Word Processing

Norma Fitzgerald

Operating Systems & Architecture

SERVER VIRTUALIZACION/ ACTIVE DIRECTORY DOMAIN CONTROLLER

STUDENT: DIANA CABEZAS. STUDENT #: 202373. DATE: 06.04.2023

Table of Contents

Project Introduction

The goal of this project is to create a fully operational server to run the network as a domain controller. This report is intended to show the method used when building the server and implies a clear process of completed tasks. This server is intended to allow for greater security on our network, as well as easier exchange of information between specific users of the system. We will also have a web server using IIS server we will add a web page to demonstrate its functionality.

Scenario: You are the assistant to the Network Administrator of a busy product services company called **DigiTech**. You are required to build a fully operational Server that will be running as a Domain Controller for the Network.

Installing the network and setting up the network infrastructure:

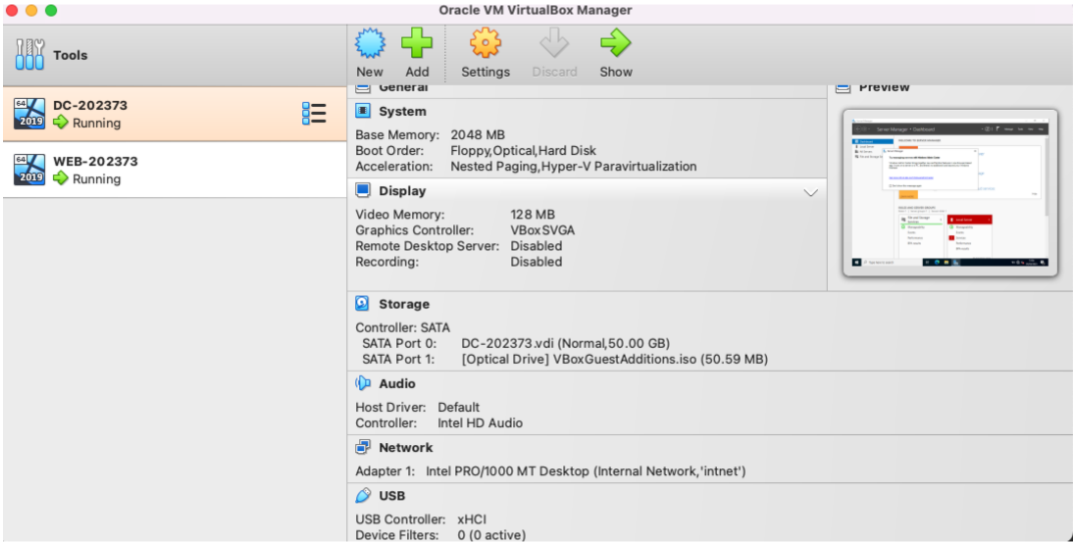


Fig 1 – Using VirtualBox two virtual machines has been installed. One VM will act as the Domain Controller Server and the other VM will act as a Web server.

Renaming servers

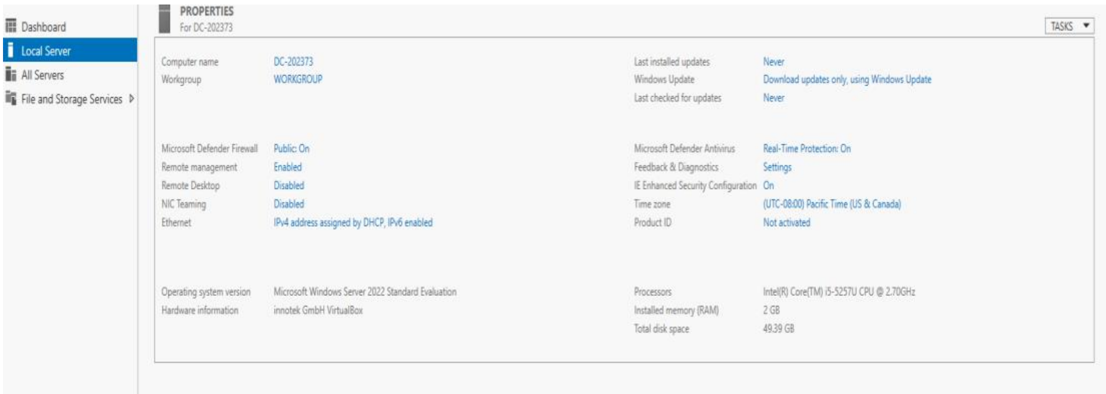


Fig 2 – Renaming servers. This server was renamed as DC-202373 using Windows System Properties.

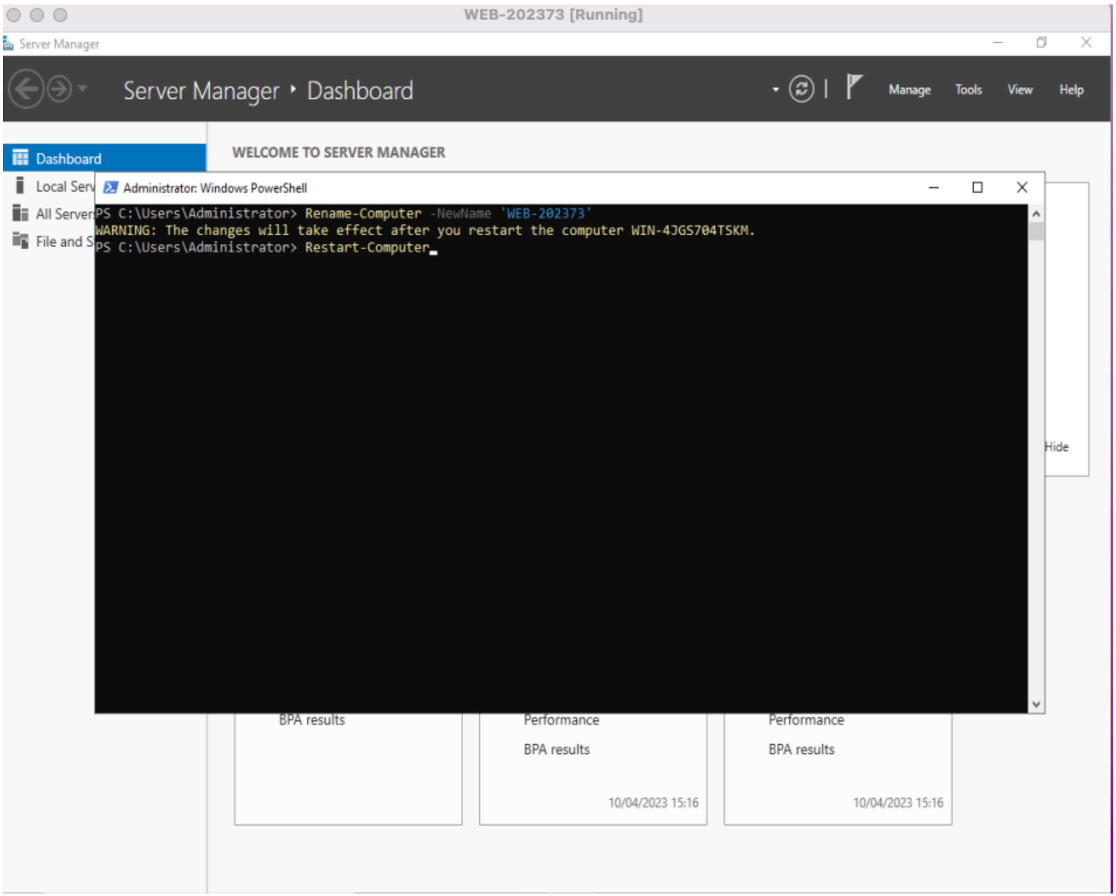


Fig 3 – Renaming servers. The second server was renamed as Web-202373 using PowerShell to rename it.

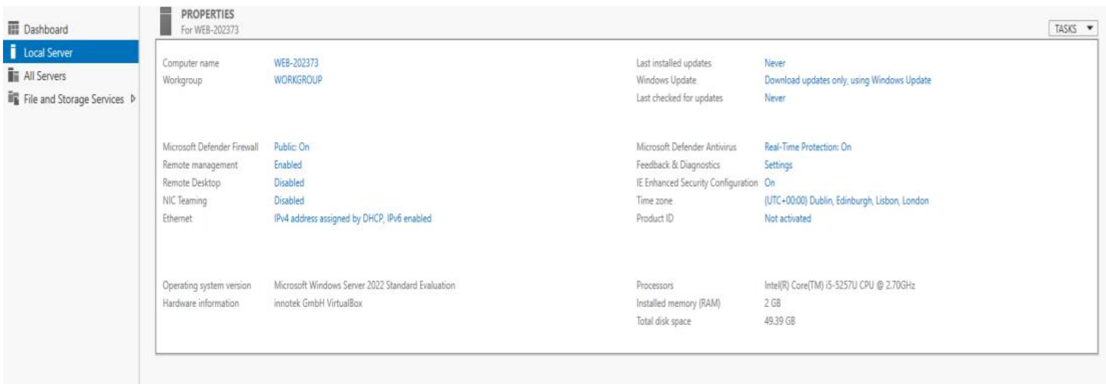


Fig 4 – Renaming servers.

