

Forum:[Hack Forums](#)

Thread: [== SECRECY == >>> NETWORKING FOR HACKERS \(Unskid Yourself Part 2\) <<<](#)

Pronouncer - August 2, 2017, 4:15 pm

Introduction

Thank you for your interest in this thread. Before reading this thread we suggest you read this <https://hackforums.net/showthread.php?tid=5676945>
That will teach you how to start un-skidding yourself this is part 2 explaining the networking.

Topics

Networking Topologies

Wan Lan Wlan

Physical and Logical Connections

Subnets

OSI Model (All layers)

Understanding Switches/Hubs

Server VS Client

Server side processing

Client side processing

TCP/IP

UDP

TCP

Common Ports

DNS Systems

Hacking DNS

Name Servers

DNS Records and Zone Files

DHCP

DHCP Client/Server Communication

The Lease Concept

DHCP Resource Exhaustion

NAT VS DHCP

Data Logging (Advantages and disadvantages)

Multi-Threading

Multi-Processing

Ipv4

Ipv6

MITM

TCP Handshake

Port scanning

L7 methods (SLOWLORIS, HEAD, GET, POST)

Congratulations

You have made the first step in going further than the majority of HF

Pronouncer - August 2, 2017, 4:19 pm

Introduction to the Internet

In this section, we'll be covering on how your Networks connect together. There are a lot of devices that will connect your device to the internet. Modems, Routers, Switches, Wireless Access Points etc.

When connecting to the internet, your computer goes through a switch, firewall, through a router or a modem and then to the Internet.

We'll be splitting the concepts into so called Physical and Logical Concepts. The physical way that a computer can be connected is through a computer, network cable, switch, router or modem. This is a physical connection as the parts are each physically connected. When it comes to logical we can split the networks, for example, let's say we have 20 computers that are all connected physically via the same switch. What we can do, even if they are physically connected, is split the computers in the network. We can split them into two separate networks, this is what we call a Logical Network.

Even though all physical devices are combined together, we, in the computing world, class them as individual devices (Logic). The only thing that's different is it's all connected in one outer shell.

Internet Speed

Speed is a very confusing topic. Data is read in two measurements. Bits and Bytes. You can mostly differentiate them in a big or small B (Bits will be marked as »b« and Bytes will be marked as »B«). They sound very similar and look similar, but they are different in size. 8 bits equals to 1 byte (Or 8b = 1B). You need to get used to the fact that b (Lowercase) is Bits and B(Uppercase) is Bytes.

Storing Data will always be in bytes. Companies like Verizon and Bt will always offer internet speeds in bits. In the early 2000 and right up to 2010 this was very popular as speeds weren't that great and the connection methods were different than what we have now, but we will cover that in another chapter. Let's get back to the offers.

For example, Bt offers you 100 megabits, and to those who don't know a lot about how speeds are classified, that seems like a lot! But think again, they're offering you 100 Mb, not 100 MB. You can get your actual speed by doing a simple math equation;

$$100 \text{ (Megabits on Offer)} / 8 \text{ (1Byte is 8 Megabits)} = 12.5 \text{ MB (Actual Download Speed)}$$

When the internet was just coming around, internet speed was advertised as 12Mb, one of the reasons for that was because at the time, the connection method we were using could only support that speed.

Modem

The modem is the first device that connects your internal network to the World Wide Web. Your modem can either be a T1 Connection, a cable connection, DSL or even Satellite. All these types need a modem to be functional. So what does the modem actually do?

The modem turns your Ethernet protocol which is the language you use on your network (An

example being turning your connection into noise). During the early 2000s, when we were using the internet via a Dial-Up Modem, you could hear the noises when your Modem would transmit it's data to the telephone company. When using a DSL Modem, your data is turned into a language that your ISP can understand. In simple terms, a modem is just a connector to the internet.

Connection Types

A T1 Connection is the best connection to the internet, but they are very expensive and in 2015, they aren't very fast. There is T2 and T3 but these are even more expensive than the T1 Connection.

The next type of connection is a DSL (Digital Subscriber Line). This is given through a phone company. DSL Signals are good and are often cheap, although with DSL, the max speed is around 12mbps, ADSL2 can bump up the speed to 24mbps. The benefit of DSL is that you have a solid connection from your modem to the internet network center. You have a Dedicated Line from your house and your speed will never be affected if your neighbors go on at the same time. DSL has faster download than upload speeds and that will never be the other way around.

Another type of connection is Cable Modem. Cable internet has introduced DOCSIS 3.0. When this is implemented, it will allow 160mbps (Which is roughly 20MBps) but we now have developed technology which is increasing speeds. The problem with Cable Connections is that you share the main bandwidth with your neighbors, if your neighbors get on you will share bandwidth with them.

There is also Satellite Internet, but very few people use this. An advantage to this is if you are somewhere without any internet access, you can use Satellite internet. It has a slow download speed of only about 1.5mbps and the upload speed is worse. With an upload speed like the one you're using the on-Satellite Internet, you cannot even use VoIP (Voice over Internet Protocol).

These days, new wireless networks such as 4G are coming to light. These connect your device to a cell tower. We will not be explaining this.

Router

The next device we will be covering is the Router. The router allows you to separate different networks (Similar to if you're a tenant in a building and they offer internet connection). They will have to separate the network so that tenants cannot take advantage of each other. Each tenant is on their own network but they will all be connected to the same modem. Routers are used logically to separate networks.

Firewall

The next Device is a Firewall. This is a device that tries to prevent hackers from gaining access to a specified network. Firewalls can get very complicated and are not only physical items, they can be software, but we won't be covering software right now. A firewall will

also prevent internet traffic that you don't want. Firewalls are mostly a security device built into your modem or router. If you turn it on, it will block the service for the entire network, for example, if you block port 80 on the firewall, nobody in that network will be able to access the internet.

Virtual Private Networks

Virtual Private Networks (or commonly known as VPNs) are a great way of allowing people from outside the network to connect to it. This allows computers from around the world to connect to a network, in computing terms, the connecting PC is now a part if your network. A VPN Allows access to shared documents. The downside of a VPN is speeds, the VPN may be hosted on a gbps upload,

but if the client has 10Mbps upload and download speeds it's going to take longer to access all of the files on the VPN.

