DOWNLOAD PDF INTRODUCTION, BY D. HOST.

Chapter 1 : HowTos/Virtualization/Introduction - CentOS Wiki

An Introduction of D Host "The whole purpose of the D Host is to create a memorable stay that is completely customized" It's the little things that count, goes the saying.

When the store-and-forward switching method is used, the switch receives the complete frame before forwarding it on to the destination. The cyclic redundancy check CRC part of the trailer is used to determine if the frame has been modified during transit. Two types of cut-through switching methods are fast-forward and fragment-free. Auto-MDIX is a feature that is enabled on the latest Cisco switches and that allows the switch to detect and use whatever type of cable is attached to a specific port. When a device is sending data to another device on a remote network, the Ethernet frame is sent to the MAC address of the default gateway. A MAC address is only useful on the local Ethernet network. When data is destined for a remote network of any type, the data is sent to the default gateway device, the Layer 3 device that routes for the local network. The ARP table in a switch maps which two types of address together? Layer 3 address to a Layer 2 address Layer 3 address to a Layer 4 address Layer 4 address to a Layer 2 address Layer 2 address to a Layer 4 address Explanation: These mappings can be learned by the switch dynamically through ARP or statically through manual configuration. Refer to the exhibit. In this scenario, what will happen next? CCNA1 Chapter 5 v5. When a network device wants to communicate with another device on the same network, it sends a broadcast ARP request. In this case, the request will contain the IP address of PC2. A switch with a default configuration connects four hosts. The ARP table for host A is shown. What happens when host A wants to send an IP packet to host D? Host A sends out the packet to the switch. The switch sends the packet only to the host D, which in turn responds. Host A sends out a broadcast of FF: Every other host connected to the switch receives the broadcast and host D responds with its MAC address. All devices on the same network receive this broadcast. Host D will respond to this broadcast. The switches are in their default configuration. Which network hosts will receive the ARP request sent by host A? The ARP broadcast would be sent to every device on the local network. Hosts B, C, and router R1 would receive the broadcast. Router R1 would not forward the message. Which statement describes the treatment of ARP requests on the local link? They must be forwarded by all routers on the local network. They are received and processed by every device on the local network. They are dropped by all switches on the local network. They are received and processed only by the target device. One of the negative issues with ARP requests is that they are sent as a broadcast. This means all devices on the local link must receive and process the request. What are two potential network problems that can result from ARP operation? On large networks with low bandwidth, multiple ARP broadcasts could cause data communication delays. Large numbers of ARP request broadcasts could cause the host MAC address table to overflow and prevent the host from communicating on the network. Large numbers of ARP broadcast messages could cause momentary data communications delays. ARP table overflows are very unlikely. Multiple ARP replies resulting in the switch MAC address table containing entries that match the MAC addresses of connected nodes and are associated with the relevant switch port are required for normal switch frame forwarding operations. It is not an ARP caused network problem. Fill in the blank. Match the characteristic to the forwarding method. Not all options are used. A store-and-forward switch always stores the entire frame before forwarding, and checks its CRC and frame length. A cut-through switch can forward frames before receiving the destination address field, thus presenting less latency than a store-and-forward switch. Because the frame can begin to be forwarded before it is completely received, the switch may transmit a corrupt or runt frame. All forwarding methods require a Layer 2 switch to forward broadcast frames. Recommend From year to year, Cisco has updated many versions with difference questions. The latest version is version 6. What is your version? It depends on your instructor creating your class. We recommend you to go thought all version if you are not clear. While you take online test with netacad. Each version have 1 to 10 different questions or more. After you review all questions, You should practice with our online test system by go to "Online Test" link below.

Chapter 2 : Introduce a guest speaker sample template - Writing Samples and Tips

The Generic Host will eventually replace the Web Host. For hosting calendrierdelascience.com Core web apps, developers should use the Web Host based on IWebHostBuilder. For hosting non-web apps, developers should use the Generic Host based on HostBuilder.

