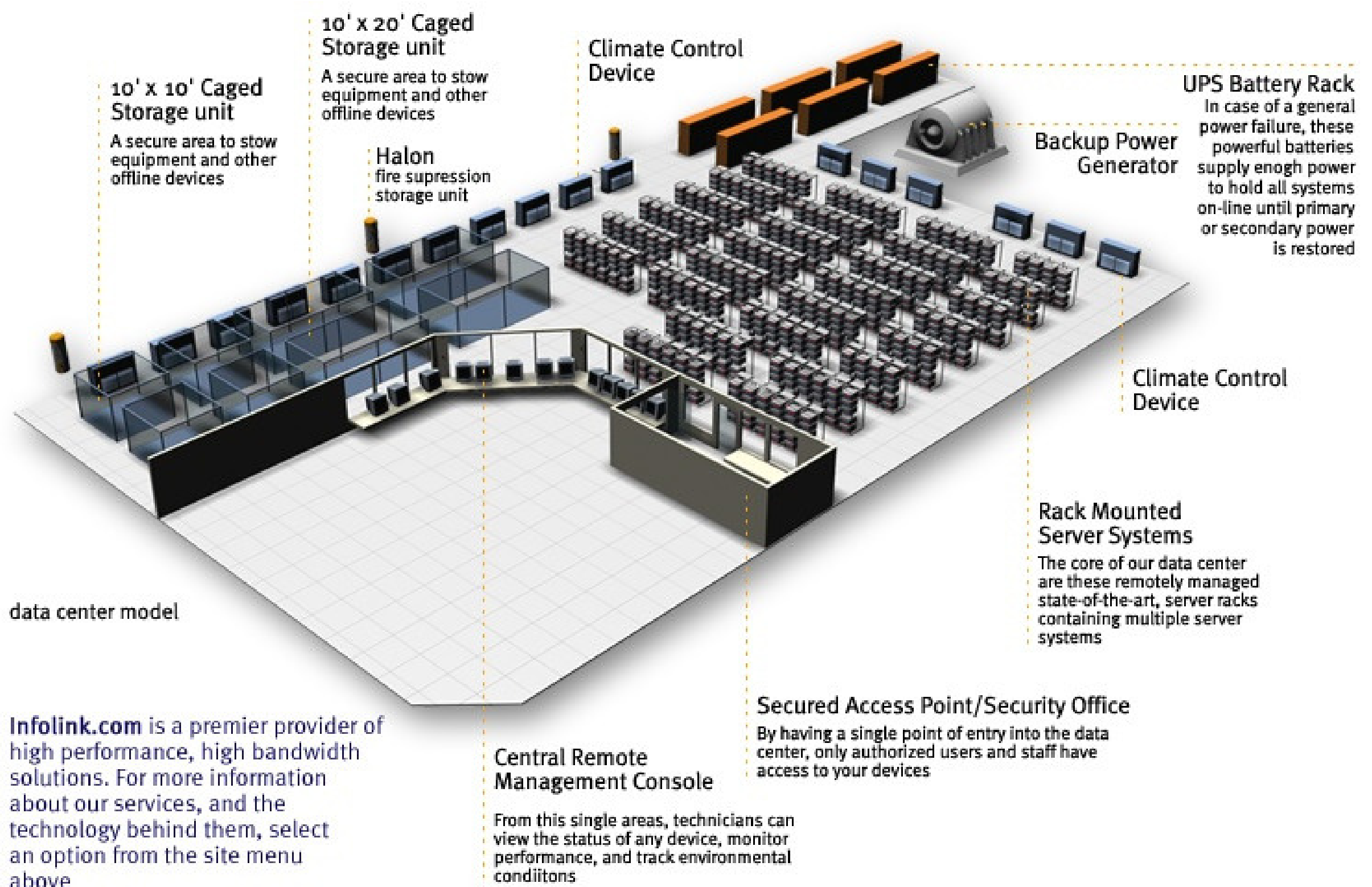
Even though almost all server hosts offer the same reliability and performance, not all offer the same support experience, pricing, or customizability/scalability. These factors can play a big part in your hosting experience. So, here's how to identify a top server host and then to compare said host with other top hosts.