Virtual Private Networks

Hubs and Switches are two different things, Hubs split the network equally to all of the clients connected to it.

Wireless Access Points

Now that we've got all of the wired devices connected and talking to each other, we need to go over how wireless devices (Such as a phone) connect. They connect via a WAN (Wireless Access Point). This is a device that allows wireless devices to join the network, this is mostly installed on modern routers (Newer modems have antennas) and have a cable that goes from the access point to a switch.

There are four standards, 802.11 A, B, G, N. 802.11A didn't succeed very well, so I'd advise against buying 802.11a devices. A is not compatible with other types while others are. 802.11B is an old standard that worked at 11mbps, but it's a slow wireless connection. 802.11G speed provides steady speeds of around 54mbps and is faster and a lot more stable than B. 802.11N is the new standard, there was Pre-N which was buggy and not perfect but still exists to this day. If you're using Pre-N, you should change to N. 802.11N allows for a larger area covered by the wireless signal, they are a lot faster than G Networks and the

best thing is they allow real-time integration, so VoIP is no longer a problem, as the delay would not be major.

OSI REFERENCE MODEL

The Open Systems Interconnection model (also known as OSI) is the standard conceptual model of the modern network communication system. This model can be seen has a networking framework which allows us to implement layered protocols. This framework is divided in 7 layers, each one passing information to another sequentially.

These seven layers are also divided in two groups, one consisting of the top three, whose objective is to define how the applications will communicate with each other and with the users, and the bottom four layers, that will define how data is transmitted from source to destination.

The OSI model is also seen as a simple but effective way to find and troubleshoot problems on a given network.

Pronouncer - August 2, 2017, 4:20 pm

Layers

Layer 1 – Physical Layer:

- The physical layer is the responsible layer for the transmission of digital data (bits) from the source, over network communication, to the destination (all this happens on the first layer).
- Some examples of Physical Layer technologies are:
 - Hubs;
 - Routers;
 - Ethernet Cables;
 - Repeaters;
- At the physical layer, all the data is transmitted at a mechanical level, with the usage of: electric voltage, radio frequencies and infrared or light pulses.

Layer 2 – Data Link Layer:

- The data link layer will have two main objectives: check for physical transmission errors and gather and package the received bits into data frames. This layer will also control the access of various network devices to the physical medium by managing physical addressing schemes (such as MAC).

The data link layer is considered the most complex layer on the OSI model, leading it to be often divided into two different parts (Logical Link Control and Media Access Control).

Layer 3 – Network Layer:

- The network layer is the one layer in command of routing the data originated from Layer 2, so it reaches its final destination. When data arrives at the network layer, the source and destination address within each frame will be examined to determine if the data has reached its final destination.

- If final destination was reached, the data will be formatted into packets that will be delivered to the transport layer (coming next). If not, the network layer will update the destination address and routes the frame back to the lower layers.

- In order to support routing, the network layer must maintain logical addresses for devices on the network (IP's for example). The network layer will also manage and organize the mapping logical addresses and physical addresses, which if in an IP network, will be accomplished through Address Resolution Protocol (ARP).

Layer 4 – Transport Layer:

- The transport layer will deliver received packets from the network layer across network connections. Different and various transport protocols can be used, many of them supporting a range of different capabilities, such as flow control, error recovery, retransmission, and the most common protocol used by the transport layer is TCP.

Layer 5 – Session Layer:

- The session's layer manages the sequence of events that open and close network connections. The session layer is built with the objective of supporting multiple types of connections that can be created dynamically.

Layer 6 – Presentation Layer:

- The presentation layer is relatively simple in function, as it handles the syntax processing of data such as format conversions and encryption/decryption, which is needed to support the next layer (Application Layer).

Layer 7 – Application Layer:

- The application layer is the final layer that supplies network services to the end user. Network services are typical protocols made to work with the end user's data. For example, the HTTP protocol is used on web browsers to package the needed data to send and receive web page content.

Understanding Switches

A network switch is a small (but important) hardware device that centralizes the communication of multiple connected devices on a local area network. These can be used to connect multiple devices

through Ethernet cable (or other types) to a LAN network, or to extend the already existing network with a bigger number of ports. Common house routers provided by the ISP usually integrate a switch into the unit as a feature and core function, but switches are also widely used in corporate networks and data centers.

The most common switch type is the Ethernet switch. These are both available for the common house hold (Consumer grade) and for corporations and major businesses. The consumer grade switch usually supports a Gigabit speed connection per switch port, and comes with 4 to 8 ports, while corporate level switches can support up to 10 Gigabit speeds per switch port, and comes with 32 to 128 ports.

Most corporate level switches also come with 1 or 2 trunk ports, which are used to move data and information between switches, thus originating daisy-chaining.

Switches can also have a different communication method. They can either be simplex/half-duplex or full duplex. Simplex/half-duplex switches cannot receive and transmit data at the same time at a certain port, while full duplex can.

Switches can be divided in two categories:

- Managed Switches.
- Unmanaged Switches.

Unmanaged switches are usually consumer grade switches, and just like the name itself, they have no special configuration settings beyond plug and play capabilities, and do port configuration by themselves. Managed switches are commonly found on major corporations and businesses, can fully customize the configuration of ports, bandwidth, and can support QOS (quality of service), SNMP monitoring and link aggregation.

Traditional managed switches can only be controlled and configured through a UNIX style command line interface, while most modern switches support web based interfaces, much like the common router.

Switches may resemble hubs and routers, but in both cases, there are small differences.

Between switches and hubs, the major difference is on how incoming data is treated. When talking about hubs, any data that reaches it will be transmitted to every other device connected to the hub, whilst switches have the capability of analyzing received packets for the source and destination, and will re-route the packet only to the specific receiving device. This is called Packet Switching, and can conserve network bandwidth, and so provides improved performance versus a common hub.

When it comes to switches and routers, the difference is simple. Although both centralize local device communications, only the router can communicate with outside networks (i.e. the internet).