In this guide, we will discuss some fundamental DNS concepts that will help you hit the ground running with your DNS configuration. After tackling this guide, you should be ready to set up your domain name with DigitalOcean or set up your very own DNS server. Domain Terminology We should start by defining our terms. Domain Name System The domain name system, more commonly known as "DNS" is the networking system in place that allows us to resolve human-friendly names to unique addresses. Domain Name A domain name is the human-friendly name that we are used to associating with an internet resource. Some people will say that the "google" portion is the domain, but we can generally refer to the combined form as the domain name. The domain name system allows us to reach the Google servers when we type "google. Each IP address must be unique within its network. When we are talking about websites, this network is the entire internet. IPv4, the most common form of addresses, are written as four sets of numbers, each set having up to three digits, with each set separated by a dot. With DNS, we map a name to that address so that you do not have to remember a complicated set of numbers for each place you wish to visit on a network. The top-level domain is the furthest portion to the right as separated by a dot. Common top-level domains are "com", "net", "org", "gov", "edu", and "io". Top-level domains are at the top of the hierarchy in terms of domain names. These parties can then distribute domain names under the TLD, usually through a domain registrar. Hosts Within a domain, the domain owner can define individual hosts, which refer to separate computers or services accessible through a domain. For instance, most domain owners make their web servers accessible through the bare domain example. You can have other host definitions under the general domain. You could have API access through an "api" host api. The host names can be arbitrary as long as they are unique for the domain. SubDomain A subject related to hosts are subdomains. DNS works in a hierarchy. TLDs can have many domains under them. For instance, the "com" TLD has both "google. A "subdomain" refers to any domain that is part of a larger domain. In this case, "ubuntu. This is typically just called the domain or the "ubuntu" portion is called a SLD, which means second level domain. Likewise, each domain can control "subdomains" that are located under it. This is usually what we mean by subdomains. For instance you could have a subdomain for the history department of your school at " www. The "history" portion is a subdomain. The difference between a host name and a subdomain is that a host defines a computer or resource, while a subdomain extends the parent domain. It is a method of subdividing the domain itself. Whether talking about subdomains or hosts, you can begin to see that the left-most portions of a domain are the most specific. This is how DNS works: Domains in the DNS system can be given relative to one another, and as such, can be somewhat ambiguous. A FQDN is an absolute name that specifies its location in relation to the absolute root of the domain name system. This means that it specifies each parent domain including the TLD. An example of a FQDN is "mail. Name Server A name server is a computer designated to translate domain names into IP addresses. These servers do most of the work in the DNS system. Since the total number of domain translations is too much for any one server, each server may redirect request to other name servers or delegate responsibility for a subset of subdomains they are responsible for. Name servers can be "authoritative", meaning that they give answers to queries about domains under their control. Zone File A zone file is a simple text file that contains the mappings between domain names and IP addresses. This is how the DNS system finally finds out which IP address should be contacted when a user requests a certain domain name. Zone files reside in name servers and generally define the resources available under a specific domain, or the place that one can go to get that information. Records Within a zone file, records are kept. In its simplest form, a record is basically a single mapping between a resource and a name. These can map a domain name to an IP address, define the name servers for the domain, define the mail servers for the domain, etc. The system is very simple at a high-level overview, but is very complex as you look at the details. Overall though, it is a very reliable infrastructure that has been essential to the adoption of the internet as we know it today. At the top of this system is what are known as "root servers". There are currently 13 root servers in operation. However, as there are an incredible number of names to resolve every minute, each of these servers is actually mirrored. The interesting thing about this set up is that each of the mirrors for a single root server share the same IP address. When requests are made for a certain root server, the request will be routed to the nearest mirror of that root server. What do these root servers do? Root servers handle requests for information about Top-level domains. So if a request comes in for something a lower-level name server cannot resolve, a query is made to the root server for the domain. They will, however, be able to direct the requester to the name servers that handle the specifically requested top-level domain. So if a request for " www. It will check its zone files for a listing that matches " www. It will not find one. It will instead find a record for the "org" TLD and give the requesting entity the address of the name server responsible for "org" addresses. TLD Servers The requester then sends a new request to the IP address given to it by the root server that is responsible for the top-level domain of the request. So, to continue our example, it would send a request to the name server responsible for knowing about "org" domains to see if it knows where " www. Once again, the requester will look for " www. It will not find this record in its files. However, it will find a record listing the IP address of the name server responsible for "wikipedia. This is getting much closer to the answer we want. Domain-Level Name Servers At this point, the requester has the IP address of the name server that is responsible for knowing the actual IP address of the resource. It sends a new request to the name server asking, once again, if it can resolve " www. The name server checks its zone files and it finds that it has a zone file associated with "wikipedia. Inside of this file, there is a record for the "www" host. This record tells the IP address where this host is located. The name server returns the final answer to the requester. What is a Resolving Name Server? In the above scenario, we referred to a "requester". What is the requester in this situation? In almost all cases, the requester will be what we call a "resolving name server" A resolving name server is one configured to ask other servers questions. Basically, a user will usually have a few resolving name servers configured on their computer system. The resolving name servers are usually provided by an ISP or other organizations. For instance Google provides resolving DNS servers that you can query. These can be either configured in your computer automatically or manually. When you type a URL in the address bar of your browser, your computer first looks to see if it can find out locally where the resource is located. It checks the "hosts" file on the computer and a few other locations. It then sends the request to the resolving name server and waits back to receive the IP address of the resource. The resolving name server then checks its cache for the answer. Resolving name servers basically compress the requesting process for the end user. The clients simply have to know to ask the resolving name servers where a resource is located and be confident that they will investigate and return the final answer. Zone Files We mentioned in the above process the idea of "zone files" and "records". Zone files are the way that name servers store information about the domains they know about. Every domain that a name server knows about is stored in a zone file. Most requests coming to the average name server are not something that the server will have zone files for. If it is configured to handle recursive queries, like a resolving name server, it will find out the answer and return it. Otherwise, it will tell the requesting party where to look next. The more zone files that a name server has, the more requests it will be able to answer authoritatively. It generally is used to configure just a single domain. It can contain a number of records which define where resources are for the domain in question.

Chapter 3 : Introduction to Host Utilities

The physical computer on which you install Workstation Pro is called the host system and its operating system is called the host operating system. To run Workstation Pro, the host system and the host operating system must meet specific hardware and software requirements.