Server address: 172.16.0.100/16

Default gateway: 172.16.0.1
DNS Server address: 172.16.0.100
Alternate DNS Server: 8.8.8.8

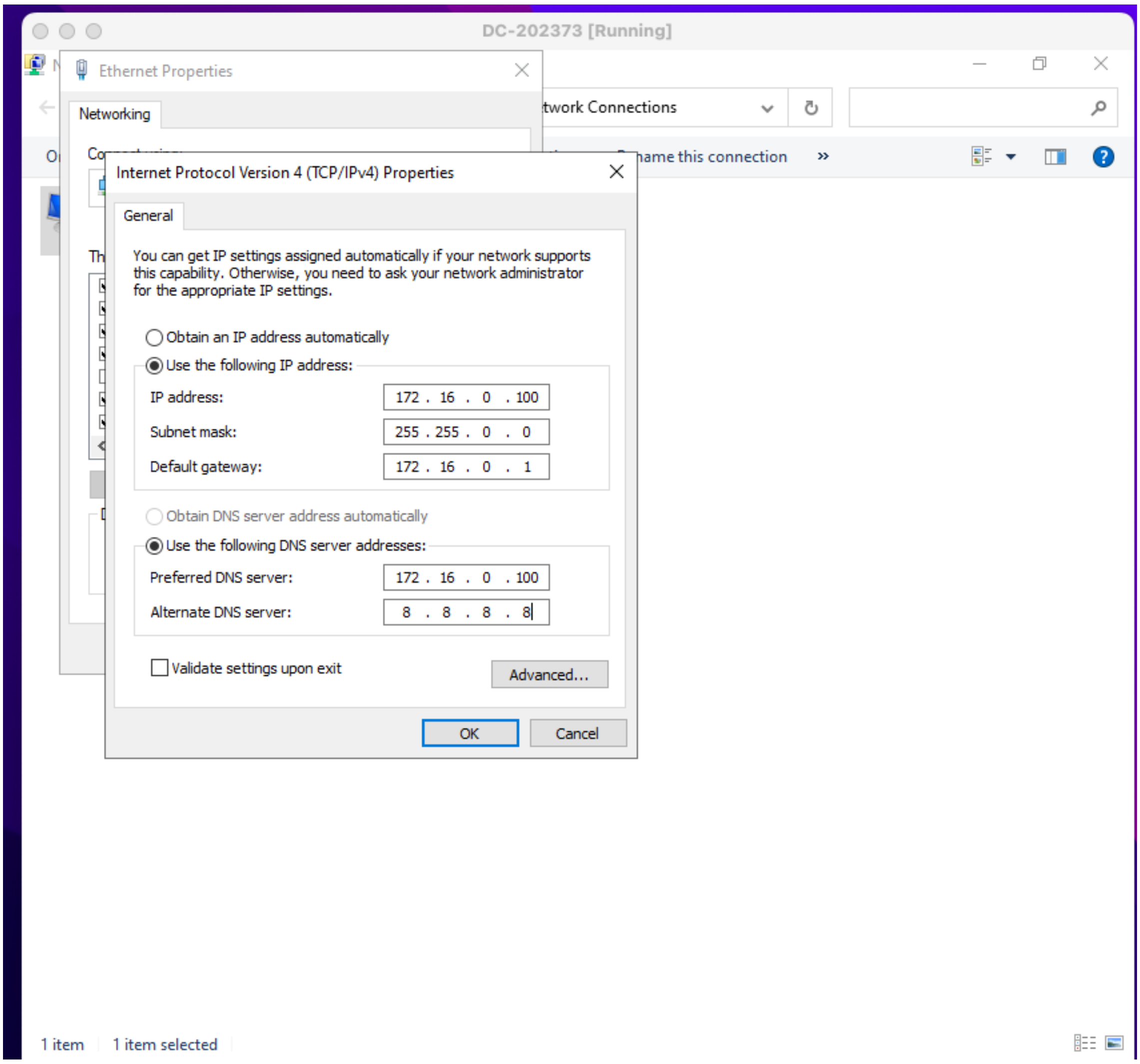


Fig 5 – assigning static IP addressing to the server.

Web address: 172.16.0.200/16

Default gateway: 172.16.0.1
Preferred **DNS Server address:** 172.16.0.100
Alternate DNS Server: 8.8.8.8

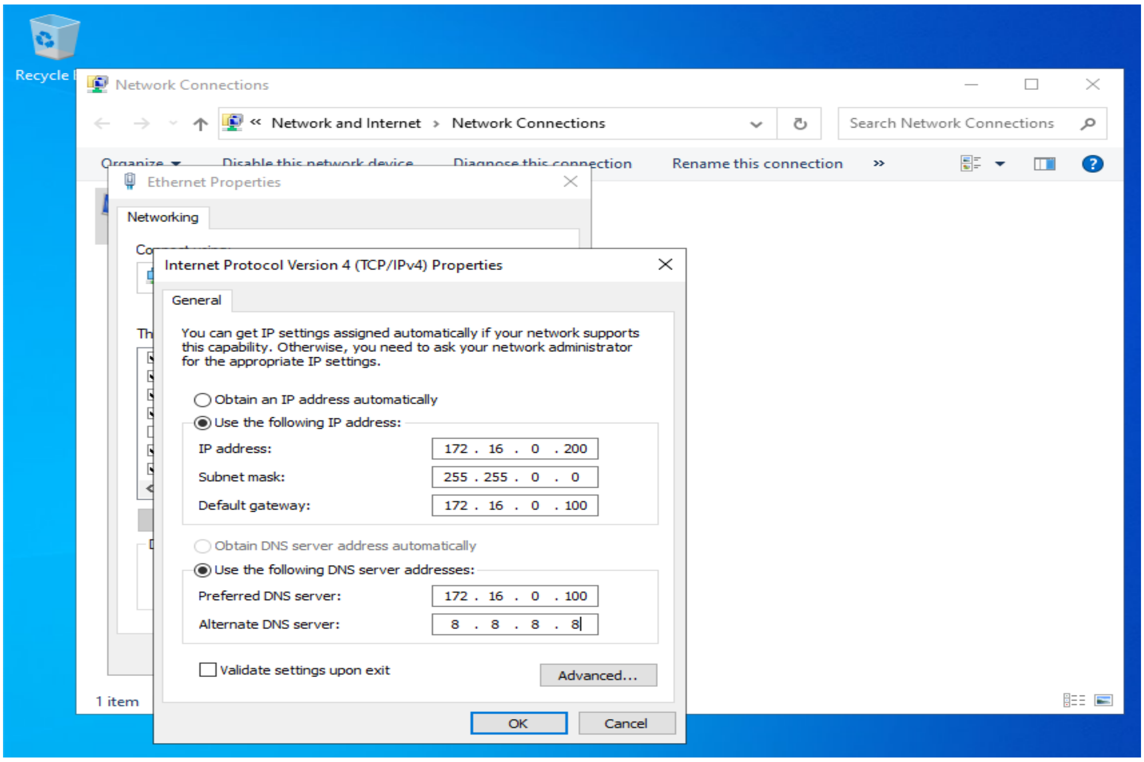


Fig 6 – assigning static IP addressing to the web server.

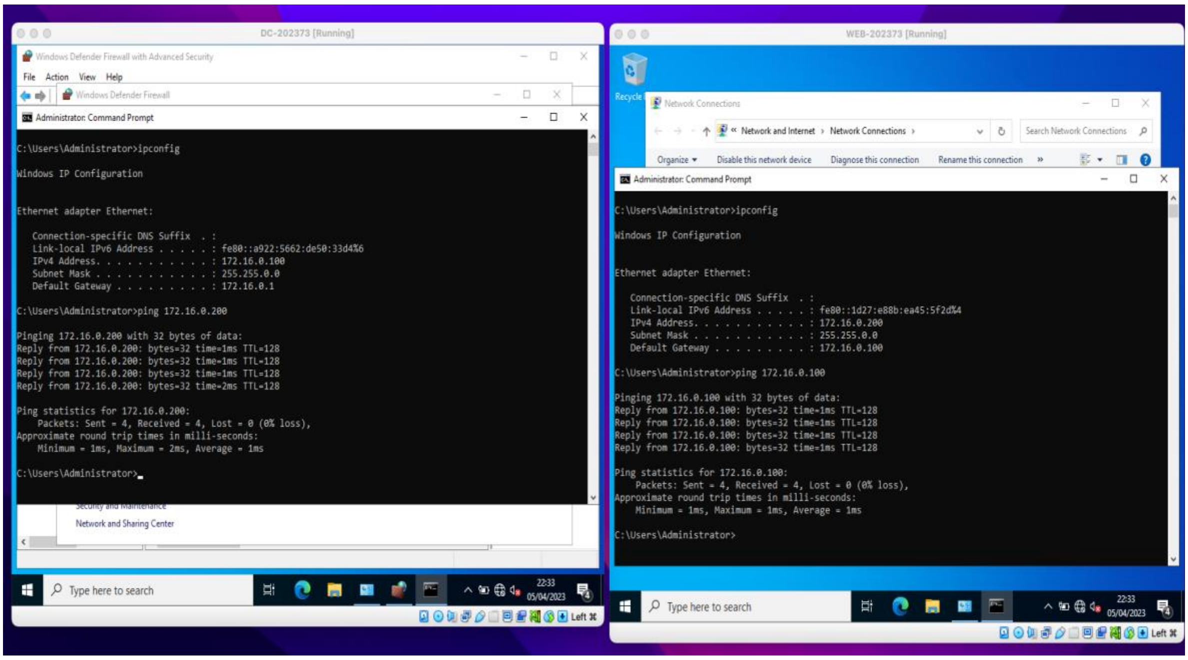


Fig 7 - Testing and demonstrating connectivity between the two devices by using the PING command.