Most switches (both traditional and modern) work at a Layer 2 level (Data Link Layer from the OSI Model) which allows the switch to examine each packet and have access to the MAC address, and so be able to identify each device in a much easier and simpler way. But nowadays some major corporations are implementing the use of Layer 3 switches (Network Layer), which combines the internal hardware logic of switches and routers,

creating an hybrid device that, compared with layer 2 switches, can have a better support for Virtual Local Area Networks (VLAN's).

TCP/IP and SUBNET

TCP/IP (Transmission Control Protocol/Internet Protocol) is the fundamental communication protocol or convention of the Internet. It was made to standardize the web so that all system could communicate effectively.

The Open Systems Interconnection (OSI Model) separates the distinctive conventions into 7 layers.

A layer serves the layer above it and is served by the layer underneath it. This is the core model for discussing network troubleshooting and programming.

Server vs Client

Client side and server side are far separated from being "similar"

A case of this is the point at which you open a website in your browser, (for example, chrome or Firefox) you'll get the page clearly, yet behind that your browser is transferring and receiving the html, css, javascript code, as well as a couple of different other parts to your page; presenting it so a human can see and read it.

When you open a site, your program doesn't have much to do with the server-side. When you open any page, the client side will essentially initiate the server-side connection and pull the source code processing it in a manner that a human can read. If you somehow managed to run server-side content, for example HTML, it will render the source code as opposed to displaying it in a design. To have your program distinguish between HTML, for example, you'd require another source, or a "go between". There are numerous examples out there for this, such as Apache. Apache will make an interpretation of your PHP into the right design so you can read it.

Server Side - In server-side computing, an advantage is that the server won't require a client to download particular modules (like Java) to run particular content, such as scripts.

The downside is page post back, which can instigate server handling and may diminish the execution and cause sit tight time for the customer. **Client Side** - with client side preparing a portion of the favorable circumstances are that operations can be executed rapidly since the customer doesn't have to interface with the server to do as such. The

inconvenience to customer side preparing is that creating with scripting dialects is an all the higher exertion work and the customer's program is required to help the dialect utilized.

Subnetting

A subnetwork or subnet is a coherent subdivision of an IP network. The act of partitioning a network into at least two subnetworks is called subnetting.

PCs that belong to a subnet are addressed with a typical, identical, bit-group in their IP address. network or routing prefix and the "rest" field or host identifier. The rest field is an identifier for a particular host or system interface.

The routing prefix might be communicated in Classless Inter-Domain Routing (CIDR) documentation composed as the principal address of a system, trailed by a cut character (/), and finishing with the bit-length of the prefix. For instance, 192.168.1.0/24 is the prefix of the Internet Protocol Version 4 arrange beginning at the given address, having 24 bits apportioned for the system prefix, and the rest of the 8 bits held for have tending to. The IPv6 address detail 2001:db8::/32 is an extensive address obstruct with 296 locations, having a 32-bit directing prefix.

For IPv4, a network may likewise be portrayed by its subnet mask, which is the bitmask that when connected by a bitwise AND operation to any IP address in the system, yields the steering prefix. Subnet veils are likewise communicated in speck decimal documentation like an address. For instance, 255.255.255.0 is the system cover for the 192.168.1.0/24 prefix.

Activity is routed between subnetworks with unique gateways (routers) when the directing prefixes of the source address and the destination address contrast. A router constitutes the consistent or physical limit between the subnets.

The advantages of subnetting a current system fluctuate with every arrangement situation. In the address distribution engineering of the Internet utilizing CIDR and in extensive associations, it is important to assign address space effectively. It might likewise improve steering effectiveness, or have preferences in arrange administration when subnetworks are authoritatively controlled by various substances in a bigger association. Subnets may be arranged logically in a hierarchical architecture, partitioning an organization's network address space into a tree-like routing structure.

Port Procols

UDP is the easiest protocol you'll have to deal with networking. It is very fast because it sends the data without checking if the receiver got them. Let's see it's diagram:

Source / Port Dest./ Port / Total Length / Checksum / Data

Only 4 pieces of information in the UDP-Packets Headers, each being 2 bytes, each packet only has an 8-byte header. That's it for UDP, it's really that easy.

TCP

In a TCP Connection, packets have roles. They can establish a connection, end it, reset it. The packet's role is in its header, of course. For that, TCP has a system of flags. These are bits which are either 0 or 1.

The flags that exist are:

1. SYN - Synchronisation -> Connection Establishment
2. ACK - Acknowledgement -> The »Copy that« for machines
3. FIN - Ends a connection
4. RST – Reset the connection (This is an important packet, if someone tries to establish a connection with a server on a closed port, the server has to reply with an RST Packet.)
5. PSH and URG -> Connection has Priority.

Let's describe how a standard TCP Connection works. (I strongly recommend you use wireshark to see all of this yourself).

Connections over TCP

1. Client sends a packet with the SYN (Synchronisation) flag, he wants to start a connection
2. The server agrees with a ACK (Acknowledgement) packet and starts a connection to the client with a SYN packet. (There is a connection for each direction)
3. The Client accepts the server connection with a SYN Packet (This is called a Threeway Handshake).
4. The connection is set and the data transmission can start. (Note: All of these packets were empty)

Now that the connection is set, the data transmission begins. All of these packets have the »ACK« flag on, the connection ends with these three empty packets.

The TCP Segment (A new word, there was the Ethernet Frame, the IP Datagram, the UDP Datagram and now the TCP Segment)

Source Port / Dest Port / ??? / Flags / ??? / Checksum / ??? / Datas

TCP is way heavier than it's UDP opposite, it has a header the size of 20 bytes. The »???« fields are a little complicated and absolutely not necessary to understand TCP, that's why I skipped them.

Final Words:

Remember: The flags only exist in TCP, not in UDP. • You can identify the OS running on a machine if you send weird TCP-Packets.

If you start a connection with a [FIN] or [RST], Microsoft and *nix systems won't answer the same thing. This is hacking: Sending unexpected packets. For example, you can troll the server if you send »X-Mas Packets«, a packet with all flags on.

DNS Hijacking

DNS Hacking is a type of malicious attack that alters a computer's TCP/IP settings. An attacker takes control of a computer to alter its DNS settings, so that it points to a rogue DNS server, the process is referred to as DNS hijacking.