Get an Overview of the Host

For starters, you'll want to get a broad overview of the host and their history. How long have they been around? Are they owned by another company? Have there been any recent changes in ownership which could result in a better (or worse) experience? The answers to these questions will be surprisingly important. For example, if a company changed ownership in the last year, you may not be able to count on the validity of previous reviews. The prior owners may not have put a focus on customer satisfaction whereas the new owners see the customer as king. If that's the case, previous negative reviews may no longer be applicable. Similarly, the reverse could happen in which case prior positive reviews could become meaningless.

Other questions you'll want to ask about the hosts is where they are based out of. (Their headquarters, not their datacenter). This is important because their time zone may be different from yours and it could result in lackluster support when you need it most.

Next you will want to know their datacenter locations. Ideally, they should have a datacenter in your region and/or the region where the bulk of your server's end users would be connecting from. This will result in the fastest connection times for end users creating a better experience overall. In addition, you may want to get more information on their data center and the steps taken to ensure security/redundancy. You can see this in ServerPronto's own Miami based datacenter:



Let's Discuss Uptime and SLAs

A lot of people think that uptime means the amount of time which a server will be online. This isn't true. It's impossible to offer uptime of 100%, unless you use a geo-redundant cloud setup, and even then, you could theoretically have the server go offline.

Uptime when advertised by server hosts is often a SLA (service level agreement) which states that if their server experiences unplanned downtime, they're bill will be discounted. The nitty gritty is a lot more technical, and many hosts have clauses in their agreement which makes obtaining a discount difficult though. Take AWS as an example. In their agreement, the SLA is promised at 99.95% uptime for the month.

The issue with their SLA however is that it's only considered an outage if 100% of your instances can't reach the Internet 99.9% of the time. In other words, 99% of your instances can have no Internet access and it's not an SLA violation. 100% of your instances can fail, be rebooted, or hang, and it's not an SLA violation. EBS, RDS, Dynamo, SQS etc. can all fail entirely and it's not an SLA violation.

What does this all amount to? Unless you dive into the SLA agreement and see that the SLA isn't just a marketing tactic (i.e. no tricks like in AWS' agreement) then the uptime guarantee by most hosts isn't worth anything. A better idea is to search for a third party website which reports on uptimes and see what their true uptime is.

You could also look for reviews to see if there are many complaints about downtime.

How's their support?

Support is one of the biggest make or break aspects to a good hosting experience. There are a few key points to consider when looking at how good a server host's support will be:

- Technical knowledge
- Certifications
- Availability
- Ways to contact support
- Other support resources
- Other support services

Let's dive into each of these points and explore how they'll impact your hosting experience starting with technical knowledge.

While it's becoming less common, there are still hosts out there that don't have support teams with extensive knowledge. They may be only skilled in simple technical support tasks rather than being true IT experts. An expert technician should be able to quickly and easily:

- Assist you with server setup
- Reboot your server
- Perform system restores
- Provide upgrade support
- Troubleshoot DNS issues
- Assist with SQL database hosting
- Set up FTP connectivity
- Enable Website hosting
- Configure POP3/IMAP/SMTP e-mail services
- And more.

Another thing to check is their availability and the ways in which you can contact support. The ideal host will have support that's available 24/7/365. Additionally, you should be able to reach them by several different channels including live phone support, live chat, email, and a ticket system.

It's also important to consider the other support services and resources they offer. Does the host have a blog where you can learn more about how to manage and maintain your server? Do they offer free monitoring of your server? Free setup assistance? These details will all accumulate to create a significantly more pleasant hosting experience.

Lastly, you'll want to know what certifications their engineers hold. Many hosts will require all technicians to either be or become certified. Look for Red Hat Certified Engineers and Cisco Certified Network Associates.

This will ensure that their support staff is able to install, configure, operate, and troubleshoot routers and networking equipment. They'll have an advanced skill level when it comes to mitigation of security threats, introduction to wireless networking concepts and terminology, and performance-based skills. They'll also be able to:

- Configure static routes, packet filtering, and network address translation.
- Set kernel runtime parameters.
- Configure an Internet Small Computer System Interface (iSCSI) initiator.
- Produce and delivering reports on system utilization.
- Use shell scripting to automate system maintenance tasks.
- Configure system logging, including remote logging.
- Configure a system to provide networking services, including HTTP/HTTPS, File Transfer Protocol (FTP), network file system (NFS), server message block (SMB), Simple Mail Transfer Protocol (SMTP), secure shell (SSH) and Network Time Protocol (NTP).