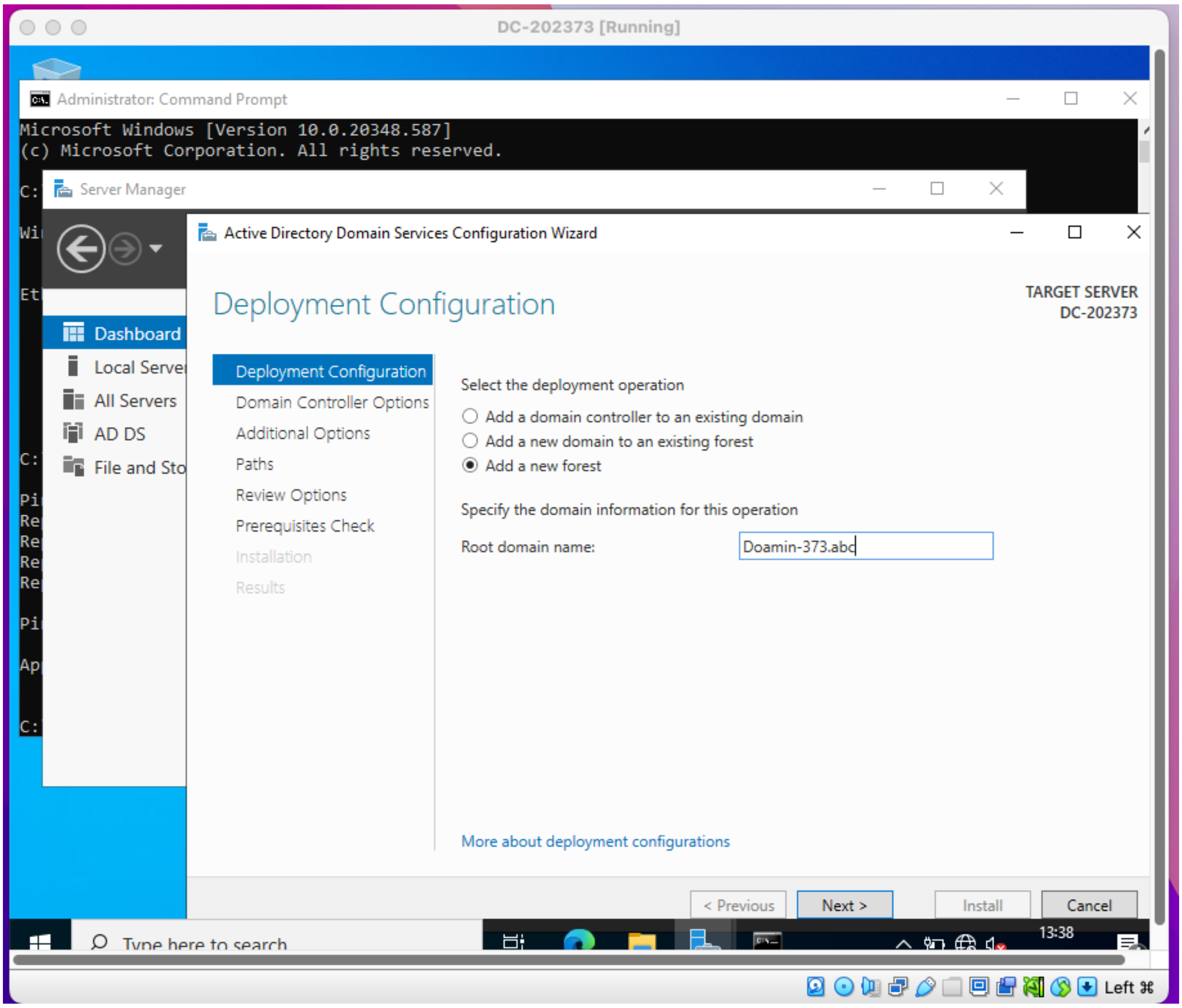


Fig 8 - Convert the first server (DC-202373) into a Domain Controller.

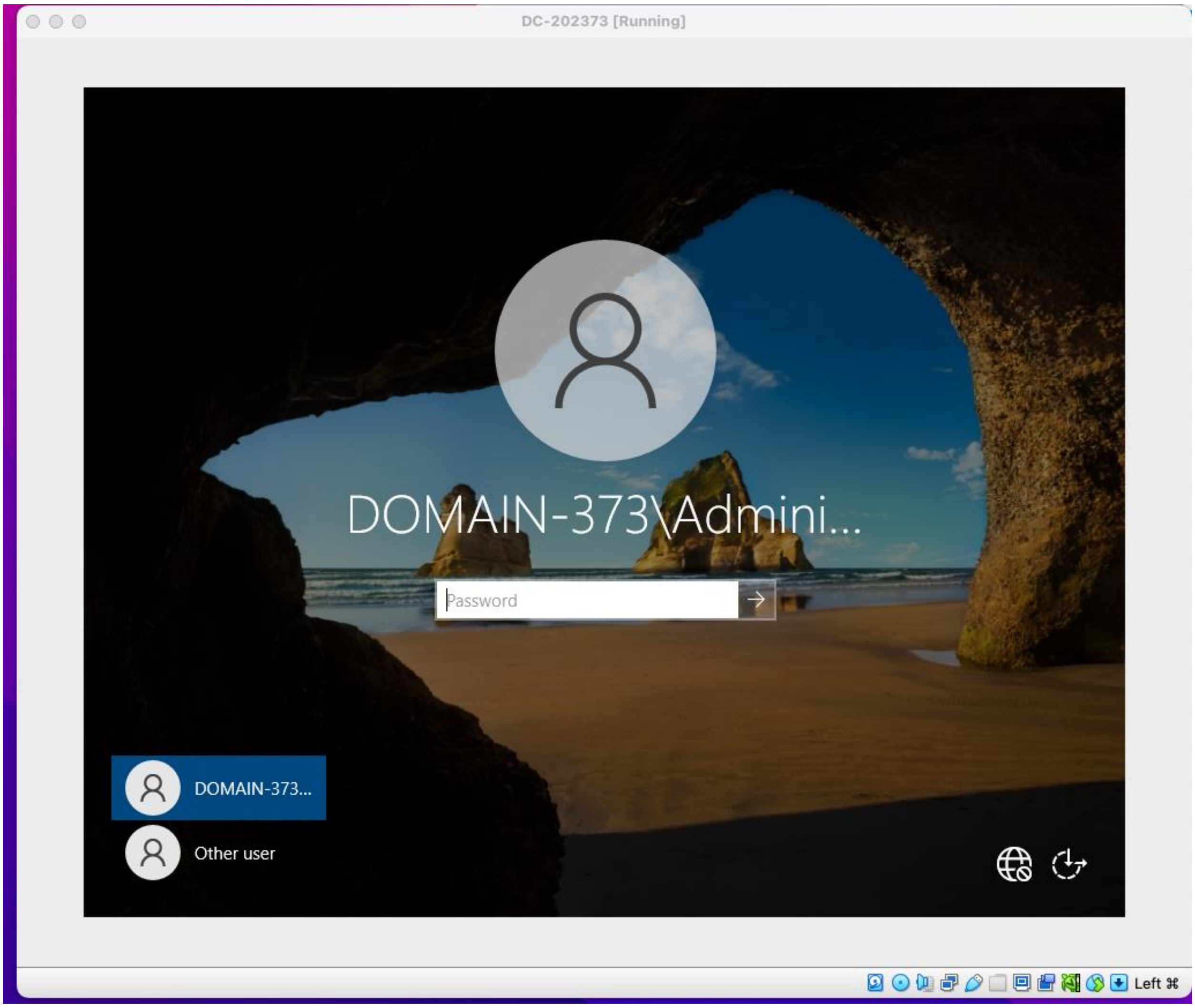


Fig 9 - Active Directory domain named as Domain-373.abc.

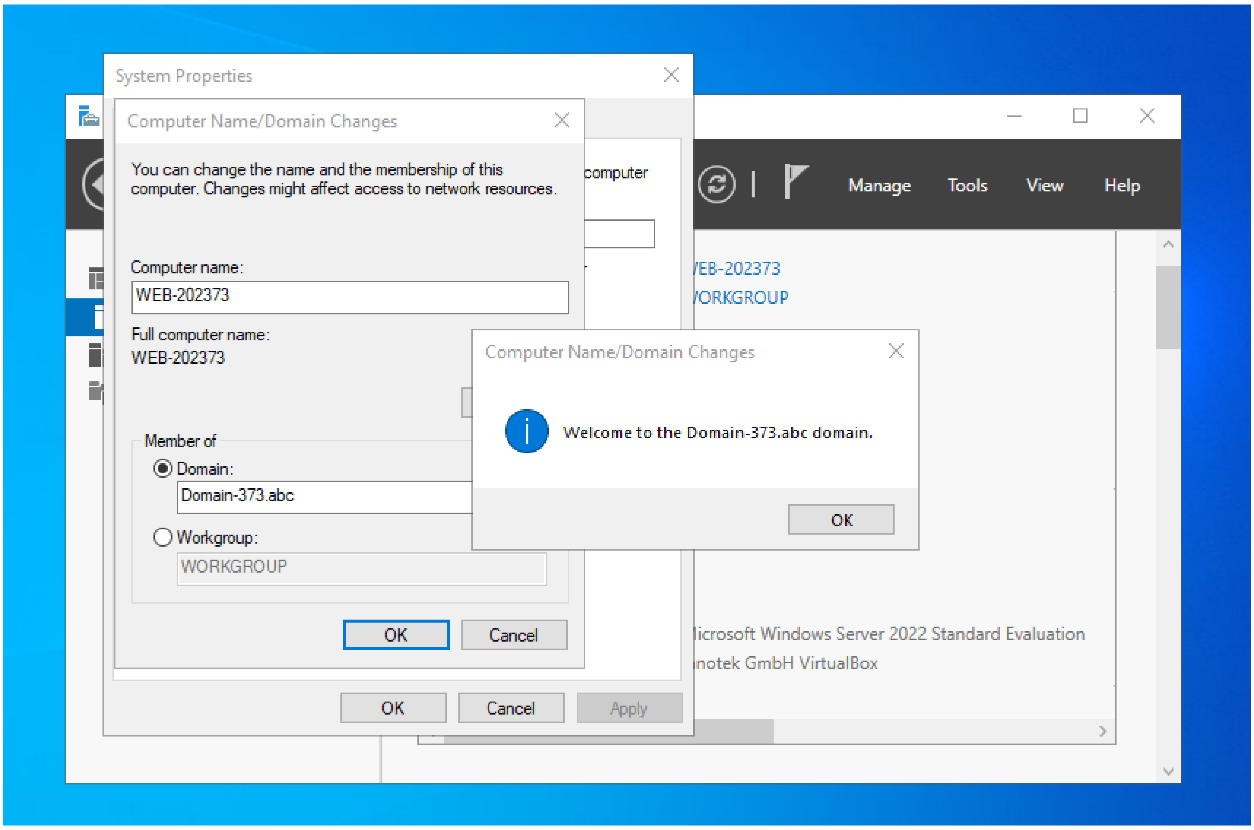


Fig 10 - Joining the Web computer to the Domain. After the Domain Controller has been installed.