In most cases, attackers make use of malware programs such as a Trojan Horse to carry out DNS Hijacking, the trojans are often disguised as videos, audio codecs, video downloaders and other free utilities. The DNSChanger Trojan is an example of one such malware that hijacked the DNS Settings of over 4 million computers, to drive a profit of about 14 million USD through fraudulent advertising revenue.

WHAT IF I ALREADY AM A VICTIM OF DNS HIJACKING?

If you suspect that your computer is infected with a malware such as DNSChanger, there's no need to panic. It's fairly simple and easy to recover from the damage caused by the malware. All you have to do is verify your current DNS Settings and make sure that you are using a correct, non-blacklisted DNS IP.

What actually is a DNS?

DNS (Dynamic Name Server) is a service that maps domain names to IP Addresses. We use domain names because they are easier to remember than IP's.

How DNS hacking works

DNS is responsible for mapping the user friendly domain names to their corresponding IP Addresses. This DNS server is owned and maintained by your ISP and many other private business organizations. Imagine a situation where a hacker or a malware program gains unauthorized access to your computer and changes the DNS Settings, so that your computer now uses one of the rogue DNS servers that is owned and maintained by the hacker.

Types of Attacks

Pharming: This type of attack is where a website's traffic is being redirected to another website that is most likely a fake one. For example, when a victim tries to visit Twitter, they may get redirected to a different website full of advertisements.

Phishing: This type of attack is where a victim is redirected or brought to a malicious website that has the same layout as another website, when the victim tries to log-in, the credentials get logged and sent back to the hacker.

Name Servers

The most important part of domain ownership is choosing and specifying name servers. The internet will not be able to find your DNS information and your domain won't resolve if you don't specify name servers for your domain. Name servers host the domains DNS information in a text file which is called the "Zone file"

DNS RECORDS AND ZONE FILES

DNS records should be specified which match domain names to IP addresses.

A “Zone file” consists of grouped DNS records.

This allows your domains correct IP to be looked up on the internet.

Zone files contain some information about the owner such as the admin’s email address, DNS records and the name servers.

How DNS works

A domain name of a site needs to get translated into the IP address of the server hosting the website.

DNS helps make the internet easier, it matches a human-readable domain to a computer friendly IP address like 216.58.201.174 (This is all done in the zone file which holds the DNS records and their corresponding IP Addresses)

Breakdown of Process

When you type a domain like “google.com” into your browsers address bar, A query is sent by your browser which goes through your ISP. DNS resolver of the ISP inquire the Root Name Server about “Where .com domains Name Server can be found”

“A query is a question that seeks to match a domain to it’s corresponding IP address.”

The Root Name Server sends a response with the IP address for the .com Name Server

The DNS resolver for the ISP visits the .com Name server using the IP that was sent by the Root Name Server, then it asks where the name server for google.com can be found.

Name Server for the .com responds with the IP addresses for the Name server of google.com

The DNS resolver of the ISP then visits the Name Server of google.com and reads the Zone file.

Zone file contains information about which IP address goes with the domain.

Your ISP now has the IP for google.com it then connects to the google.com server.

Lastly the webserver handles your request and makes sure it serves the correct files and folders which get displayed on your browser.

Pronouncer - August 2, 2017, 4:21 pm

MORE OSI LAYERS

Layer 5

This layers name tells you allot about what it’s purpose is It allows devices to connect and manage sessions. In simple terms, a session is a logical persistent linking of two software applications processes to enable them to exchange data over a longer period of time. In some cases, these

sessions are also known as dialogs; they are practically equivalent to a telephone call made between two people.

Layer 4

There's two main players on this 4th layer. We have TCP and UDP. TCP is; Transmission Control Protocol and its all about "CONTROL"

On the other hand, UDP is called "User Datagram Protocol."

UDP "Welcome to networking and this is jackass" There's nothing secured and No control.

Proper Packet arrival by forming a connection, validating everything and closing it are made possible by TCP.

Layer 3

The routing layer or the third layer, here's where the IPv4, IPv6 and your router reside.

We won't discuss much about the router but we'll touch a few things that are needed to be said.

The IP is a numeric address which consists of 4 Octets between dots, each unique connection receives 1 IP address.

Why do I accentuate on "unique" Simply because there are ways?

Layer 2+1

The Data-link links.

The Physical layer is all about 1's and 0's.

INTRO TO DHCP

DHCP, otherwise known as Dynamic Host Configuration Protocol, is a server where an IP address and network configuration are dynamically assigned to a client or computer. When using a dynamic IP address, it's beneficial to have a portable device like a laptop because dynamic IP addresses can be changed. DHCP servers are very useful because they allow you to easily configure your machine and it can even account for network addressing changes automatically which can save a lot of time.

DHCP CLIENT SERVER COMMUNICATION

DHCP can be a broad topic, it is very important that you understand the general messages that are used throughout the process. Only then you can gain a full understanding of how DHCP works.

Some DHCP messages include:

DHCPDISCOVER - This is an initial message that marks the beginning of a DHCP interaction, and is sent by a client which is connected to a local subnet. The client uses 255.255.255.255 as a destination IP address, and uses 0.0.0.0 as a source IP address.

DHCPOFFER – The DHCP server sends this message as a response to the initial DHCPDISCOVER message. The client requests the network configuration settings from the DHCP server and this message contains those settings.

DHCPACK – Once the client has received the DHCPOFFER message, the DHCP server will send out this message a confirmation and acknowledgment by the server. The client will then be able to use the network configuration.

DHCPNAK – While the DHCPACK was to acknowledge that the server could satisfy the client's request, this message is only sent by the DHCP server to a client if the server cannot satisfy the DHCPREQUEST message sent by the client.

DHCPDECLINE – This message will only be sent to the DHCP server from the DHCP client if that client discovers that the IP address is already being used.

DHCPINFORM – This is a message that is sent from the DHCP client to the DHCP server. It occurs when the IP address being used is already statically setup on the client computer. It also tells the server that the network settings and configuration are only to be received from the DHCP server itself.

DHCPRELEASE – The DHCP client will send this message to the DHCP server when it wants to terminate the lease of the network address it was given by the server to start with.