Pricing

Pricing is always a factor when comparing hosts, but you may be surprised to hear that it's often not as simple as it would seem.

Pricing for server hosts is often more complicated than simply comparing the price of two packages because many times, hosts will be having sales, they'll show pricing based on a 3 year contract, or in amazon's case it's not displayed monthly at all! Amazon shows pricing per hour of usage, and while this idea may seem great at first, it has a hidden flaw.

A pay as you go pricing model will never be consistent, this means that you could experience a bill that is significantly higher than you were prepared to pay! A better pricing model for most businesses and organizations is the no contract, monthly pricing model. This ensures two things: 1) you can always switch hosts or downgrade without the worry of a cancellation fee and 2) you'll always know what to expect when it comes to your monthly bill.

But the number is only one side of the equation. To get a full idea of the value provided, you'll need to understand the server host's performance as well.

Performance

Several factors contribute to the performance of a server, whether it be a cloud server, a dedicated server, or some other hosting option. For starters, you've got the CPU or in the case of a cloud host/VPS vCPU. After that, you've got your RAM, Hard Disk (speed and space), and lastly your port.

Each of these plays an important part in the performance of your server, and if one lags behind the others, you'll likely experience a bottleneck and inferior performance. Hard Disk and your port are often the hidden reasons that some hosts perform better than others. This is why it's important to take each of these into consideration when choosing a server host.

Let's take a look at two fictional servers: server host A offers a server with a great CPU, great RAM, 1 gigabit port, but a slow Hard Disk. The price is \$100.

Server host B offers a server with the same exact CPU, the same exact RAM, the same port, but a solid-state drive. Their price is \$140.

While many would say that the \$100 server is the better value (and when purchasing many services and products it would be) in a server hosting situation it's actually a worse value. See when that server comes under load and needs to serve files from its hard disk, the slower disk will present a bottleneck. This results in longer load times and an overall worse experience for end users. Many first-time server owners make this mistake and when attempting to remedy their slow server throw more RAM and CPU at the problem. This leads to a lot of wasted money and a lot of headaches.

8

HOW TO CONTINUALLY EVALUATE YOUR HOSTING NEEDS

We're not going to lie to you. This is the hardest part of this entire eBook. Knowing how your hosting needs change is a skill that even advanced IT professionals may struggle with. This is because growth is often hard to predict. Take for example a startup who's just beginning to attract users for their mobile app. It can be difficult, near impossible to know how many users are going to be using their app a month or two in the future. The app could explode in popularity overnight and attract thousands of new users. Or it could remain at a steady growth rate of a couple hundred users a month.

Even enterprise businesses may have trouble evaluating their future hosting needs. Growth is often unpredictable and even if they've had a stable curve the last few months, they may experience a drop or an increase in resources needed at any moment.

If you don't accurately evaluate and predict your hosting needs, you may pay for more resources than needed (resulting in wasted funds) or suffer from not having enough server resources and suffering from a slowdown or even outage as a result. These effects can cause end users to become frustrated and as such you may lose revenue, suffer damages to your reputation/brand image, and more.

Evaluating your hosting needs is crucial. It'll result in the best possible experience for you and end users at the best price. It's difficult though, and you have to understand a few key points. Here's what you need to know:

How often do you need to evaluate your needs?

Let's start with the simplest part of the equation. What kind of schedule should you follow for evaluating your hosting needs. The answer to this question will change based on what size/kind of organization you are, so we'll lay out an answer for each of the major sizes: small business, medium business, enterprise, and startup. We'll take a look at small businesses first.

As a small business, you'll actually want to be analyzing your hosting performance and needs on a regular basis. Checking your server's performance on a weekly basis is a great starting point. By doing so, you'll understand when your server is reaching a tipping point and needs upgrading. At this stage, you may be upgrading a server often, and that's important if you want to maximize your bang for buck in terms of server performance. Check each week to see the number of daily active users and begin tracking the speed at which that is growing. If you using BI Software and machine learning tools, predicting what that number will be a week, a month, and even a year from now is feasible. Then, use that prediction, but check it for accuracy each week and if it's off adjust. You'll want to begin building an accurate growth model while keeping tabs on server performance.