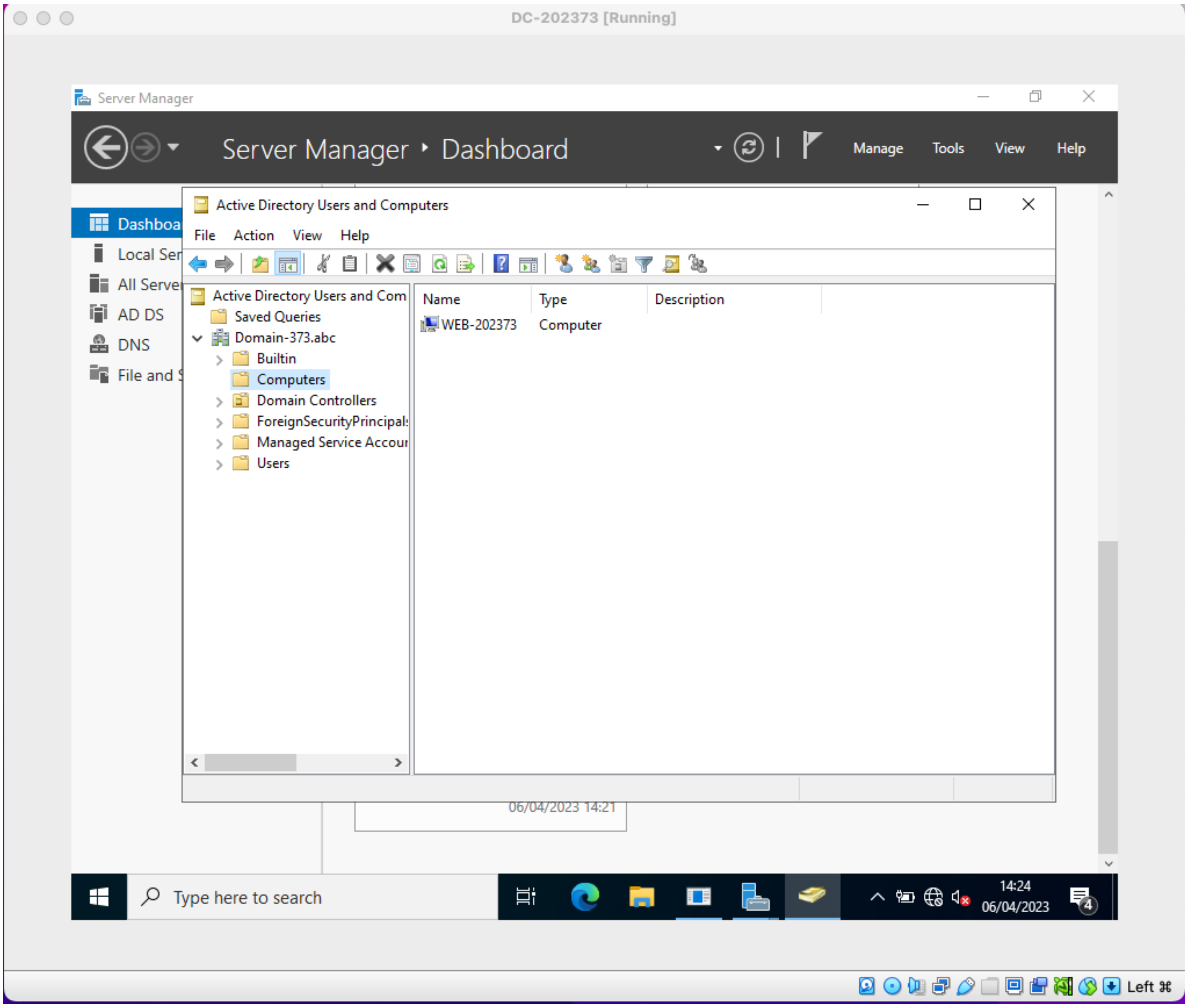


Fig 11 - Web computer to the Domain joined to the Domain.

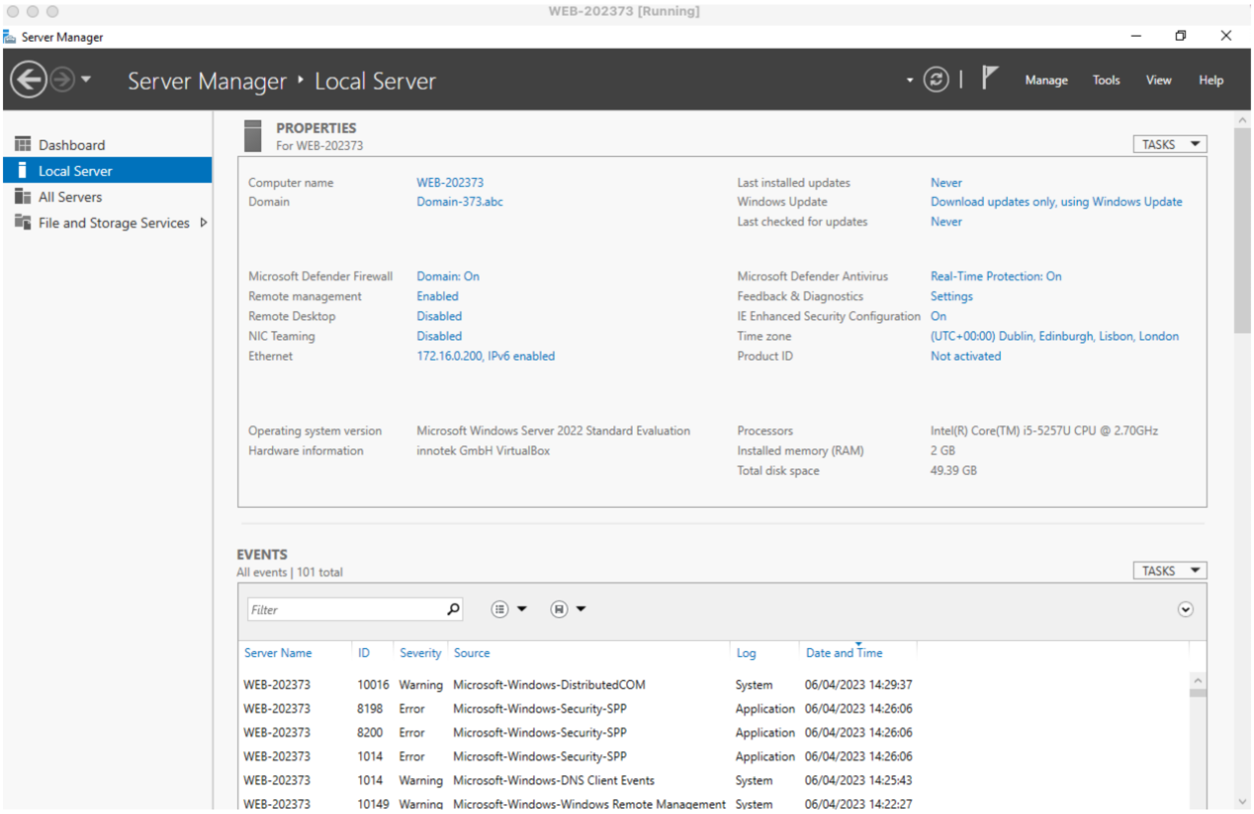


Fig 12 - Web computer to the Domain joined to the Domain.