It includes definitions of all sites, applications, virtual directories and application pools, as well as global defaults for the web server settings similar to machine. It is also special in that it is the only IIS configuration file available when the web server is installed however, users can still add web. It has definitions for locking-down most IIS sections to the global level, so that by default they cannot be overridden by lower-level web. This document walks through all the sections, in the order they appear in the file, and explains them one by one. The most complex section is system. This document specifies the content of each configuration section, as appears in applicationHost. By design, many of the sections are empty or not complete only some of their content appears in the XML. The rest of the values are taken from the schema defaults. This is done to avoid too much information and cluttering of the file, and in order to keep it reasonably readable. For full schema reference, including default values for all properties in every section, their valid ranges, etc. For convenience, chunks of these files are included in this document in the appropriate sections so the reader can understand which properties are available, what the default values are, etc. See the additional note below about how to read schema information. Make a backup of the file before making any changes to it. How to Read Config Schema As noted above, this document contains snippets of schema information for each section, so the reader can discover what properties are available and what their default values and valid ranges are. The snippets are taken directly from the configuration schema file for IIS settings: This section explains how to read schema information. The schema for each configuration section is defined in a XML element. There is no schema definition for section groups. The following format is used here to explain how to read the schema: Every attribute must have a name. Not all attributes have default values for example, site name. In this case, the syntax will be "". The runtime type of the attribute. This is one of "bool", "enum", "flags", "int", "int64", "String", "timeSpan". Every attribute must have a type. Every such value has a numerical value and a friendly name. The syntax is using the character " " as a delimiter between the friendly names: Therefore the numerical values should be in multiples of 2, so they can be ORed together to form combinations. The syntax is identical to "enum": It can be persisted as a number representing seconds, or minutes ; or as a formatted string in the form of "[dd: The other elements represent numbers of hours, minutes and seconds, respectively. The "timeSpanFormat" attribute specifies which format should be used: Required attributes are marked "Required". It means that a value for them must be set in the XML. For example, site name is a required attribute every site must have a name in IIS 7. All other schema information is specified within it. It has one attribute directly in it "name" , and then the rest of the schema is in sub-elements within it: An attribute schema must specify a name and a runtime type for the attribute. It may mark the attribute as required. It may mark the attribute as the unique key if inside a collection , or as part of a collection key together with other attributes. It may specify a default value for the attribute. It may mark the attribute for automatic encryption on-disk. It may specify if the word "Infinite" is allowed as a value for the attribute only for numeric types such as int and in64, and for timeSpan. It may specify the timespan format seconds, minutes or formatted string for timespan attributes. It may specify validation rules for the attributes see Attribute Validation section below in this document. Elements can be nested. An element is simply a container for other attributes, or sub-elements. Collections contain multiple elements, which can be added and removed from them individually. Typically the collection directive names are "add", "remove" and "clear", but some collections use different names for clarity for example, the collection is using "site" instead of "add". This is done by specifying values for addElement, removeElement and clearElement in the collection schema. If a collection directive is missing from the schema, the collection will not support it. The collection schema may specify the name of a default element that will be used as a container of default values for collection elements this complements isCollectionDefault in the element schema. For example, the collection is using siteDefaults as the default element. For example, consider two levels of configuration:

DOWNLOAD PDF INTRODUCTION, BY D. HOST.

Chapter 4 : Introduction to Fungi

Introduction to Programmable Host On-Demand. The Programmable Host On-Demand API is a set of Java APIs that allow developers to integrate various pieces of the Host On-Demand client code, such as terminals, menus, and toolbars, into their own custom Java applications and applets.

This page provides a basic introduction to these technologies, as well as some additional concepts. To understand these virtualization technologies, you have to be aware that there are some different approaches to virtualization. For the discussion of KVM and Xen, two approaches are relevant: Normally, this means that various hardware devices are emulated. Normally, such virtualization platform attempts to run as many instructions on the native CPU which is a lot faster than CPU emulation , catching and handling privileged instructions appropriately. Some virtualization platforms support or require CPU extensions to assist virtualization. This is usually called hardware-assisted virtualization. In most situations paravirtualization is preferable over full-virtualization, because it is faster. Xen supports paravirtualization and hardware-assisted full-virtualization. Both paravirtualized virtual machines PVMs and hardware-assisted fully virtualized machines HVMs can run at the same time. Xen requires a hypervisor that booted before any Linux kernel. The hypervisor then boots a CentOS that is named the administrative or privileged domain, meaning that it is used to launch additional virtual machines and has defacto access to many real hardware devices. This administrative is usually called dom0. Additional unprivileged domains can be started via dom0. These unprivileged domains are usually called domUs. Both dom0 and paravirtualized domUs have to run the CentOS Xen kernel, rather than the default kernel. KVM currently only supports hardware-assisted full-virtualization although paravirtualization support is being worked on. KVM is a kernel module, and works with the default non-Xen kernel. CentOS 5 also includes comfortable management tools like virt-install and virt-manager. For detailed information, please see the Virtualization manual html and pdf. There is a KVM howto as well.

Chapter 5 : Introduction to the Viruses

as a foreign antigen, and the immune system attacks host tissues. D. Treatment - Penicillin G effective at all stages of disease. A single injection can eradicate T.