HOW DHCP WORKS

Firstly, when the client's device is turned on or even just connected to the network, the DHCPDISCOVER message will be sent from the DHCP client to the recipient which is the DHCP server. The client won't have the network configuration at this time, so instead it sends 255.255.255.255 as a destination address and 0.0.0.0 as a source address for the server so that it can contact the client. Ideally, the DHCP server will be connected to the local subnet, this means that it would directly receive the message that was sent from the client. However, if it isn't connected to the local subnet, then a relay agent that is connected on the client's subnet can transfer the DHCPDISCOVER message over to the DHCP server via another subnet. This uses a transport protocol called UDP and the port for this is port 67.

Once the DHCP server receives that DHCPDISCOVER message from the client, it will respond with the DHCPOFFER message. The server sends this as a broadcast message which is to the 255.255.255.255 address so that the client can receive it directly. However, if the server isn't connected to the same subnet as the client, it will have to use the relay agent which was talked about in the previous part to transfer the message to the appropriate subnet. If it does need to use the relay agent, it will use the UDP protocol when transferring the message. This will be done on port 68.

Now that the client has received the DHCPOFFER message from the server, it will respond with the DHCPREQUEST message to indicate that it wants to accept the network configuration that was received from the DHCPOFFER message. This DHCPREQUEST message will still be using 0.0.0.0 as the source address because the client has not been privileged to the IP address that was offered in the DHCPOFFER message yet.

The server will now have received the DHCPREQUEST message from the client so it will respond with a DHCPACK message. This message acknowledges that the server has given access to the client to use the IP address that was assigned to it.

THE LEASE CONCEPT

Within the DHCP process, the server basically leases the client an IP address. The lease does expire, however, the server can give the client the same IP address as it was given the first time. The lease expires when the client shuts down their device but then when the device is turned back on, the server simply re-leases it so it gets the same IP address as it had the first time.

DHCP RESOURCE EXHAUSTION

This is a type of attack which is done by broadcasting large numbers of DHCP requests with lots of spoofed MAC addresses all at once. You can carry out this attack using tools like Gobbler. However,

the severity of this attack is rather low.

If a lot of the requests that are sent are flooded, the attacker could exhaust the address space given by the DHCP servers from an unlimited amount of time.

Starving DHCP addresses can also make a Rogue DHCP Server a lot more efficient too.

DHCP ROUGUE ATTACK

It's possible for an attacker to setup a Rogue DHCP Server so that they can gain access to network traffic. This is done by spoofing responses that are usually sent by authorized DHCP servers only. The attacker can then view the contents of the packets he picks up. Due to the attack being rather complicated, the severity is quite high.

Attackers are not always successful with this attack because it's a race between the legitimate DHCP server and the Rogue DHCP server, so it's all about the speed and luck you have with your server.

To stop an attack like this, the administrators need to constantly scan their traffic for fraudulent DHCP offer packets coming from anywhere that isn't the proper DHCP server.

Pronouncer - August 2, 2017, 4:21 pm

NAT

Dynamic NAT solves a factor that limits the number of devices connected to the internet. It's the number of IP Addresses available (Attributed Layer 3). Static NAT gives every device a different IP.

An IP Address (v4) has 4 bytes, so 32 bits. It can take 2^{32} Values, approximately 4 billion, and there are 7 billion people on earth.

And here comes the solution...

Dynamic NAT

Dynamic NAT gives every device connected to the same router a private IP (It often starts with either 192.168.x.x or 10.x.x.x) and a public IP to the router. Where 82.238.22.47 is your Public IP, it represents your local network, 255.255.255.0 represents your submask. 192.168.0.1 and 192.168.0.2 are two devices on your network.

The router knows how to correctly send packets because of the NAT Table. The NAT Table works the exact same way as ARP or Routing tables work, so there's no need for extra explanation.

The RFC1918 stands that all Private IP's should start with either 192.168.x.x or 10.x.x.x

Below is a table showing the use of IP Blocks in 1993, 2000 and 2007. Each number stands for all the IP's that have this number as a first byte (Field 52 represents IP's that begin with 52.X.X.X)

DHCP

We have learned that an IP identifies as a Device on a Network, and that it's needed to communicate with every device in the network, but how are IP's assigned?

The DHCP (Dynamic Host Configuration Protocol) Server's job is to supply IP Addresses (And Masks) to devices asking for it. If your network board is configured to automatically fetch it's IP, your device will (try to) contact a DHCP server on your local network.

The device contacts the DHCP server with an Ethernet Frame (As it can't do it any other way, because it doesn't have an IP yet), it uses a broadcast address (ff:ff:ff:ff:ff:ff), because of that, the DHCP server has to be on the same network as the device that is requesting an IP. There are ways that allow routers to relay these specific frames.

SO HOW DOES IT WORK?

1. The Device sends the Ethernet frame through a broadcast (DHCPDISCOVER)
2. The Server responds with a DHCPOFFER
3. The Device, through the same broadcast, replies with a DHCPREQUEST (This means that the device received and accepted the offer)
4. The Server then validates the DHCP lease and attributes the IP (This is a lease because this attribution expires.)
5. Once the lease expires, the device has to ask for an IP again. But this time it only broadcasts a DHCPREQUEST, while skipping the DHCPDISCOVER.

Networking Topologies

In Computer Networking, Topology refers to the layout of connected devices. Network topology is the settlement of the assorted elements of a computer network. Essentially, it is the topological structure of a network and may be depicted physically or logically

Bus Topology

Bus networks (not to be confused with the system bus of a computer) use a common backbone to connect all devices

Tree Topology

A tree topology joins multiple star topologies together onto a bus. In its simplest form, only hub devices connect directly to the tree bus, and each hub functions as the root of a tree of devices.

Ring Topology

In a ring network, every device has exactly two neighbors for communication purposes. All messages travel through a ring in the same direction.

Mesh Topology

Mesh topology introduces the concept of routes. Unlike each of the previous topologies, messages sent on a mesh network can take any of several possible paths from source to destination.

Star Topology

Many home networks use the star topology. A star network features a central connection point called a »hub node« that may be a network hub, switch or router.