Speaking of server performance, let's discuss when you may want to upgrade your server. Remembering those server sizes in the first chapters of the book, you'll want to be on the lookout for when you'll be making a shift from one size to the next. When nearing this point, you'll want to prepare for a larger server. This would lay out an upgrade plan similar to this:

- Upgrade at 20 active users
- Upgrade at 100 active users
- Upgrade at 500 active users
- Upgrade at 2,000 active users
- Upgrade at 5,000 active users

Now obviously, you can (and in many cases, should) upgrade more often. But, this is a good minimum upgrade path to follow. Medium businesses are at a point where they may begin to level out in terms of growth and grow at a more predictable rate. This means that they can check server performance at greater intervals than a small business. A bi-weekly schedule will work well, with a report on server health and projected growth being great resources.

An enterprise business/organization will have the greatest interval for checking server performance. If you're here, a monthly report will do, and you've likely got IT staff monitoring it daily anyway who can notify of any urgent changes. Startups can vary, because the definition of a startup varies so much, so we'll do our best to breakdown what different startups should do. First, if you're funded, the answer may be simple. Buy a large server and worry about performance later. This may sound strange, but as a startup that's funded, you're likely looking to achieve very rapid growth. If that's the case, purchasing a smaller server and upgrading when the time comes simply won't be sufficient.

Purchase a large server and wait. Check bi-weekly until you begin approaching your server limit, at which point you'll likely want to check more frequently until you upgrade once more. Smaller, unfunded startups can go with a weekly check like small businesses previously mentioned.

When and How to Upgrade

When you begin to grow out of your current server, you'll need to upgrade. If you're utilizing a public cloud, they'll upgrade for you. But, that's not always a good thing. If you're using another form of hosting, upgrading will take a little more effort, but there are some benefits to that. Let's start this section by taking a look at the public cloud (AWS).

When using AWS, you're on a pay-as-you-go model. This means that you only pay for the resources you use, and that the service automatically assigns or removes resources as your needs increase or decrease. At first, this seems like a great idea, but there are a few major downsides and reasons that you need to constantly evaluate your hosting needs.

First, because you pay for resources as you need them, when you receive a spike in traffic, your bill will spike up as well. In this case, your end users won't experience any issues, but the bill may be significantly higher than what you've budgeted for. As you begin to use more and more resources, the cost of being on the public cloud will become more and more prohibitive. At some point you will need to move to a more budget friendly solution. This is why it's important to keep track of your usage and understand how and when to upgrade from AWS.

Often, small businesses and startups will use AWS as their needs are hard to track. They may be in a major growth phase where the instant scalability is more important than the funds. Once your growth begins to even out and become predictable, you are going to want to consider moving away from the cloud. The price you're paying is simply not worth it when you can purchase a dedicated server for 3x less and scale up later when needed.

Which brings us to the next point, upgrading your dedicated servers. We laid out the minimum upgrade framework above, but honestly you'll need to upgrade more frequently than that. Those are often just the major upgrade points, in which you'll switch from one server to the next. In between you may perform several minor upgrades which could include:

- Increasing RAM
- Adding or upgrading hard drives
- Switching to a more powerful port
- Increasing the amount of traffic you can use
- and more

These minor upgrades help fill in the gaps between moving to a new, more powerful server. How do you know when to upgrade though? Well, the easiest way is to wait until your end users experience slower speeds, and upgrade then. This however is problematic as you'll be waiting too long and providing end users with poor user experiences. Remember the easy way isn't always the best way!

A better method for upgrading servers is to calculate how many end users your server can support. This is a little tricky, and requires some effort on your part, but you'll know when you need to upgrade. What you'll want to do, is track the number of active users and the utilization of your server's resources. Take these measurements several times so you have a large amount of data to work with. When you're done, you should have something like this (numbers are completely random):

Active Users	CPU	RAM	Disk	Network
5	10%	40%	2%	6%
3	5%	40%	1%	0%
9	17%	55%	4%	10%
43	83%	100%	20%	50%

With this spreadsheet created, you can begin to calculate the average resources one user takes to support. We'll explain how to do this using the example above. Let's start with CPU. Take your CPU usage and divide it by the number of active users for each record. The first day's average CPU use per user is 2%, second day is 1.67%, third is 1.8% and fourth is 1.93%. Combine these together and take an average. In this case, 1 active user takes up about 1.85% of your server's CPU. Using this info we can calculate that the server can support about 54 active users. Now you just need to watch your active user count and when it begins approaching your theoretical maximum, upgrade. This process is the same for all other server resources.