Setting up and testing a simple web site:

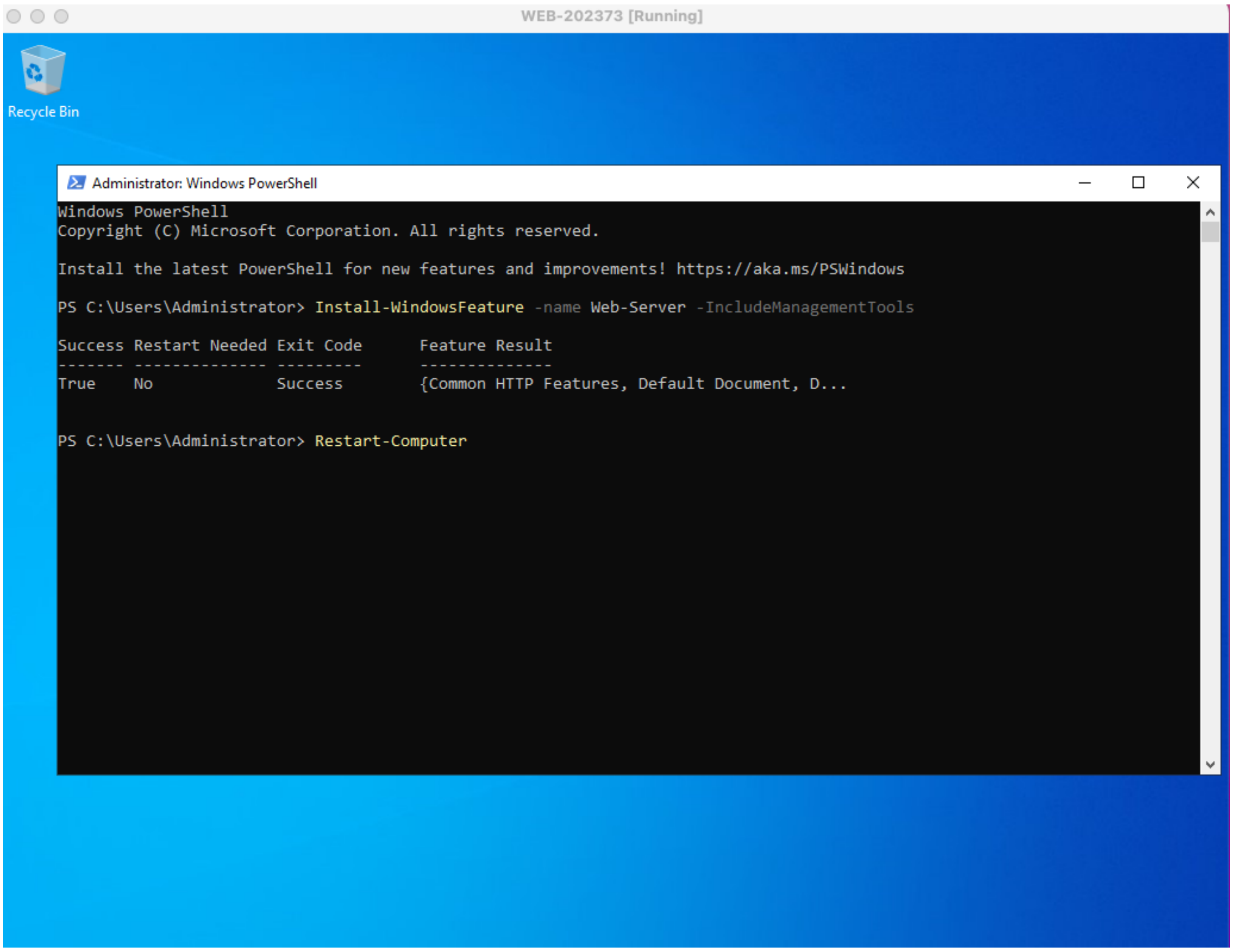


Fig 13 - Installing the Windows IIS Web hosting software onto the Server 2 computer. Using PowerShell commands.

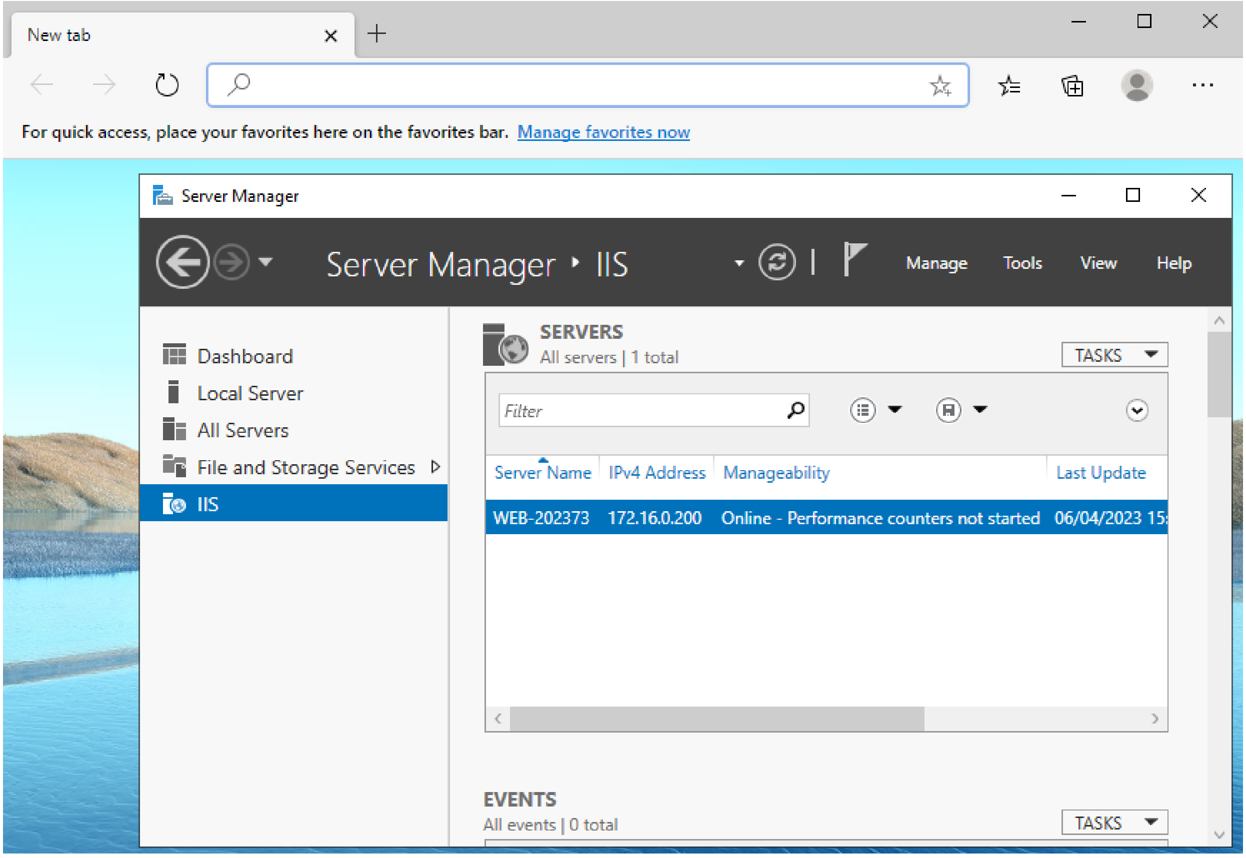


Fig 14 - Windows IIS Installed.

Fig 15 – My web site DigiTech.

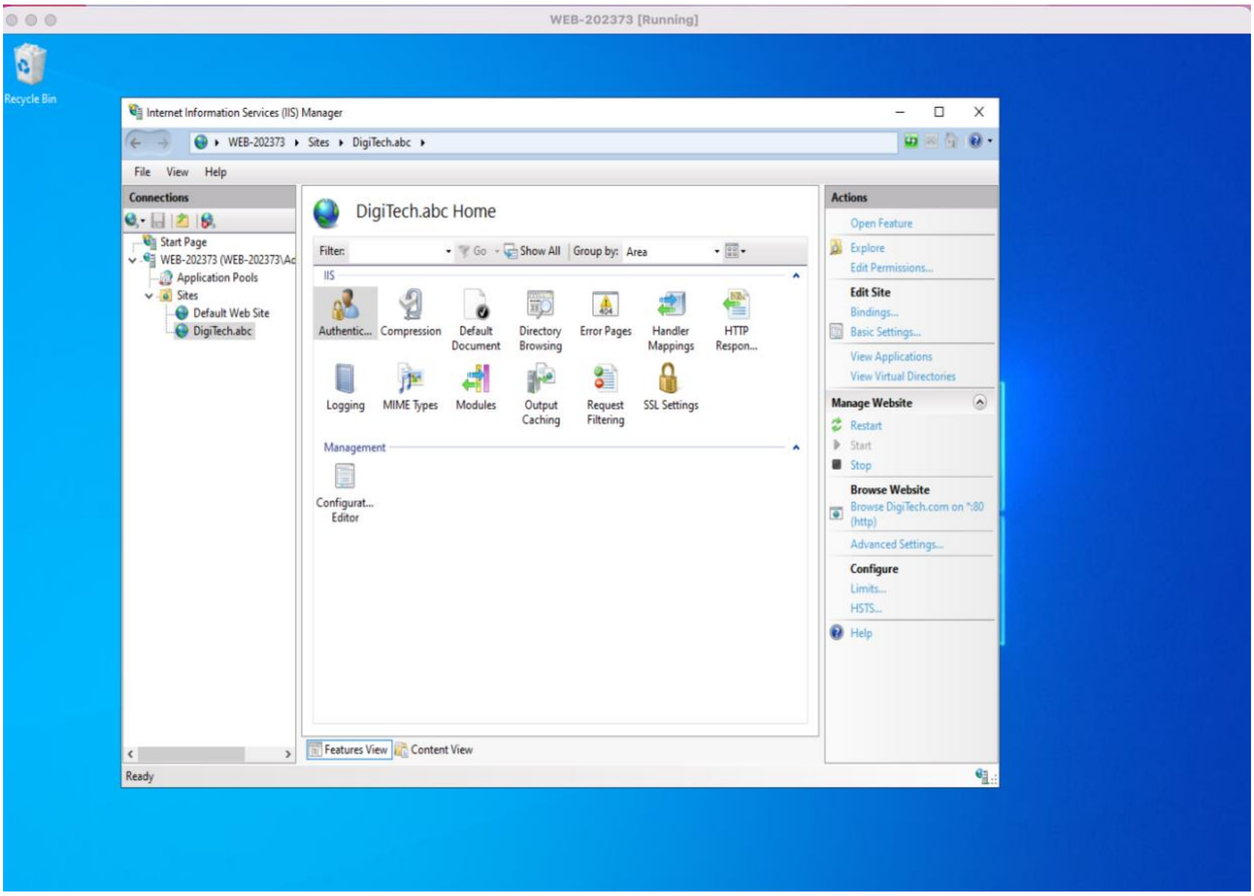


Fig 16 – Internet Information Service (IIS) and the web site created and set up on the web computer.