Docker overview Estimated reading time: Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. The Docker platform Docker provides the ability to package and run an application in a loosely isolated environment called a container. The isolation and security allow you to run many containers simultaneously on a given host. This means you can run more containers on a given hardware combination than if you were using virtual machines. You can even run Docker containers within host machines that are actually virtual machines! Docker provides tooling and a platform to manage the lifecycle of your containers: Develop your application and its supporting components using containers. The container becomes the unit for distributing and testing your application. This works the same whether your production environment is a local data center, a cloud provider, or a hybrid of the two. Docker Engine Docker Engine is a client-server application with these major components: A server which is a type of long-running program called a daemon process the dockerd command. A command line interface CLI client the docker command. The daemon creates and manages Docker objects, such as images, containers, networks, and volumes. Docker is licensed under the open source Apache 2. For more details, see Docker Architecture below. What can I use Docker for? Fast, consistent delivery of your applications Docker streamlines the development lifecycle by allowing developers to work in standardized environments using local containers which provide your applications and services. Consider the following example scenario: Your developers write code locally and share their work with their colleagues using Docker containers. They use Docker to push their applications into a test environment and execute automated and manual tests. When developers find bugs, they can fix them in the development environment and redeploy them to the test environment for testing and validation. When testing is complete, getting the fix to the customer is as simple as pushing the updated image to the production environment. Running more workloads on the same hardware Docker is lightweight and fast. It provides a viable, cost-effective alternative to hypervisor-based virtual machines, so you can use more of your compute capacity to achieve your business goals. Docker is perfect for high density environments and for small and medium deployments where you need to do more with fewer resources. Docker architecture Docker uses a client-server architecture. The Docker client talks to the Docker daemon, which does the heavy lifting of building, running, and distributing your Docker containers. The Docker client and daemon can run on the same system, or you can connect a Docker client to a remote Docker daemon. The Docker daemon The Docker daemon dockerd listens for Docker API requests and manages Docker objects such as images, containers, networks, and volumes. A daemon can also communicate with other daemons to manage Docker services. The Docker client The Docker client docker is the primary way that many Docker users interact with Docker. When you use commands such as docker run, the client sends these commands to dockerd, which carries them out. The docker command uses the Docker API. The Docker client can communicate with more than one daemon. Docker registries A Docker registry stores Docker images. Docker Hub and Docker Cloud are public registries that anyone can use, and Docker is configured to look for images on Docker Hub by default. You can even run your own private registry. When you use the docker pull or docker run commands, the required images are pulled from your configured registry. When you use the docker push command, your image is pushed to your configured registry. Docker store allows you to buy and sell Docker images or distribute them for free. For instance, you can buy a Docker image containing an application or service from a software vendor and use the image to deploy the application into your testing, staging, and production environments. You can upgrade the application by pulling the new version of the image and redeploying the containers. Docker objects When you use Docker, you are creating and using images, containers, networks, volumes, plugins, and other objects. This section is a brief overview of some of those objects. Images An image is a read-only template with instructions for creating a Docker container. Often, an image is based on another image, with some additional customization. For example, you may build an image which is based on the ubuntu image, but installs the Apache web server and your application, as well as the configuration details needed to make your application run. You might create your own images or you might only use those created by others and published in a registry. To build your own image, you create a Dockerfile with a simple syntax for defining the steps needed to create the image and run it. Each instruction in a Dockerfile creates a layer in the image. When you change the Dockerfile and rebuild the image, only those layers which have changed are rebuilt. This is part of what makes images so lightweight, small, and fast, when compared to other virtualization technologies. Containers A container is a runnable instance of an image. You can connect a container to one or more networks, attach storage to it, or even create a new image based on its current state. By default, a container is relatively well isolated from other containers and its host machine. A container is defined by its image as well as any configuration options you provide to it when you create or start it. When a container is removed, any changes to its state that are not stored in persistent storage disappear. If you do not have the ubuntu image locally, Docker pulls it from your configured registry, as though you had run docker pull ubuntu manually. Docker creates a new container, as though you had run a docker container create command manually. Docker allocates a read-write filesystem to the container, as its final layer. This allows a running container to create or modify files and directories in its local filesystem. Docker creates a network interface to connect the container to the default network, since you did not specify any networking options. This includes assigning an IP address to the container. Because the container is running interactively and attached to your terminal due to the -i and -t flags , you can provide input using your keyboard while the output is logged to your terminal. You can start it again or remove it. Services Services allow you to scale containers across multiple Docker daemons, which all work together as a swarm with multiple managers and workers. A service allows you to define the desired state, such as the number of replicas of the service that must be available at any given time. By default, the service is load-balanced across all worker nodes. To the consumer, the Docker service appears to be a single application. Docker Engine supports swarm mode in Docker 1. The underlying technology Docker is written in Go and takes advantage of several features of the Linux kernel to deliver its functionality. Namespaces Docker uses a technology called namespaces to provide the isolated workspace called the container. When you run a container, Docker creates a set of namespaces for that container. These namespaces provide a layer of isolation. Each aspect of a container runs in a separate namespace and its access is limited to that namespace. Docker Engine uses namespaces such as the following on Linux: Managing network interfaces NET: Managing filesystem mount points MNT: Isolating kernel and version identifiers. Control groups Docker Engine on Linux also relies on another technology called control groups cgroups. A cgroup limits an application to a specific set of resources. Control groups allow Docker Engine to share available hardware resources to containers and optionally enforce limits and constraints. For example, you can limit the memory available to a specific container. Union file systems Union file systems, or UnionFS, are file systems that operate by creating layers, making them very lightweight and fast. Docker Engine uses UnionFS to provide the building blocks for containers. Container format Docker Engine combines the namespaces, control groups, and UnionFS into a wrapper called a container format. The default container format is libcontainer. Next steps Get hands-on experience with the Getting started with Docker tutorial. Check out examples and deep dive topics in the Docker Engine user guide.

Chapter 6 : IntroductionToDBus

writing your talk show script Use the following as a guide for your script. Print the page and work directly on it OR write on a separate sheet and modify the wording and format as necessary.

