7/17/17

**Complete Infrastructure Environment Build**

If you own your own Infrastructure for hosting, testing, development, production, or even a redundant system for backup purposes:

1. You have complete control to make changes any time you want;
2. You have efficiency, prestige, and elegance in making professional presentation to sell or promote your product;
3. You have smooth and productive progress in software, employees, and company status itself;
4. You are cost efficient and safe in learning, and experimenting your properties;
5. You have the ability to own and deploy as many backup systems as you like never fearing problems, attacks, and down times;
6. You cut the costs required to pay third-party “engineers” and costs to pay in-house employee to talk with those engineers and most important of all, reduce the time to address the problems (that come from long wait times on the phone);
7. You own your own database and its credentials, own all other credentials in the infrastructure and have no one dictating the company from outside or within;
8. You also cut others costs like service fees to other third parties;
9. You have a great hosting power, such as large RAM 4GB—200GB, large HDD space 100GB—10TB, any powerful CPU in the market running your server, and unlimited bandwidth. Incidentally, paid hosting isn’t cheap. Share hosting, virtual machines, or even “dedicated servers” prices explode as you want more increments in the specs mentioned above. We can’t see the “dedicated machine” they claim to have made for—whether it is real or a lie.
10. You have better security proof from hacking. Companies pay another company because they were afraid of hackers and security issues. In reality, data breaches, service breaches, and hacking occurs with their ways and they have are still helpless. Their credentials for these scattered infrastructures are spread out among different current and past employees that they are completely at the mercy of the attackers in the dark.

 This infrastructure environment can be used for application software, web applications and development, or smart phone apps.

My infrastructure environment includes:

1. Apache Server
2. SSL certificates as many as you like for different domains or sub domains
3. MariaDB or other database options
4. PHP My Admin
5. Tomcat Server or Glassfish Server for Java2EE apps
6. Perl, Ruby, PHP or Python capable Apache with CGI settings for them
7. Eclipse, Net Beans, or other IDEs with unit testing capabilities, version controls like GIT, and any other desired tools like Maven or Gradle
8. Filezilla FTP Server
9. Mercury mail server or other mail server options
10. Secure network configuration with static IPs or DDNS services to link to outside domains
11. Any Windows machine or Linux machine. Operating systems don’t have to be server, but a PC can be turned into a server.

I have the knowledge to build robust, secure, and scalable development environment from scratch with time efficiency. A package like XAMPP has great disadvantages:

1. 99% of the people don’t know how it’s packaged and how to alter and secure it.
2. Sub packages, such as Apache, MySQL or MariaDB, phpMyAdmin, PHP, PERL, Tomcat, mail server, and other software tools are constantly upgrading versions. XAMPP cannot keep up with them. The way they package those individual packages become a great difficulty for one package to upgrade or uninstall without affecting others.
3. Most XAMPP users (developers) have no clue with in-depth knowledge for install, configure, administration, and life cycle managements for those bundle packages. If they ever need to go to production, commercial demonstration, or live testing with real domain and real IPs (not with localhost), 99% of the developers are stuck, have to approach setup with **expensive disorganized commercial setups,** or have to spend considerable efforts with time, money, and energies to move the software stages like staging to production, SSL installed URL, localhost to real domain, and etc.
4. XAMPP has no real support.

In truth, it is more challenging intellectually to setup an infrastructure environment or an app development projects with various items than actual coding or debugging. To understand and control the technical infrastructure setup and truly own and maintain the software is extremely important. All companies I worked for or worked with in my career so far have bits and pieces of their software infrastructure in all over the places. When modifications were needed, times dragged on tremendously for even simple tasks.