Use a web browser to demonstrate that the Web site is working.

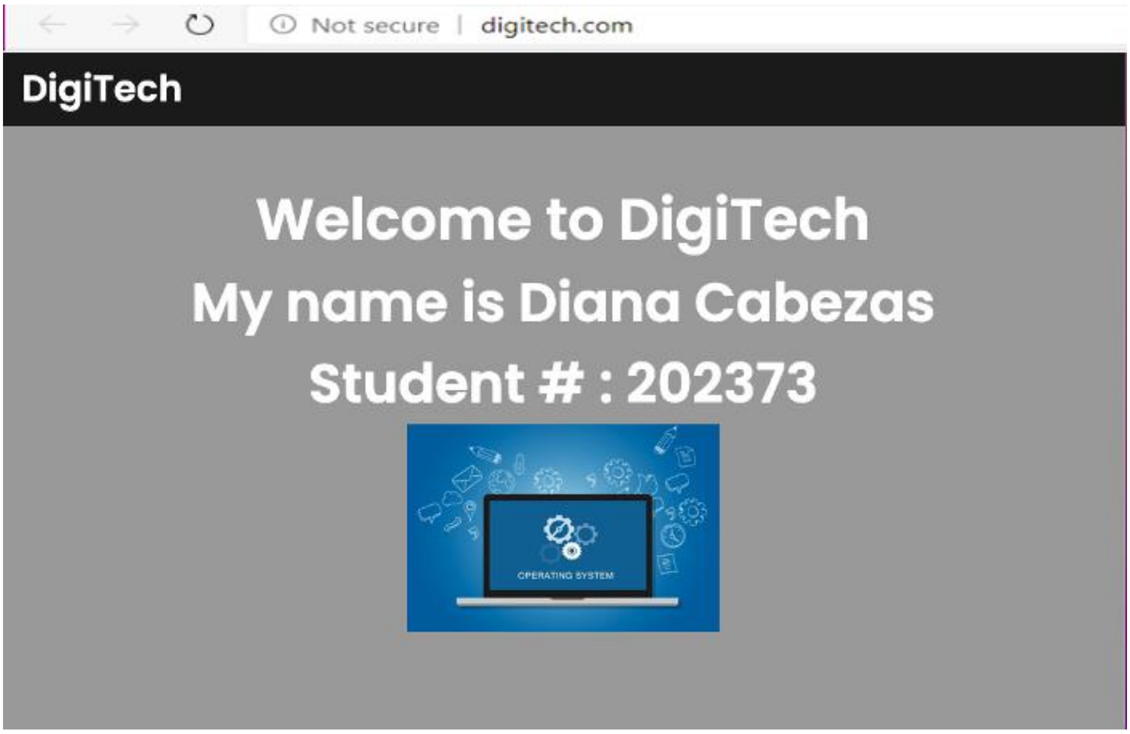


Fig 17 – DigiTech website.

Research

The Raspberry Pi and Arduino

Arduino is an open-source electronic device that reads and generates outputs based on inputs (such as light on a sensor, a finger on a button, or a tweet, such as turning on an LED or activating a motor). Arduino was developed at the Interaction Design Institute in Ivrea as a basic tool for students who had no prior knowledge of electronics or programming. It evolved to suit new demands and obstacles after gaining popularity. Arduino boards are microcontrollers, not full-fledged computers with their own operating system like the Raspberry Pi. They just execute C/C++ programs stored in their firmware. The Arduino Integrated Development Environment (IDE) is free software that allows you to program an Arduino board and upload your code to it.

The Raspberry Pi is a small, low-cost computer the size of a debit card that connects to a monitor or television and utilizes a conventional keyboard and mouse. It has a dedicated processor, memory, and a graphics driver, just like a PC. It also comes with its operating system, Raspberry Pi OS, a modified version of Linux. Although the Raspberry Pi lacks storage, you can use microSD cards to store whichever operating system you choose (Raspberry Pi, Ubuntu Mate, etc.). Because the Raspberry Pi has

Bluetooth, ethernet, and Wi-Fi connectivity, it may be used to transfer files over the internet. The software and the design of the Raspberry Pi project are not open source.

Arduino vs. Raspberry Pi
License

Arduino is an open-source electronics platform that uses simple hardware and software to make it easy. It's for everyone who want to do interactive projects. Arduino perceives the environment by receiving data from various sensors and controls lights, motors, and other actuators to influence its surroundings.

Raspberry Pi's hardware and software are both proprietaries. The Raspberry Pi has never claimed to be an open source. Many aspects of it, particularly the software, are open source, but not all of it is. The Raspberry Pi has been criticized for having closed-source components.

Architecture

The Harvard architecture is used by Arduino's CPU, which means that the program code and program data have distinct memory. It comprises two types of memories: program memory and data memory. The data is saved in the memory, while the code is kept in the flash program memory.

Raspberry Pi contains a single-core ARMv6 processor running at 700 MHz, a VideoCore IV GPU, and 512MB RAM. Its operating system and data are stored on an SD card. Raspbian, a lightweight Linux OS based on Debian, is officially supported by the Raspberry Pi.

Use cases.

Arduino

Weighing Machines, Traffic Light Countdown Timer, Parking Lot Counter, Embedded systems, Home Automation, Industrial Automation, Medical Instrument, Emergency Light for Railways.

Raspberry Pi

Desktop PC, Wireless Usage, Game Servers, Retro Gaming Machine, Robot Controller, Stop Motion Camera, Time-Lapse Camera. (Simpli Learn, 2023)

Seymore Cray’s contribution to computer science.

Seymour R. Cray had a profound effect on NSA’s mission from the 1950s to the 1990s. His work was crucial to NSA in technical aspects of the Cold War. Mr. Cray’s designs in the 1950s significantly advanced NSA’s early computing abilities. Adopting his inventions, NSA fielded a widespread system that for the first time allowed analysts remote access to signals intelligence information (SIGINT). Over the next four decades, Seymour Cray designed the world's fastest, most powerful supercomputers. NSA acquired two that made innovative use of transistors and proved critical to signals intelligence (SIGINT) processing in the Vietnam War. In 1978, Mr. Cray adapted the software for his Cray-1 to NSA-specific tasks. This computing power made it possible for NSA to tackle previously intractable analytic challenges. In 1985, the Cray-2 provided NSA massive memory capabilities not seen before. (NSA Open Source, 2018)

Some of the world's most challenging algorithms can be worked out in mere moments by supercomputers, a type of computing technology which goes well beyond the processing power found in a typical desktop PC. A typical desktop PC has only has one processor even when that processor is a multi-core option capable of running more than one task, and simply cannot touch the computing power of a supercomputer.

Supercomputers today are constructed with tens of thousands of central processing units (CPUs) installed in parallel and in communication with thousands of gigabytes worth of random-access memory (RAM), carrying an incredible amount of processing power capable of creating real-time weather models, decrypting data transferred over secure channels or performing genetic analysis. It seems almost like they have been around forever, and for many of us they have been around forever if by forever you mean for our entire life. But computers in general, and supercomputers specifically, have only been around for a couple generations.

If you are a computer aficionado, you undoubtedly know the name Seymour Cray. Cray became practically synonymous with supercomputing for decades, in fact Cray is widely known as the father of supercomputing.

Today we take a moment to remember Cray on an important patent anniversary. It was on December 5, 1978 that the United States Patent and Trademark Office issued U.S. Patent No. 4,128,880, entitled Computer Register Vector Processing, which listed Cray as the sole inventor. It was the '880 patent for which Cray was posthumously inducted into the National Inventors Hall of Fame in 1997.

Supercomputers

Supercomputers are designed to be so fast that their processing speed is measured on a different order of magnitude entirely than PCs. Processing speed, also known as clock speed, typically tops out at the order of gigahertz, a measure of oscillation indicating a billion pulses per second. A CPU can perform about one instruction per pulse. Instead of hertz, however, processing speed in supercomputers is measured in terms of floating-point operations per second, or flops. As of June 2015, the world's fastest supercomputer was the Tianhe-2, a Chinese supercomputer developed at the National University of Defense Technology. The Tianhe-2 has 3.12 million processing cores and operates at a processing speed of 33.86 petaflops, allowing it to conduct quadrillions of calculations every second.