The Plant Health Instructor. Little, and Carol M. A fungus is a eukaryote that digests food externally and absorbs nutrients directly through its cell walls. Most fungi reproduce by spores and have a body thallus composed of microscopic tubular cells called hyphae. Fungi are heterotrophs and, like animals, obtain their carbon and energy from other organisms. Some fungi obtain their nutrients from a living host plant or animal and are called biotrophs ; others obtain their nutrients from dead plants or animals and are called saprotrophs saprophytes, saprobes. Some fungi infect a living host, but kill host cells in order to obtain their nutrients; these are called necrotrophs. Fungi were once considered to be primitive members of the plant kingdom, just slightly more advanced than bacteria. We now know that fungi are not primitive at all. In fact, recent taxonomic treatments such as the Tree of Life Project show that fungi and animals both belong to the group Opisthokonta Fig. Fungi may not be our next of kin, but they are more closely related to animals than they are to plants. We also recognize that organisms traditionally studied as "fungi" belong to three very different unrelated groups: Recent studies have provided support for the recognition of additional phyla, such as Glomeromycota, a group of fungi once placed in Zygomycota that form an association with the roots of most plants Fig. A group of parasitic organisms called Microsporidia that live inside the cells of animals are also now considered to belong in the fungal kingdom Fig. This classification is used in the Dictionary of the Fungi Kirk et al. However, the classification system will undergo additional changes as scientists use new methods to study the fungi. For example, Jones et al. Figure 2 How old are fungi? Fungi are an ancient groupâ€”not as old as bacteria, which fossil evidence suggests may be 3. Mushrooms exquisitely preserved in amber from the Late Cretaceous 94 million years ago tell us that there were mushroom-forming fungi remarkably similar to those that exist today when dinosaurs were roaming the planet Hibbett et al. However, the fungal fossil record is incomplete and provides only a minimum time estimate for when different groups of fungi evolved. Molecular data suggest that fungi are much older than indicated by the fossil record, and may have arisen more than one billion years ago Parfrey et al. How many fungi are there? No one knows for sure how many species of fungi there are on our planet at this point in time, but what is known is that at least 99, species of fungi have been described, and new species are described at the rate of approximately per year Blackwell ; Kirk et al. A conservative estimate of the total number of fungal species thought to exist is 1. To come up with this figure, Hawksworth estimated the known numbers of plant and fungal species from countries in which both plants and fungi have been well-studiedâ€”Great Britain and Ireland, in this caseâ€”and determined there were six fungal species for every native plant species. The total number of plant species worldwide is approximately ,, and if the ratio of fungi to plants in Great Britain is typical of what occurs elsewhere, there should be at least 1. Assuming a relatively constant rate at which new species are described, it will take more than years to catalog and describe all remaining fungi. However, many of these fungi are likely to become extinct before they are ever discovered given current rates of habitat and host loss. These habitats are exceedingly rich in fungal species Hawksworth and Rossman Callan and Carris estimated that an , ha neotropical forest, such as in Costa Rica, could contain over 81, different species of plant parasitic fungiâ€”almost as many as all the known species of fungi! Consider that this estimate was based only on plant parasitic fungi, and did not take into account other ecological groups of fungi such as saprotrophs. What do fungi do? Fungi are involved in a wide range of activitiesâ€”some fungi are decomposers, parasites or pathogens of other organisms, and others are beneficial partners in symbiosis with animals, plants or algae. Fungi associated with animals Fungi have the ability to grow on and in both invertebrate and vertebrate animals. Many fungi can attack insects and nematodes, for example, and may play an important role in keeping populations of these animals under control. Insect-attacking fungi, called "entomopathogens," include a wide range of fungi in phyla Ascomycota, Zygomycota and Chytridiomycota. Some of the best-known and most spectacular entomopathogens belong in the Ascomycota genus Ophiocordyceps and related genera.