Another way is to watch your server's current resource usage and if it begins approaching 20% upgrade. This however won't give you a way to predict when the upgrade is needed, but will result in a reactionary upgrade.

There's been a lot of content in the eBook, so to wrap things up, we'll keep it simple. Think of this section as your cliff notes, your outline, your TL;DR of the eBook. Let's dive in.

In Chapter 1, we discussed how to assess your hosting needs. Even though everyone's needs are different, you can use the following to see what category of server usage you'll fall under:

- Very Light | 20 or less active users
- Light | 21-100 active users
- Medium | 101-500 active users
- Heavy | 501-2000 active users
- Very Heavy | up to 5000 active users

Once you identify which category your server will fall under you can use the following as a starting point for choosing your server. Note that you may require more resources than what's mentioned here, or you may need less. This is just a good base point for choosing your next server:

- Very Light Usage | Dual-Core CPU (1.9GHz), 2GB RAM
- Light Usage | Dual-Core CPU (3.2 GHz), 3GB RAM
- Medium Usage | Quad-Core CPU (3.3 GHz), 8-16GB RAM
- Heavy Usage | Dual Quad-Core CPU (2.4 GHz), 32GB RAM
- Very Heavy Usage Dual Octa-Core CPU (2.8 GHz), 64GB RAM

In Chapter 2, we provided you with a checklist for comparing your hosting options. There wasn't much content here, but rather just a spreadsheet that you can plug in different server hosts offers to see which is best. Here's the link to that.

Chapter 3 saw us take an in depth look at AWS. The important facts to remember are:

- AWS' vCPU is on par with dedicated servers.
- AWS' Hard Disk Speeds are faster than SATA, but slower than SSD.
- AWS' Memory is about 30% slower than the DDR3 RAM commonly found in dedicated servers.
- There have been 5 major outages since 2006, but uptime is still one of the top for cloud hosts.
- Their pricing can be 3x higher than a comparable dedicated server. Even private cloud hosting is significantly cheaper.
- Proprietary knowledge makes using AWS even more costly both in terms of time and money.

After that, in chapter 4, we took an in-depth look at dedicated servers. We created a simple one sentence definition: A dedicated server is a server in which all resources are dedicated to one client. And after, took a look at some considerations when choosing a dedicated server. These were:

- Choose whichever OS you're most comfortable with
- Ensure your CPU is server-grade, and one of the latest generations
- Always get DDR3 RAM
- Opt for SSD when available for increased performance
- Ensure you'll have enough traffic. (most dedicated servers come with an abundance, but it's always good to check).

In chapter 6 we discuss managed hosting and unmanaged hosting. We defined managed hosting as hosting in which a server host also offers maintenance and management of the server. We laid out some of the benefits of managed servers in the chapter and here they are again:

- You don't need much technical knowledge to use a managed server
- You can save time as the server host will take care of a lot of routine tasks for you
- Your IT team can focus on other, more important uses of their time

Remember, AWS doesn't include managed services, so if you want this you'll need to go with a dedicated server.

Next, in chapter 7, we detailed what you should be looking at when choosing a server host. Here's a quick rundown of what you should be looking for:

- Get an overview of the host. How long have they been in business, how well are they reviewed.
- Don't mind uptime/SLAs too much. Most hosts have similar uptime, and SLAs often are setup in a way which they won't refund you.
- Check for their support options and hours. 24/7/365 and multiple means of contacting are what you need.
- Obviously price is a factor in your search, but don't look for the cheapest host. Look for the best bang for buck host.
- Performance is a major factor. This mostly boils down to what server packages they offer, whether it's server grade hardware, and whether they use latest generation processors.

In the last chapter, we discussed how you should continually evaluate your hosting needs. In it we detail a method for calculating how much resources a single user takes from your server. The formula is:

$\% \text{ resource (CPU or RAM) utilization} / \# \text{ of active users.}$

This should be taken over the course of several days (2 weeks is recommended) for the most accurate result. Once it's done, you'll know your server's theoretical maximum user count and be able to upgrade when necessary.

More questions?

Do you have more questions about dedicated servers, AWS, or anything else server related? Reach out to our expert staff now.

Contact us by visiting www.serverpronto.com/contact-us.php