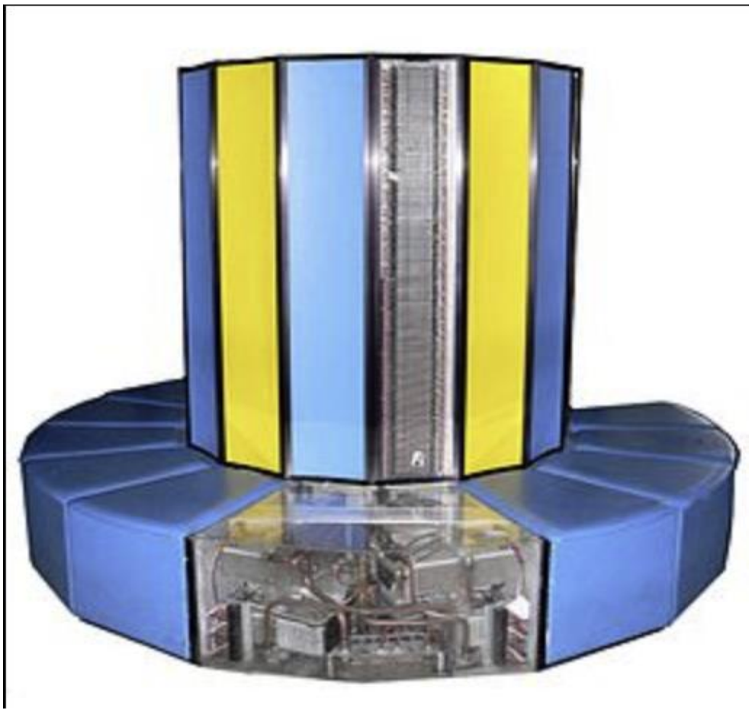


Fig - Cray-XMP48 Supercomputer at the EPFL" by Rama. Licensed under CC BY-SA 2.0.

Three of the supercomputers in that top 10 list of the most powerful commercially available computers maintained by TOP500, including the second-most powerful supercomputer at the Oak Ridge National Laboratory in Oak Ridge, TN, bear the name of the man known as the father of the supercomputer: Seymour Cray. This December 5th marks the 37th anniversary of the issuance of an important patent in the field of vector processing, a major step forward in supercomputing technology.

Developers of supercomputers have been in pursuit of an infinitely fast clock which can execute a single instruction stream program on a single memory which is both infinitely large and incredibly fast. The creation of supercomputers goes back in large part to the development of the FORTRAN computing language by American computer scientist John Backus, who was working at the time for IBM. FORTRAN became the world's first high-level programming language that could be widely used among various types of computer architectures, as opposed to the assembly languages of previous years, which was always specific to the architecture on which it was intended to run.

Although the Cray-1 was revolutionary in the supercomputing world for many reasons, it was by no means the first computer developed to handle incredibly complex scientific models. By the early 1960s, supercomputers had been developed by IBM (the Stretch), England's University of Manchester in collaboration with Ferranti Ltd. (the Atlas) as well as Remington Rand (the Univac LARC). These and later models would prove to be incredibly expensive, thus relying a great deal on government funding and a handful of worldwide scientific and engineering institutions.

The name of Seymour Cray, however, would go on to take the leading position from these rivals and become synonymous with the entire industry of supercomputing for decades. From a young age, Cray, a native of Chippewa Falls, WI, always showed an interest in electrical engineering from a young age and when he joined the U.S. Army and arrived in Europe the day after D-Day, he served in an infantry communications platoon. After returning home and earning a bachelor’s in electrical engineering and a master’s in applied mathematics, Cray would begin working on computer development as an employee with Engineering Research Associates, based in St. Paul, MN.

It wouldn’t be until 1957, when Cray helped with the founding of the Control Data Corporation (CDC), that his developments in supercomputing would begin to truly take flight. In October 1959, his company released the CDC 1604, the world’s fastest computer at the time and the first fully transistorized computer that worked without vacuum tubes. Cray was the primary engineer on the CDC 1604. A few years later in 1964, Cray outdid himself with his design of the CDC 6600, a machine with the world’s fastest clock time of 100 nanoseconds. It sold for around $7 million when it was first released and is cited by many as the world’s first true supercomputer. Other firsts established by the CDC 6600 include the use of Freon as a cooling agent to keep electronic equipment from overheating and to reduce data transmission times further; the CDC 6600 was also the first computer to utilize a cathode ray tube monitor, establishing the screen display which is now a major component for all human-computer interactions.

Although Cray found engineering success at CDC, he would clash at times with management and eventually requested that a CDC engineering facility be built in his hometown of Chippewa Falls, where he could work without the growing corporate distractions. The next major steps towards more powerful supercomputing technology would be undertaken by Cray at a new business venture that he founded, Cray Research. It was at his own company that he would develop the Cray-1, a marvel of computer engineering and aesthetic design that greatly outdid his previous designs for CDC. This model replaced transistors with integrated circuits and took advantage of the innovative concept of vector processing.

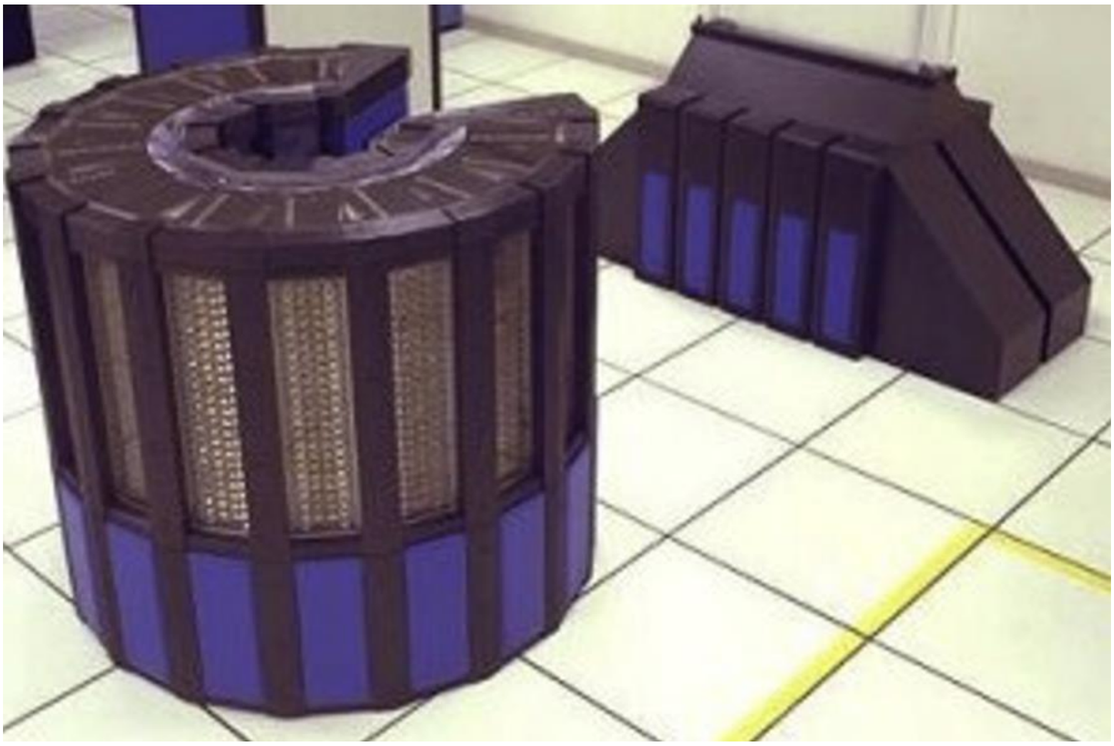
Vector Processing

The ‘880 patent claimed a vector processing apparatus for a computer having a main memory, the apparatus having a plurality of vector registers capable of holding a plurality of elements in an ordered set of data, an arithmetic or logical function unit with a segmented means for holding data for incomplete operations and having an input for receiving operands and an output for delivering results, a path select means associated with the vector registers and responsive to program instructions for selectively connecting individual vector registers for data transmission, and then a control means associated with the vector registers and operative to transmit elements of the ordered set of data from a vector register to a functional unit on successive clock periods. The innovation addressed shortcomings with conventional vector processing machines of the day, enabling the efficient processing of short, moderate, and long length vectors.

Vector processing made the Cray-1 supercomputer, which Cray said he designed himself without the use of computer-aided design software but rather with #3 pencils on a quadrille pad, the gold standard of its industry for decades. In vector processing, computers implement an instruction set that operated on one-dimensional data arrays, greatly improving workload performance over the scalar processing techniques used at that time. The use of integrated circuits enabled the Cray 1 to perform 64-bit processing, reaching performance speeds of 136 megaflops. This was, by contrast, much greater than the 3-megaflop performance produced by the CDC 6600.

Commercial Success

The Cray-1 wasn't simply an ultra-powerful piece of computing equipment; it was also a model of elegant design in computer development where appealing design elements actually served functional purposes. An overhead view of the supercomputer shows that the entire computer is shaped like the letter C, which simply sounds like a smart piece of branding until it's realized that the curved shape allowed speed-dependent modules to be installed closer to the middle with shorter wire lengths, enabling the entire unit to achieve 80 megahertz of processing speed. A cushioned bench surrounding the computer earned the machine, which retailed for $8.8 million when released in 1977, the nickname of "the world's most expensive loveseat." Under the cushions rested the supercomputer's power supply and air conditioning units and the cushions served the purpose of making maintenance technicians more comfortable when servicing the machine.



Cray-2 Supercomputer" by NASA. Public domain.

After the commercial success of the Cray-1 supercomputer, Seymour Cray continued to pursue improvements in the field over the course of the next two decades. In 1985, Cray Research released the Cray-2 supercomputer which utilized up to 8 CPUs and utilized logic devices packed so tightly that the specially formulated electronic cooling liquid Fluorinert was required to keep the components from overheating. The Cray-2 offered calculation speeds of up to 1.9 gigaflops, making it the first supercomputer to break the gigaflop threshold. Seymour would also pursue the development of a Cray-2 successor that would be known as the Cray-3, but it did not achieve the commercial success of earlier Cray models.