Paradoxically, humans have been using one of these entomopathogens, Ophiocordyceps sinensis, for thousands of years to treat a wide range of ailments. This fungus is an important component of traditional Asian medicine Fig. Colony collapse disorder of honeybees has been associated with co-infection by a virus and a microsporidian fungus, Nosema ceranae Bromenshenk et al. One group of fungi called Entomophthorales "insect killers" includes a number of highly specialized entomopathogens. A common example is Entomophthora musae, which is often observed forming a ring of white spores discharged around the body of a parasitized fly on panes of glass. Some fungi are specialized parasites of nematodes, rotifers, and other microscopic animals in the soil Barron A common nematode predator is Arthrobotrys oligospora, a fungus that has evolved sticky networks of hyphae for trapping nematodes. Once the nematode is immobilized, the fungus invades and consumes its body. Fortunately, there are relatively few fungal pathogens of vertebratesâ€”only speciesâ€”but some of these fungi can have devastating impacts. Consider the well-publicized frog killer, Batrachochytrium dendrobatidis, a member of phylum Chytridiomycota Berger et al. The frog chytrid is implicated in the widespread decline of frog populations around the world. Fortunately, this is the only chytrid known to parasitize a vertebrate animal and it appears to infect only amphibians. This fungus colonizes the skin on the muzzles, ears and wing membranes of some types of bats, and infected bats exhibit unusual behavior. The bat fungus is associated with declines in bat populations in the northeastern U. As of , white-nose syndrome had been confirmed in 16 states and four Canadian provinces. In humans, there are several different types of fungal infections, or "mycoses. Some fungi are members of the resident microflora in healthy people, but become pathogenic in people with predisposing conditions. For example, Candida species cause annoying yeast infections in the mucosal tissues of many healthy people, but can also cause diseases collectively called candidiasis in babies and immunocompromised individuals. Another group of fungi are inhaled as spores and initiate infection through the lungs. These fungi include Coccidioides immitis coccidioidomycosis, commonly known as valley fever , and Histoplasma capsulatum histoplasmosis. Opportunistic fungal pathogens are normally not associated with humans and other animals, but can cause serious infections in weakened or healthy individuals when inhaled or implanted in wounds. Aspergillus fumigatus, one of the most important of these opportunists, produces small, airborne spores that are frequently inhaled; in some individuals the fungus starts growing invasively, causing a disease known as aspergillosis, especially in immunocompromised individuals. A remarkable discovery was that Pneumocystis carinii, the organism causing pneumonia-like symptoms in immunocompromised patients, is a fungus and not a protozoan as had been thought for decades. Why was this pathogen classified as a protozoan? It does not respond to the common drugs used to treat fungal infections, but does respond to anti-protozoan drugs. This unusual fungus emerged as one of the leading causes of death in AIDS patients in the late twentieth century. Fungi and plants The association of fungi and plants is ancient and involves many different fungi. Plant pathogenic fungi represent a relatively small subset of those fungi that are associated with plants. Most fungi are decomposers, utilizing the remains of plants and other organisms as their food source. Other types of associations that will be discussed here include the role of fungi as decomposers, as beneficial symbionts, and as cryptic plant colonizers called endophytes. Most fungi are associated with plants as saprotrophs and decomposers. These fungi break down organic matter of all kinds, including wood and other types of plant material. Wood is composed primarily of cellulose, hemicellulose, and lignin. Lignin is a complex polymer that is highly resistant to degradation, and it encrusts the more readily degradable cellulose and hemicellulose. Fungi are among the few organisms that can effectively break down wood, and fall into two main typesâ€”brown and white rot fungi. Brown rot fungi selectively degrade the cellulose and hemicellulose in wood, leaving behind the more recalcitrant lignin. The decayed wood is brown in color and tends to form cubical cracks due to the brittle nature of the remaining lignin Fig. Brown rot residues are highly resistant to decomposition and can remain in the soil for up to years. White rot fungi are more common than brown rot fungi; these fungi degrade cellulose, hemicellulose, and lignin at approximately equal rates. The decayed wood is pale in color, light in weight, and has a stringy texture Fig. White rot fungi are the only organisms that can completely degrade lignin. Figure 4 Figure 5 An important group of fungi associated with plants is mycorrhizal fungi. There are seven major types of mycorrhizal associations, the most common of which is the arbuscular mycorrhizae, involving members of phylum Glomeromycota associated with roots of most major groups of plants. Another common type of association is ectomycorrhizae formed between forest trees and members of phyla Basidiomycota and Ascomycota. In this association, the fungus forms hyphae around host root cortical cellsâ€”the "Hartig net"â€” and a sheath of hyphae around the host roots called a "mantle. A valuable group of ectomycorrhizal fungi are truffles, members of phylum Ascomycota that form underground fruiting bodies. Figure 6 Lichens are examples of a symbiotic association involving a fungus and green algae or less frequently Cyanobacteria. The lichen thallus is composed mostly of fungal hyphae, usually with the alga or cyanobacterium confined to discrete areas of the thallus. In lichens, reproductive structures of the fungus are often conspicuous, for example disc- or cup-like structures called apothecia Fig. The fungus obtains carbohydrates produced by photosynthesis from the algae or cyanobacteria, and in return provides its partner s with protection from desiccation and ultraviolet light. Lichens grow in a wide range of habitats on nearly every continent. Figure 7 Some fungi are hidden inside their plant hosts; these are endophytes , defined by their presence inside asymptomatic plants. All plants in natural ecosystems probably have some type of symbiotic association with endophytic fungi Rodriguez et al. Endophytic fungi have been shown to confer stress tolerance to their host plant, for example, to disease, herbivory, drought, heat, salt and metals. The clavicipitaceous endophytes in the genus Neotyphodium phylum Ascomycota are among the best studied. These fungi produce alkaloid compounds that protect the grass host from insects that would otherwise feed on them; endophyte-infected turfgrass seed is sold commercially for seeding lawns and other types of grassy recreational areas. Unfortunately, livestock such as sheep, cattle, llamas and horses also are negatively affected by toxins produced by endophytes when they eat infected grass. Afflicted animals develop symptoms including tremors and jerky or uncoordinated movements. Plant pathogenic fungi are parasites, but not all plant parasitic fungi are pathogens. What is the difference between a parasite and a pathogen? In this sense, endophytic fungi discussed in the preceding paragraph are plant parasites because they live in intimate association with plants and depend on them for nutrition.

DOWNLOAD PDF INTRODUCTION, BY D. HOST.

Chapter 7 : An Introduction of D Host - Phuket, Thailand | Beach Front Resort | Official Website

Introduction to calendrierdelascience.com 11/22/; 10 minutes to read Contributors. In this article. by Tobin Titus. Introduction. calendrierdelascience.com is the root file of the configuration system when you are using IIS 7 and above.

