**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **21/07/2020** | **Name:** | **Nayanashree** |
| **Course:** | **Coursera** | **USN:** | **4AL16EC042** |
| **Topic:** | **Network security and vulnerabilities** | **Semester & Section:** | **8th B** |
| **Github Repository:** | **Nayana\_online** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
|  |
| **REPORT**  **BASICS OF IP ADDRESSING AND THE OSI MODEL**  **The Basics of Binary:**  This lesson is being presented by Ben Briggs, and is based upon a lecture series developed by Moises Monge. Let's talk about networking fundamentals, specifically IP addressing   * This lesson will cover the basics of IPv4 addressing and touch on IPv6. I'm sure you've all been exposed to base 2 or binary numbers in school, but I'll give you a quick overview since that was a long time ago for some of us. We're used to thinking, talking, and calculating in base 10, or decimal number system * Humans adopted the base 10 number system most likely because we have 10 fingers and 10 toes, so it's easy to speculate that early counting was done on our ancestors fingers * Computers, however, only know two different states, on and off or high-voltage, low-voltage, or open gate, closed gate, which we represent logically as one or zero * Computers can only differentiate between these two states, so this is how the decimal system works. We have digits for ones, tens and hundreds, thousands, ten thousands, etc. Each digit being worth 10 times as much as the preceding digit * Each digit can hold 10 different values, ranging from 0-9. Remember that the word decimal means 10. In the binary system, on the other hand, each digit or placeholder holds a different value * There are placeholders for 1s, 2s, 4s, 8s, 16s, etc. Each digit being worth two times as much as the preceding digit. In binary, each digit can hold only two different values, zero or one. Remember that binary means two * So converting from binary to decimal is easy, just add up all the values of the digits that contain a one and ignore those that contain zeros. In this example, we have an eight digit binary number, so there are eight placeholders * The decimal equivalent of this number would be 2 plus 8 plus 16 plus 64 plus 128, or 218. Converting from decimal to binary is a little more awkward, that might remind you of the long division you learned in grade school. Of course, there are plenty of calculators and online tools that will make these conversions for you, but here's how it's done. In this example, we have to convert 235 from decimal to binary.   **IP ADDRESS STRUCTURE AND NETWORK CLASSES:**   * IPv4 uses a 32-bit addressing schema that is divided into four octets of eight bits each. Now you should remember from the previous video that eight digits of base two, ones and zeros can have values ranging from 0-255 in decimal notation, which is two raised to the eighth power. Expressed in decimal format, this gives IPv4 addresses arranged from 0.0.0.0 with all bits off to 255.255.255.255 with all bits on * Since the value of any of the four octets could range between 0 and 255, this gives IPv4 a very large number of possible addresses, 4,294,967,296 to be exact. This seems like a very big number, but we're already getting short of IPv4 addresses. If we convert this IP address to binary, the way the computer sees it, this will be its representation. Using the same method we used in the last video, we can convert each octet from decimal to binary * For the decimal number 10, we check the 16 place holder. Sixteen is greater than 10, so we put a zero there. So the next place holder is eight, and we can subtract 8 from 10. So that place holder gets a one and we have two left over. You cannot subtract 4 from 2, so the four place holder gets a zero. You can subtract 2 from 2, so the two place holder gets a one and we have zero left over. With zero left over, we're done * So any remaining place holders get a zero. You can see each octet contains eight bits, which is why it is called an octet. An IP address is divided into a network portion and a host portion, which is something that you can configure on your own computer. But most of the time computers are set up now to allow DHCP or Dynamic Host Configuration Protocol to dynamically configure IP addresses for you. So let's take a look at this in action. By logging into our server, server 100, let's take a look at the interface IP address * This whole number is called the CIDR range. The slash 24 defines how many bits of the IP address are dedicated to the network portion of the address. So each IPv4 address has a network portion and a host portion. The size of the host portion defines how many hosts or endpoints this network segment can hold. In this example, the network portion uses 24 bits of the 32-bit address, which leaves eight bits for the host portion * Two raised to the eighth power is 256. So that's the largest number of hosts that this network segment can support, 0-255. In the early days of IPv4, networks used the classful addressing schema, which allowed for only five different address ranges * Class A goes from 0.0.0.0 to 127.255.255.255. This is for special use and unicast. The default subnet mask is 255.0.0.0, which we will explore more later on. This is the class B, the class C, the class D, and class E, use these address ranges * Class D is reserved for multicast groups. So you will see protocols like that bios using addresses and this range to communicate. Finally class E, which is reserved for research development and future uses. So this is classful addressing. In class A networks, the first octet is used for the network portion and the last three octets are used for the host portion * In class B networks, the first two octets we dedicated to the network and the last two to the host. Class C networks have the first three octets dedicated to the network, and only the last octet is dedicated to the host.   **IP PROTOCOL AND TRAFFIC ROUTING**  The Internet Protocol, or IP protocol, works with layer 3 devices which use the IP header to identify and process traffic. All routers inspect the destination address of each packet, but stateful firewalls also inspect the source address so they can identify where the traffic is coming from. As we saw in the last video, IP addresses are represented by a quad dot notation or a string of four numbers separated by dots, for example, 10.195.210.10. As you can see, there are four octets or four groups of eight binary bits separated by dots. In decimal form, an eight-digit binary number can take on a value from 0-255, always a positive integer. In binary form, the range is expressed as 00000000-11111111. A routable protocol is a protocol that can be routed outside of the network it was originated in. Normally, this would be the Internet. IP is a routable protocol, but not all IP addresses are routable.  **INTRODUCTION TO THE IPV6 ADDRESS SCHEMA**  Since the world is running out of IP addresses using IPv4 protocol, the latest version of the IP protocol, IPv6 extends the address length from 32 bits to 128 bits. We're going to be dealing with hex numbers now, so refer back to the first video in the lesson, if you need a refresher   * An IPv6 address being 128 bits long, is four times longer than the 32 bit IPv4 address. But that does not give us only four times as many addresses. In the first video in this lesson, we learned that 2 the 32nd power gives us just under 4.3 billion possible addresses. Well, 2 to the 128th power in decimal format is about 3.4 times 10 to the 28th power. Which, according to Wikipedia would be called 34 octillion * But that's a very big number, like the number of atoms in an elephant big. An IPv6 address is divided into eight four-digit hexadecimal values, each separated by a colon as shown here. Each single hexadecimal digit can have 16 possible values, which makes it a four-bit long binary. So a group of four hex numbers would be 16 bits. And there are eight of these in the IPv6 address, so 8 times 16 brings us up to 128 bits. There are a few rules to remember when representing an IPv6 address.   An IPv6 address is not case sensitive. You don't need to specify leading zeros in the address, and you can use a double colon to represent any number of consecutive zeros. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date:** | **21/07/2020** | **Name:** | **Nayanashree** | |
| **Course:** | **Salesforce** | **USN:** | **4AL16EC042** | |
| **Topic:** | **Salesforce platform basics** | **Semester& Section:** | **8th A** | |
| **AFTERNOON SESSION DETAILS**    **s2.PNG** | | | |
| **REPORT**  **LEARNING OBJECTIVES**   * After completing this unit, you’ll be able to: * Locate Setup and identify its key elements. * Identify important menus for customizing your org. * Use Quick Find to access menu items.   **Setup: Your New Work Home**   * Earlier, we mentioned that you’ll spend a lot of time in Setup during your time as a Salesforce administrator. And we weren’t kidding. Setup is your one-stop-shop for customizing, configuring, and supporting your org * Since there’s so much you can do in the Setup area, it’s important to get comfortable with navigating it. There are a few ways to approach it. As you learn what’s available to you, you’ll get more comfortable finding the things you need * You can get to Setup from any page in your Salesforce org. From the gear menu at the top of the screen ( The gear icon to open Setup.), click Setup. Let’s get familiar with the Setup area * **Object Manager:** Object Manager is where you can view and customize standard and custom objects in your org. * **Setup Menu:** The menu gives you quick links to a collection of pages that let you do everything from managing your users to modifying security settings. * **Main Window:** We’re showing you the Setup home page, but this is where you can see whatever it is you’re trying to work on.   The Setup Menu is the trickiest piece to navigate because there are so many different pages you can access. There are two ways to get where you want to go. If you already have an idea where to look, expand the appropriate menu and select the page you want. If you aren’t sure where to look, use the Quick Find box to search. Lets say you wanted to manage your user permission sets. If you happen to know that permission sets are in the Users menu under Administration, just open that menu and click Permission Sets. Otherwise enter Permission Sets in the Quick Find box.  **Get Cozy with the Setup Menu**  There are three main categories in the Setup menu: Administration, Platform Tools, and Settings. Let’s take a look at what’s available.   * **Administration:** The Administration category is where you manage your users and data. You can do things like add users, change permissions, import and export data, and create email templates. * **Platform Tools:** You do most of your customization in Platform Tools. You can view and manage your data model, create apps, modify the user interface, and deploy new features to your users. If you decide to try your hand at programmatic development, Platform Tools is where you manage your code as well. * **Settings:** Finally, Settings is where you manage your company information and org security. You can do things like add business hours, change your locale, and view your org’s history.   **UNDERSTAND THE SALESFORCE ARCHITECTURE**  **Learning Objectives**   * After completing this unit, you’ll be able to: * Define key terms related to the Salesforce architecture. * Find information related to trust. * Explain at least one use case for Salesforce APIs. | | | |