Legacy

Seymour Cray would continue his pursuit of ultra-high speed supercomputing technologies until his death in October 1996. Tragically, Cray lost his life as a result of injuries sustained in a car crash. Cray's legacy in the computing world, however, continues to be legendary, as is evidenced by the fact that so many Cray designs still top the list of the world's most powerful supercomputers. In 1997, the Institute of Electrical and Electronic Engineers (IEEE) established the Seymour Cray Computer Engineering Award to recognize innovative contributions to high-performance computing which reflects the creative spirit that made Cray a mastermind of the supercomputing world. (Brachmann, 2015)

Architectures and uses of Mainframe computers were:

An architecture is a set of defined terms and rules that are used as instructions to build products. Each generation of mainframe computers has included improvements in architecture, while remaining the most stable, secure, and compatible of all computing platforms. In computer science, an architecture describes the organizational structure of a system. An architecture can be recursively decomposed into parts that interact through interfaces, relationships that connect parts, and constraints for assembling parts. Parts that interact through interfaces include classes, components, and subsystems. Starting with the first large machines, which arrived on the scene in the 1960s and became known as "Big Iron" (in contrast to smaller departmental systems), each new generation of mainframe computers has included improvements in one or more of the following areas of the architecture: More and faster processors More physical memory and greater memory addressing capability Dynamic capabilities for upgrading both hardware and software Increased automation of hardware error checking and recovery Enhanced devices for input/output (I/O) and more and faster paths (channels) between I/O devices and processors More sophisticated I/O attachments, such as LAN adapters with extensive inboard processing A greater ability to divide the resources of one machine into multiple, logically independent and isolated systems, each running its own operating system Advanced clustering technologies, such as Parallel Sysplex®, and the ability to share data among multiple systems. Despite the continual change, mainframe computers remain the most stable, secure, and compatible of all computing platforms. The latest models can handle the most advanced and demanding customer workloads yet continue to run applications that were written in the 1970s or earlier. How can a technology change so much, yet remain so stable? It can by evolving to meet new challenges. In the early 1990s, the client/server model of computing, with its distributed nodes of less powerful computers, emerged to challenge the dominance of mainframe computers. Industry pundits predicted a swift end for the mainframe computer and called it a "dinosaur." In response, mainframe designers did what they have always done when confronted with changing times and a growing list of user requirements: They designed new mainframe computers to meet the demand. With a tip of the hat to the dinosaur naysayers, IBM®, as the leading manufacturer of mainframe computers, code-named its then-current machine T-Rex. With the expanded functions and added tiers of data processing capabilities such as Web-serving, autonomics, disaster recovery, and grid computing, the mainframe computer is poised to ride the next wave of growth in the IT industry. Mainframe manufacturers such as IBM are once again reporting annual sales growth in the double digits. And the evolution continues. While the mainframe computer has retained its traditional, central role in the IT organization, that role is now defined to include being the primary hub in the largest distributed networks. In fact, the Internet itself is based largely on numerous, interconnected mainframe computers serving as major hubs and routers. As the image of the mainframe computer continues to evolve, you might ask: Is the mainframe computer a self-contained computing environment, or is it one part of the puzzle in distributed computing? The answer is that The New Mainframe is both— a self-contained processing centre, powerful enough to process the largest and most diverse workloads in one secure "footprint", and one that is just as effective when implemented as the primary server in a corporation's distributed server farm. In effect, the mainframe computer is the definitive server in the client/server model of computing.

Businesses today rely on the mainframe to: Perform large-scale transaction processing (thousands of transactions per second) Support thousands of users and application programs concurrently accessing numerous resources Manage terabytes of information in databases Handle large-bandwidth communication The roads of the information superhighway often led to a mainframe. Mainframe strengths: Reliability, availability, and serviceability the reliability, availability, and serviceability (or "RAS") of a computer system have always been important factors in data processing. When we say that a particular computer system "exhibits RAS characteristics," we mean that its design places a high priority on the system remaining in service at all times. Ideally, RAS is a central design feature of all aspects of a computer system, including the applications. Mainframe strengths: Security One of a firm's most valuable resources is its data: Customer lists, accounting data, employee information, and so on. This critical data needs to be securely managed and controlled, and, simultaneously, made available to those users authorized to see it. The mainframe computer has extensive capabilities to simultaneously share, but still protect, the firm's data among multiple users. Mainframe strengths: Scalability It has been said that the only constant is change. Nowhere is that statement more true than in the IT industry. In business, positive results can often trigger a growth in IT infrastructure to cope with increased demand. The degree to which the IT organization can add capacity without disruption to normal business processes or without incurring excessive overhead (non-productive processing) is largely determined by the scalability of the computing platform. Mainframe strength: Continuing compatibility Mainframe customers tend to have a very large financial investment in their applications and data. Some applications have been developed and refined over decades. Some applications were written many years ago, while others may have been written "yesterday." The ability of an application to work in the system or its ability to work with other devices or programs is called compatibility. (IBM, 2010)

Patch management

We know the reasons patching regularly is a dreaded activity: It’s difficult to keep up with the number of patches required across systems, there is a risk that untested patches can take down business-critical systems, and the time spent manually scanning, testing, and applying patches takes resources away from other pressing IT security needs. This explains why Fortinet1 recently reported that 60% of the companies they protect experienced attacks trying to exploit vulnerabilities that were more than 10 years old. So, what are the consequences if I don’t patch regularly? Security vulnerabilities will be exploited Some companies take a calculated risk in not patching, assuming that their firewalls or antivirus technology will catch major threats before they cause too much harm. However, as malware gets more sophisticated, firewalls are antivirus are less successful at detecting a breach, and the cost of an attack has gone up in the US in recent years. Out-of-date patches account for 50%2 of all information system vulnerabilities, and the quick spread of ransomware WannaCry in May 2017 exposed the danger of those who had not patched. The majority of those affected by WannaCry were running Windows XP, which had been unsupported for several years, or Windows 10, which had released a patch of the exploited vulnerability two months prior to the ransomware spread. The HeartBleed OpenSSL vulnerability was also exploited once publicized and led to the breach of 4.5 million patient records a week after the fix had been released. Once a vulnerability has been publicized, malware is quickly developed to take advantage of it. With thousands of vulnerable endpoints at risk, hackers know that it takes time for companies to download, test, and implement a new patch. In the above cases, not patching regularly led to lost money to ransom seekers, lost business while systems were impacted, potential lawsuits, and ongoing damage to company reputation. The cost to recover is high the cost of delaying patching falls into two categories: Cost to recover from an attack of an unpatched system, and cost to catch up with patching if it has been ignored. The costs of an attack vary widely based on the company, size, and attack type. Large data breaches from Target and Yahoo cost $202 million and $350 million respectively, and loss of customer trust can go on for years. In addition to the lost business and settlements that are possible after an attack, there are also huge costs for an IT team to stop an attack and put new procedures in place to prevent a future occurrence. With an average of 15 new patches released each day3, failure to apply updates on a regular schedule can quickly snowball and result in legacy systems running outdated and unsupported software. In the best-case scenario, not patching regularly will result in an extended fire drill for most of the IT team, as soon as a compliance audit is due, or the CEO decides to check on the status of data security. In the worst-case scenario, you’ve fallen victim to an attack and may be looking for a job tomorrow. (AUTOMOX, 2022)

Automated patch management:

Key Benefits of Automating Patch Management over 10 years ago 3 minute read time Patch management has become so critical in the IT landscape that it’s now a must-have security solution in enterprises. When you utilize effective patch management, you’ll be able to eliminate application vulnerabilities and the security threats that arise from them. Today, IT admins are looking for ways to make software patching more efficient and error free. This can be achieved by implementing centralized and

automated solutions. Let's look at some of the key benefits of automating patch management.
#1 Enhance Endpoint Security Automating patch management allows you to be more efficient because it allows you to instantly and uniformly patch all the systems in your IT infrastructure. With this approach, you'll be able to schedule timely patch deployments to multiple systems at the same time and keep them protected against critical vulnerabilities. You can schedule automated patch deployment to all your vulnerable systems as soon as a security update, bug fix, or a newer application version is released. In addition, patch management automation allows you to: Propagate software updates to target systems uniformly Update vulnerable systems with security updates Timely updates of security patches will reduce the risk of exposing systems to attacks like security breaches, and data loss. (Solarwinds IT Community, 2010)

Bibliography

AUTOMOX. (2022, August 19). *What is Patch Management, Really?* Retrieved from AUTOMOX: https://www.automox.com/blog/definition-patch-management

Brachmann, G. Q. (2015, December 05). *Seymour Cray, the father of supercomputers and vector processing*. Retrieved from IP Watchdog: https://ipwatchdog.com/2015/12/05/seymour-cray-father-of-supercomputers/id=63515/

IBM. (2010). *Mainframe concepts*. Retrieved from IBM: https://www.ibm.com/docs/en/zos-basic-skills?topic=today-mainframe-architecture-secure-compatible-still-evolving

NSA Open Source. (2018). *NSA Historical Figures*. Retrieved from National Security Agency/Central Security Service: https://www.nsa.gov/History/Cryptologic-History/Historical-Figures/Historical-Figures-View/Article/1711612/seymour-r-cray/#:~:text=Over%20the%20next%20four%20decades,1%20to%20NSA%2Dspecific%20tasks

Simpli Learn. (2023, March 10). *Arduino Vs. Raspberry Pi: Which Is the Better Board?* Retrieved from SimpliLearn: https://www.simplilearn.com/tutorials/programming-tutorial/arduino-vs-raspberry-pi

Solarwinds IT Community. (2010). *https://www.automox.com/blog/definition-patch-management*. Retrieved from THWACK: https://thwack.solarwinds.com/resources/b/geek-speak/posts/key-benefits-of-automating-patch-management#:~:text=You%20can%20schedule%20automated%20patch,vulnerable%20systems%20with%20security%20updates