For example, GoldQuest Refinement Services, LLC, has their domain hosted with Go Daddy, SSL Certificates purchased for $160/year, their software hosted at RackSpace for $700/mo (little disc space 5GB or less, 512MB ram, with bandwidth limit and slow processor), their customer signup data collected at some third party company about $30/mo, and transfer that data to manage at SalesForce database $1500/six month, some third party mail advertising company $50/month for emailing. Their monthly bill for the infrastructure is at about $2000 or more. When problems occurred, or modifications needed, the times to address or wait are always 2hrs minimum on phone, but this could carry on 3 days or more or even a week or two to have the problem resolved. I build an in-house hosting, testing, backup and development environment during my time there. I learned their infrastructure in about two weeks and built another one in just two more weeks. The owner couldn’t believe that monthly bill he paid could be reduced down to $0 or 90% less. The owner wanted to sell his company or sell shares of the company publicly. I built an in-house infrastructure for him that he could make professional presentations for investors and others outside of the company’s building. Without my infrastructure, it was not possible before me.

Certo Bros Distribution Inc, has no competent high-level software engineer or even real IT technician in their IT or maintenance department. They have a very poorly setup network that is compromised according to even minimal default standards, many visible traces of faults and tampering (hacks) in the network infrastructure that I had proven to a guy who was in charge of it, unacceptable setups in their Cisco router settings that same guy Ed “Doug” Douglass has no idea about, and many other issues that I didn’t have the time and energy to address. Even for a small issue like SSL certificate expiration is a big issue for the company and email server has to be down for couple of weeks because the guy in charge never knew or cared to know how to install new SSL certs. The guy to pay to do was not available for about two weeks. For all their machines, problems, software applications, and infrastructures, the executive members pay unqualified people to talk with paid third-party services all day. The executive bosses love and feel comfortable around the stupid people as their employees. My own boss revealed and whispered to me about that from the beginning. I talk about “Doug” with his name, but others (I don’t have time to mention in details) in maintenance department, sales department, executive branch, and a manager (Kenny) were even worse. They were not only unqualified uneducated lowlifes, but also shameless and rude liars. I’ll write more about them in details when I have time. I estimate their IT support bill from third parties is about $5000-$10000 weekly, and they have about five in-house maintenance technicians, and “IT staffs” who are on salary in addition to that spending.

Banks are also perfect examples or victims of these issues I am talking about. I worked for First Niagara Bank. I was shocked to know that most everything they had in software and infrastructure came from third parties. They have direct access to or clear understanding of anything. Still, banks have to spend and hire in-house IT support staffs who sit on telephones to address the problem with the software and infrastructure they are renting from multiple companies. Much of their infrastructure setups and software application they use are actually very simple in my understanding. The third parties brainwashed banks to buy junk services and software seem like big deals and pay high prices. Their combined bills for both third parties and in-house employees I estimate $100,000--$50,0000/wk.

Another critical issue I see is that the corporations of big and small alike became a salve to low-level unqualified employees because of scattered infrastructure setup style. This style has multiple third-party accounts. The unqualified employee knows all these accounts’ credentials. Although he or she was unimportant, uneducated, unqualified, and unskilled, company became dependent on the little peculiarities, credentials, and social knowledge of that employee (about who to call and talk to). Now that low-level unqualified employee calls shots and asks for big moneys. Shallow companies fear truly intelligent and capable people to become their employees because they are afraid true intelligent intellectual will call shots and control the company—very ironic. All the problems I wrote above in costs didn’t include hidden costs from delays and production halts. Incidentally, I was shocked when I found out most hospital equipments and IT infrastructures were set up the same way.

This kind of disorganized scattered infrastructure is like standard practice or expectations in present for more than 90% of the companies. I always spoke out to everyone within the companies and everywhere else that companies and businesses should strive to own, control, and understand their own infrastructures. The executives and share holders were afraid that building and taking control of the infrastructure would be time consuming and expensive, and would be too difficult to maintain by themselves. In truth, it is more problematic, more expensive, and less productivity for any company with their scattered disorganized third-parties-owned infrastructures and software. Companies are willing to spend money with salary employee coders (“software engineers”), or unqualified people who they will dub as “IT Technician”, but never thinking to build and own an infrastructure (a foundation) on which their software will live or exist and keep real engineers as employee.

Contact:

Christopher McGrath

Mcgrath\_ch@hotmail.com

716 240 7786