Contact Us Introduce a guest speaker speech sample This page contains a sample introduction speech template for introducing a guest speaker or key note speaker for an event, banquet, conference, lecture and more. It also contains tips for how to write an introduction speech and techniques for delivering introductory remarks. I am so pleased to be with you tonight and to have the chance to introduce our guest speaker, say speakers name. Mention if the person was the first to do something, if they have published anything, have held impressive positions, done impressive things, etc. Without further ado, please join me in welcoming name of speaker. Tips for introducing a guest speaker 1. Keep your introduction to between 45 seconds to 3 minutes. Bios are a great place to get information. Almost all speakers will have a bio online or the event planner may have a copy of their bio. You can also search their name online for other interesting tidbits about them, but do not to include something that may be controversial or potentially embarassing to them, unless you ask them first. Ask the speaker what he or she would like you to say in their introduction. If they are very high-profile and hard to speak with personally, ask their staff or administrative assistant for information. Try to find transcripts of other speeches they have done online. Often the introductory remarks will be included and you can use them to frame your own remarks. Did you like this sample speech template for introducing a guest speaker or key note speaker? Since you were interested in this introductory speech, chances are you will like the following topics as well: Itroductory letter for Conference or Event and Anecdotes for speakers. We hope this page was helpful and provided you with some information about How to introduce a guest speaker sample template. Check out our main page for more articles here Can U Write.

DOWNLOAD PDF INTRODUCTION, BY D. HOST.

Chapter 8 : Docker overview | Docker Documentation

Introduction to the Viruses. In , Friedrich Loeffler and Paul Frosch found evidence that the cause of foot-and-mouth disease in livestock was an infectious particle smaller than any bacteria.

Introduction to the Viruses In , Friedrich Loeffler and Paul Frosch found evidence that the cause of foot-and-mouth disease in livestock was an infectious particle smaller than any bacteria. This was the first clue to the nature of viruses, genetic entities that lie somewhere in the grey area between living and non-living states. Viruses depend on the host cells that they infect to reproduce. When found outside of host cells, viruses exist as a protein coat or capsid, sometimes enclosed within a membrane. While in this form outside the cell, the virus is metabollically inert; examples of such forms are pictured below. To the left is an electron micrograph of a cluster of influenza viruses, each about nanometers billionths of a meter long; both membrane and protein coat are visible. On the right is a micrograph of the virus that causes tobacco mosaic disease in tobacco plants. An infected cell produces more viral protein and genetic material instead of its usual products. Some viruses may remain dormant inside host cells for long periods, causing no obvious change in their host cells a stage known as the lysogenic phase. But when a dormant virus is stimulated, it enters the lytic phase: The diagram below at right shows a virus that attacks bacteria, known as the lambda bacteriophage, which measures roughly nanometers. Viruses cause a number of diseases in eukaryotes. In humans, smallpox, the common cold, chickenpox, influenza, shingles, herpes, polio, rabies, Ebola , hanta fever, and AIDS are examples of viral diseases. Even some types of cancer -- though definitely not all -- have been linked to viruses. Viruses themselves have no fossil record, but it is quite possible that they have left traces in the history of life. It has been hypothesized that viruses may be responsible for some of the extinctions seen in the fossil record Emiliani, It was once thought by some that outbreaks of viral disease might have been responsible for mass extinctions, such as the extinction of the dinosaurs and other life forms. This theory is hard to test but seems unlikely, since a given virus can typically cause disease only in one species or in a group of related species. Even a hypothetical virus that could infect and kill all dinosaurs , 65 million years ago, could not have infected the ammonites or foraminifera that also went extinct at the same time. On the other hand, because viruses can transfer genetic material between different species of host, they are extensively used in genetic engineering. Viruses also carry out natural "genetic engineering": This is known as transduction, and in some cases it may serve as a means of evolutionary change -- although it is not clear how important an evolutionary mechanism transduction actually is. The tobacco mosaic virus picture was provided by the Rothamstead Experimental Station. Both servers have extensive archives of virus images. The Institute for Molecular Virology of the University of Wisconsin has a lot of excellent information on viruses, including news, course notes, and some magnificent computer images and animations of viruses.

Chapter 9 : Writing Your Talk Show Script

It is important to note that, it is impossible to connect the two peripheral devices using USB unless there is a separate host available that controls the communication and serves as the main handling device in the whole arrangement of communication between the peripheral devices.