LAN

A Local Area Network is a network confined to one building or site. Often a LAN is a private network belonging to an organization or business. Because LANs are geographically small, they usually use cables or low-power radio (wireless) for the connections.

WLAN

A Wireless LAN (WLAN) is a LAN that uses radio signals (WiFi) to connect computers instead of cables. At the centre of the WLAN is a wireless switch or a router – a small box with one or two antennas sticking out the back – used for sending and receiving data to computers connected to it. Most laptops have a wireless antenna built into the case. It is much more convenient to use wireless connections instead of running long wires all over a building. WLAN's are, however, more difficult to secure since other people can also try to connect to the wireless network. It is very important to have a good, hard to guess password for the WLAN connections.

WAN

A Wide Area Network (WAN) is a network that extends over a large area. A WAN is often created by joining several Local Area Networks together, such as when a business that has offices in different countries links the office LANs together. Because WANs are often geographically spread over large areas and links between computers are over long distances, they often use exotic connection technologies such as optical fibre cables, satellite radio links, microwave radio links and more.

Server VS Client Processing

Pros of client side:

Client side, user interactions can be managed

Reduces server resource usage (e.g. doing transformations on client, fewer round trips to server)

advantages of processing on server:

making code cross-browser compatible is a non-issue

No need to let clients see all the data

Data logging is simply where you use a computer to collect data.

Data logging is most commonly used for analysis of science experiments and monitoring systems when you need to collect data really fast over long periods of time.

Positive:

It can be used in malicious or non-malicious ways

It can be done at any time constantly

It can be done regularly

It can be fully automated.

Data logging is more accurate than humans who are prone to error

Negatives:

Malfunctions can cause corruption or loss of data

IT can get expensive.

Data logging must be done carefully and precisely or else it may be prone to bad readings.

Other Uses:

It can be used to store processor temperatures and memory usage percentages as well as bandwidth usage over a period of time.

It also allows staff to perform analysis over who was accessing systems information and is capable of assessing audit trails in order to trace malicious software and identify sketchy processes. Normally data logging keeps track of events like as data size, modification dates and who was modifying the data.

Pronouncer - August 2, 2017, 4:22 pm

MULTITHREADING

Multithreading - when a processor executes multiple processes or threads simultaneously. whilst multiprocessing is when processes and threads utilize a single (or multiple) core.

multiprocessing systems have a large number of units; whereas multithreading utilizes a single core.

PROS:

- If a cache misses, other threads may utilize the unused resources
- If a thread is unable to utilize the computer resources, running another thread may allow the resources to be used.

Cons:

- Multiple threads can interfere with each other

MULTIPROCESSING

Having more than one processor on a computer. Can also refer to running multiple processes at once. More than one processor can run the same instructions at once.

Processor Coupling

Tightly coupled: multiple CPUs. central shared memory. memory hierarchy (local and shared).

Loosely coupled: multiple single or dual cored processors.

- Tightly coupled perform better and are smaller. However, they require more investment are likely to depreciate.

loosely coupled have the ability to run multiple Operating systems at once.

Kinds of Multiprocessing

Symmetric - when a CPU usage isn't equal system resources will be divided in various ways (non-uniform memory access, asymmetric, and clustered).

Pros:

Because of multiplicity, multiprocessing yields better performance. It also has better reliability.

Cons:

Multiprocessing setups need more investment and memory. They also require complex OS.

IPV4

IIPv4 addresses are created with 32 bits. IPv4 addresses are commonly displayed in dotted decimal format because it's not easy for us to read a string of 32 ones and zeros. So Each IP address has four decimal numbers separated by three dots, like this: 192.168.1.5.

Each decimal can be represented with eight bits (also called an octet). Four octets (or four sets of eight) add up to 32 bits.

If all bits are a one in any octet (1111 1111), the value is 255. This is important to remember because an IPv4 address cannot have any decimals greater than 255.

IPv4 uses three primary types of addressing when sending traffic. The types are unicast, broadcast, and multicast.

IPV6

IPv6 addresses are composed of eight 16-bit fields, for a total of 128 bits. The added fields and the larger address size result in an increase of 2^{96} (or 4 billion times 4 billion times 4 billion) available IP addresses in the IPv6 addressing scheme.

An IPv6 address is represented by hexadecimal numbers separated by colons. An example of a valid IPv6 address is FE22:00FF:002D:0000:0000:0000:3012:CCE3. Shorthand for IPv6 allows us to represent multiple fields of zeros by eliminating the extra zeros and adding a colon-

FE22:00FF:002D::0:3012:CCE3. This shorthand can only be used for one group of zeros, otherwise we won't know how many groups of zeros go where.

A third difference between the two types of IP addresses is that in IPv6, addresses can reflect the scope of a transmission's recipients—for example, a single node, a group, or a special kind of group- see unicast, multicast, and any cast addresses.

IPV4 + 6

4 sets of 8 bit octets-IPv4

32 bit address-IPv4

128 bit address-IPv6

Format is hexadecimal-IPv6

Short hand for IPv6 allows you to replace consecutive sets of 0's can omit leading 0's per section

Hexadecimal Character Set

4 groups of 4 binary digits-IPv6

IPv6 is used more and makes Internet Protocol easier. IPv4 isn't used as much as IPv6.

Types of DDOS

Slowloris

Developed by Robert "RSnake" Hansen, Slowloris is DDoS attack software that enables a single computer to take down a web server. Due the simple yet elegant nature of this attack, it requires minimal bandwidth to implement and affects the target server's web server only, with almost no side effects on other services and ports.

Slowloris has proven highly-effective against many popular types of web server software, including Apache 1.x and 2.x.

Slowloris can be made easier if you understand the HTTP protocol.

HTTP stands for Hyper Text Transfer Protocol and was made by Tim Berners-Lee around the 90s.

HTTP is an application layer protocol that works in a web browser and Web server software by default it uses port 80 and TCP to connect to the application layer.

Resource requests made using HTTP work by using a URL (Uniform Resource Locator).

HTTP is a stateless protocol, this means that every single connections is independent of each other.

If an HTTP request is not complete, or if the transfer rate is very low, the server keeps its resources busy waiting for the rest of the data.

Using this information someone with malicious intent has the ability to open multiple HTTP requests by not closing previous requests.