Scanning electron micrograph of HIV-1 viruses, coloured green, budding from a lymphocyte In the French microbiologist Charles Chamberland invented a filter, known today as the Chamberland filter or Chamberlandâ€"Pasteur filter, that has pores smaller than bacteria. Thus he could pass a solution containing bacteria through the filter and completely remove them from the solution. His experiments showed that extracts from the crushed leaves of infected tobacco plants remain infectious after filtration. At the same time several other scientists proved that, although these agents later called viruses were different from bacteria, they could still cause disease, and they were about one hundredth the size of bacteria. In the Dutch microbiologist Martinus Beijerinck observed that the agent multiplied only in dividing cells. Having failed to demonstrate its particulate nature, he called it a " contagium vivum fluidum ", a "soluble living germ". Counting these dead areas allowed him to calculate the number of viruses in the suspension. They have probably existed since living cells first evolved. The origin of viruses remains unclear because they do not form fossils, so molecular techniques have been the most useful means of hypothesising how they arose. However, these techniques rely on the availability of ancient viral DNA or RNA but most of the viruses that have been preserved and stored in laboratories are less than 90 years old. Over time, genes not required by their parasitism were lost. The bacteria rickettsia and chlamydia are living cells that, like viruses, can reproduce only inside host cells. They lend credence to this theory, as their dependence on parasitism is likely to have caused the loss of genes that enabled them to survive outside a cell. The escaped DNA could have come from plasmids â€"pieces of DNA that can move between cellsâ€"while others may have evolved from bacteria. The escape hypothesis does not explain the structures of virus particles. The coevolution, or virus-first hypothesis, contravenes the definition of viruses, in that they are dependent on host cells. The arrangement of the capsomers can either be icosahedral sided , helical or more complex. Some viruses are surrounded by a bubble of lipid fat called an envelope. Size[edit] Viruses are among the smallest infectious agents, and most of them can only be seen by electron microscopy. They are so small that it would take 30, to , of them, side by side, to stretch to one cm. Some viruses such as megaviruses and pandoraviruses are relatively large. At around 1 micrometer , these viruses, which infect amoebae , were discovered in and They are around a thousand times larger than influenza viruses and the discovery of these "giant" viruses astonished scientists. These genes encode structural proteins that form the virus particle, or non-structural proteins, that are only found in cells infected by the virus. The influenza virus, for example, has eight separate genes made of RNA. When two different strains of influenza virus infect the same cell, these genes can mix and produce new strains of the virus in a process called reassortment. Cells produce new protein molecules from amino acid building blocks based on information coded in DNA. Each type of protein is a specialist that usually only performs one function, so if a cell needs to do something new, it must make a new protein. Viruses force the cell to make new proteins that the cell does not need, but are needed for the virus to reproduce. Protein synthesis consists of two major steps: These migrate through the cell and carry the code to ribosomes where it is used to make proteins. Information is hence translated from the language of nucleic acids to the language of amino acids. For this reason, these viruses are called positive-sense RNA viruses. These are called negative-sense RNA viruses. The species of viruses called retroviruses behave completely differently: Viral life cycle and Viral entry Life-cycle of a typical virus left to right ; following infection of a cell by a single virus, hundreds of offspring are released. When a virus infects a cell, the virus forces it to make thousands more viruses. This specificity restricts the virus to a very limited type of cell. Plant viruses can only attach to plant cells and cannot infect animals. This mechanism has evolved to favour those viruses that only infect cells in which they are capable of reproducing. Penetration follows attachment; viruses penetrate the host cell by endocytosis or by fusion with the cell. Uncoating happens inside the cell when the viral capsid is removed and destroyed by viral enzymes or host enzymes, thereby exposing the viral nucleic acid. Replication of virus particles is the stage where a cell uses viral messenger RNA in its protein synthesis systems to produce viral proteins. Assembly takes place in the cell when the newly created viral proteins and nucleic acid combine to form hundreds of new virus particles. Release occurs when the new viruses escape or are released from the cell. Most viruses achieve this by making the cells burst, a process called lysis. Other viruses such as HIV are released more gently by a process called budding. Effects on the host cell[edit] The range of structural and biochemical effects that viruses have on the host cell is extensive. Cells in which the virus is latent and inactive show few signs of infection and often function normally. This is often the case with herpes viruses. One of the results of apoptosis is destruction of the damaged DNA by the cell itself. Some viruses have mechanisms to limit apoptosis so that the host cell does not die before progeny viruses have been produced; HIV , for example, does this. Ten Norovirus particles; this RNA virus causes winter vomiting disease. It is often in the news as a cause of gastro-enteritis on cruise ships and in hospitals. Common human diseases caused by viruses include the common cold , the flu , chickenpox and cold sores. Many viruses cause little or no disease and are said to be "benign". The more harmful viruses are described as virulent. Viruses cause different diseases depending on the types of cell that they infect. People chronically infected with a virus are known as carriers. They serve as important reservoirs of the virus. If there is a high proportion of carriers in a given population, a disease is said to be endemic. Many viruses that infect plants are carried by organisms ; such organisms are called vectors. Some viruses that infect animals, including humans, are also spread by vectors, usually blood-sucking insects. However, direct transmission is more common. Some virus infections, such as norovirus and rotavirus , are spread by contaminated food and water, hands and communal objects and by intimate contact with another infected person, while others are airborne influenza virus. Viruses such as HIV, hepatitis B and hepatitis C are often transmitted by unprotected sex or contaminated hypodermic needles. It is important to know how each different kind of virus is spread to prevent infections and epidemics. Plant pathology Peppers infected by mild mottle virus There are many types of plant virus , but often they only cause a loss of yield , and it is not economically viable to try to control them. Plant viruses are often spread from plant to plant by organisms vectors. These are normally insects, but some fungi , nematode worms and single-celled organisms have been shown to be vectors. When control of plant virus infections is considered economical perennial fruits, for example efforts are concentrated on killing the vectors and removing alternate hosts such as weeds. Bacteriophage The structure of a typical bacteriophage Bacteriophages are viruses that infect bacteria and archaea. The International Committee on Taxonomy of Viruses officially recognises 28 genera of bacteriophages that belong to 11 families. Bacteriophages are useful in scientific research because they are harmless to humans and can be studied easily. These viruses can be a problem in industries that produce food and drugs by fermentation and depend on healthy bacteria. Some bacterial infections are becoming difficult to control with antibiotics, so there is a growing interest in the use of bacteriophages to treat infections in humans.