Rudy

Named after an album by Finish melodic death metal band Children of Bodom, R.U.D.Y. (short for R-U-Dead-Yet?) is a DoS tool used to execute slow-rate attacks (similar to Slowloris), which is implemented via long form field submissions.

Slow rate, Layer-7 DDoS attacks, also called “low and slow” attacks, attempt to open a relatively few connections to the targeted server or web site over a period of time, and leave the sessions open as long as possible.

Rudy attacks web applications by starvation of available sessions on the web server.

Eventually, the number and length of open sessions exhaust the target's resources, making it unavailable to legitimate traffic. Because low and slow attack traffic appears legitimate, these attacks often fly under the radar of traditional mitigation tools.

When data is sent by the user it goes over 1 or 2 packets to the server, it then handles the request, closes the connection and moves on to the next.

Someone can send data to the server but not by sending 1 or 2 packets, the data is broken into a lot of packets, each packet containing only 1 byte of data.

All requests are sent at different times to avoid the server stopping the connection, forcing it to complete the requests. In a short time Rudy can make several thousands of packages stopping the server from receiving requests.

ARME

A.R.M.E, also known as Apache Remote Memory Exhaustion or Apache Remote Denial of Service, is a memory based attack written in Pascal that makes use of the http header called »range«. Range is commonly used for partially downloading data like videos, streams and more.

For example, we're trying to watch a video, our client will request a specific byte range or multiple byte ranges from an object. By doing so it decreases the amount of traffic coming out of the webserver since it's not requesting the entire thing. As the Range header is a good thing (Which is why it's a part of the http spec) but the issue is that it has a few flaws. To begin, this http spec allows for overlapping and has no limit on the number of requested ranges. You will now see why this is bad and how a malicious user can abuse this.

First, let's see how a user would make a range request.

When we are for example trying to watch a video, we would make a range request to the apache server asking to load the content off an application handler (php, etc) then we ask it to return partial content followed by a response of multiple objects from the ranges we requested. This, for example, would request only a part of the data instead of the whole thing, making sure the bandwidth from the server is limited. When a malicious person uses A.R.M.E it executes a loop and puts together a range header with overlapping ranges. It then stitches the range into a http request and sends it via a net socket. When the server receives the request, it will attempt to complete the range request just like it would with a legit person. When Apache begins loading the object into its memory, it will have to keep loading new versions of the objects to satisfy every single byte range we requested previously. As a result, we create a memory fork. Now what happens next is that we keep the server process busy resulting in a process fork attack, keeping them, all stuck doing read operations.

POST & GET

GET and POST are two main request types used between the communication of an HTTP client

(Your browser) to the HTTP Server. Basically, GET is used to request data from the server in URL to display static data as a response. POST Submits data to the server, data that has to be processed, for example data coming out of an input field such as a contact form, login page ...

Now while flooding is a malicious user's goal is to overwhelm the server with many requests to drain its computing resources. The most effective of the two is POST flooding, simply because it requests parameters that trigger complex processing on the server like accessing a database. These take a lot of resources compared to the GET method, for this reason, POST based flooding is the best method of the two. GET is more common and can get the malicious user help from people that view an inline image from a popular site, when they do so, each one of them sends a GET request to the targeted server. Since it's more common, this can also be an effective method.

SSDP

SSDP or Simple Service Discovery Protocol is commonly used with Universal Plug And Play (UPNP) Devices. UPNP are networking protocols that make it possible for networking devices to find and connect to one another without the intervention of a user. SOAP or Simple Object Access Protocol delivers the control messages to the UPNP Devices. When a Malicious user launches an SSDP Reflection Attack he spoofs the target's IP and begins sending SOAP requests to available UPNP devices on the internet. As a result, the devices then send back responses to the same target's IP Address. Depending on how the request is crafted, the response could be amplified by 30 using one single request. According to statistics, about 80 million devices on the internet are vulnerable to this attack.

To make it simple; When a malicious user sends SOAP requests over SSDP to a large amount of UPNP services the responses that are amplified will be sent back to the Victims IP and result in the consumption of all available bandwidth on his machine.

Pronouncer - August 2, 2017, 4:22 pm

IP Spoofing

IP Spoofing, also referred to as Host Jacking or IP Address forgery is a method in which a malicious user hides his identity to gain access to a website, hijack browsers or gain access to secured networks. If a user interacts with a hijacked page, the hijacker can get sensitive information from the user. The hijacker can steal credit card information, SSNs, Emails and more, the list goes on and the possibilities are endless.

HOW TO SPOOF AN IP?

There are many Linux tools you can use to spoof your IP Address. You can tell the Linux system in which you're using to act like a router. By doing this, you can modify your own IP Address. To do it in a simpler way, you can connect to a VPN or use a chain of proxies.

IP SPOOFING METHODS

Non-Blind Spoofing: This attack takes place when the attacker is on the equal subnet as the victim. The biggest threat of spoofing this instance is session hijacking which is accomplished by corrupting the DataStream of a well-established connection.

Blind Spoofing: Is a more complex attack because the sequence and acknowledgment numbers are unattainable.

To find out how these attacks take place, you must check out the structure of the TCP/IP protocol

suite. An essential understanding of these network exchanges is crucial to the process.

Bounce Back Attack

Bounce-Back Attacks uses UDP services. Someone can spoof an IP packet resulting in the two services sending traffic as fast as possible until a packet is lost on the network. This will damage the resources and network bandwidth. Anything that uses UDP reply can be attacked, things such as routers and firewalls.

Three Way Handshake

If you wanted to access an image on a webpage for whatever reason the host would have a SYN packet which has information of the sequence number and maximum number of bytes in the packet. A three-way handshake is primarily used to create a TCP socket connection. It works when a client node sends a SYN data packet over an IP network to a server on the same or an external network. The objective of this packet is to ask if the server is open for new connections. The target server must have open ports that can accept and initiate new connections. When the server receives the SYN packet from the client node, it responds and returns a confirmation receipt the ACK packet or SYN/ACK packet. The client node receives the SYN/ACK from the server and responds with an ACK packet

Man In The Browser

Malicious code that has infected a browser. Code inserted into the browser can read, copy, and redistribute anything the user enters in a browser

The Man-in-the-Browser attack is the same approach as Man-in-the-middle attack, but in this case a Trojan Horse is used to intercept and manipulate calls between the main application's executable and its security mechanisms or libraries on-the-fly.

Man-In-The-Browser attacks are used through malware that infects an Internet Browser.

This attack can hide from anti-virus software which makes it dangerous.

Man-In-The-Browser can see information within the browser, since the browser has no encryption.

Because how Man-In-The-Browser works, many network level devices such as application firewalls IDS and IPS systems have a hard time detecting this attack since it occurs client side.

Man-In-The-Browser captures packets as they travel from one host to another with the intent of altering the contents of the packets is a form of which security concern.

Man-In-The-Browser Uses a Trojan horse program to catch and control the session between the browser and the security components or libraries.

Man-In-The-Browser attacks uses many functions and features within a browser.

This happened based on information gathered and what can be stolen.

Man-In-The-Browser can uses extensions which can be used to exploit the OS.

There is no clear way to protect against Man-In-The-Browser

ARP

ARP is also known as Address Resolution Protocol. This is used when we want to map an IP address to a physical address like a MAC address for instance. This physical address will be recognised by the local network. A table which is often called an ARP Cache will keep track of MAC address as well as the IP address that is connected to it.

How does ARP work

We have a gateway, and when a packet arrives at one of these, the gateway has to ask the ARP program to view its cache to check if the IP address matches up with a MAC address. If the program does find the MAC address, then it can simply send the package straight to the right computer. However, if the program doesn't find the MAC address associated with the IP address, then the ARP will send a request to all other computers on the LAN to see if they can find the needed MAC address. The ARP will then update its cache to include the MAC address so it can now send the package to the correct machine.

RARP

RARP is the abbreviation for Reversed Address Resolution Protocol. It's quite easy to see that this is the opposite of ARP. While ARP searches an IP address to find an associated MAC address, RARP uses a MAC address to find an associated IP address.

Pronouncer - August 2, 2017, 4:23 pm

Thank you

Congratulations on making it to the end of the thread. We would appreciate any feedback and any corrections if mistakes were made, we are not perfect.

You are now even closer to finishing how to unskid yourself. Knowing this information gives you the basis of going into networking and hacking. This may seem like "common sense" to most however just knowing something and not knowing how it works is a big issue.

We will release a cleaner PDF version of this with more photos and information. Thank you again.

Kind Regards

-- Secrecy

Credits

Thread Contents:

Great White Shark

Travellers

SysAdmin

Socket

Molino

James

Shm3tzi

Lads

Pronouncer

(There is much more as this was sourced of an old group project, you know who you are we don't mean to miss you our or take credit for your contributions)

Formula1 : Thank you so much for allowing us to use your sub forum to create this thread without being disturbed in the process.

Sources

<https://en.wikipedia.org/>

Sonder

<https://www.youtube.com/user/elithecomputerguy>

<https://www.youtube.com/watch?v=xpXhudbsrr8>

<https://www.youtube.com/watch?v=t2mRNxfHTjw&list=PL7134FC0815ADB8EB>

Certified ethical hacker v9
Comptia Networking
and a shit tone of google

Thank you for taking the time to read this thread. We appreciate your effort and look forward to helping you out. If you are interested in joining us please read this thread.

<https://hackforums.net/showthread.php?tid=5673489>

sm?ked - August 2, 2017, 5:30 pm

Great job guys! So glad we could put this together!

Dr. Gonzo - August 2, 2017, 5:31 pm

This is the big one we all worked on! Glad to be a part of this group and I'm glad to be putting out content and helping the community and new members grow!

Zeke - August 2, 2017, 5:32 pm

Wow. So much info. It's great. Thanks for this informative tutorial. It's fun how unofficial groups are putting more effort into #makinghfgreatagain than official groups lol.

SysAdmin - August 2, 2017, 5:32 pm

I'm glad we could put this together in a time span such as this, i hope everyone enjoys the read!

Jake???????????????? - August 2, 2017, 5:34 pm

(Aug 2, 2017 05:32 PM)virgin2 Wrote: [CITE](#) Wow. So much info. It's great. Thanks for this informative tutorial. It's fun how unofficial groups are putting more effort into #makinghfgreatagain than official groups lol. Literally. I thought groups were here to make things great but all I've seen is they are meant for Userbar notoriety.

Rhinoo - August 2, 2017, 5:34 pm

you are really pumping out these HQ tuts

Travelers - August 2, 2017, 5:36 pm

Glad to be a part of this group and be able to contribute! ?

Fire - August 2, 2017, 5:40 pm

Interesting read, even though I know many would skip directly to the DDoS part, but the fundamentals are a must.

Xing - August 2, 2017, 5:45 pm

This took team effort, glad i was able to help!

ida - August 2, 2017, 6:15 pm

Great thread you guys, I have finally learned how to Distributed Denial of Service ?

Thank you!

Pronouncer - August 2, 2017, 6:15 pm

(Aug 2, 2017 06:15 PM)amid Wrote: [\[CITE\]](#) Great thread you guys, I have finally learned how to Distributed Denial of Service ?

Thank you!
Ddos the world buddy

Dr. Gonzo - August 2, 2017, 6:15 pm

(Aug 2, 2017 05:32 PM)virgin2 Wrote: [\[CITE\]](#) Wow. So much info. It's great. Thanks for this informative tutorial. It's fun how unofficial groups are putting more effort into #makingHfGreatAgain than official groups lol.

Deadass my dude. not throwing shade at anyone but your profile says it all about the current state of "makingHfGreatAgain.

(Aug 2, 2017 05:40 PM)Jaeger Wrote: [\[CITE\]](#) Interesting read, even though I know many would skip directly to the DDoS part, but the fundamentals are a must.

That particular demographic will thin itself out by getting arrested quickly because they didn't understand networks properly before they started doing illegal shit.

Pronouncer - August 2, 2017, 6:18 pm

(Aug 2, 2017 06:15 PM)Great White Shark Wrote: [\[CITE\]](#) Deadass my dude. not throwing shade at anyone but your profile says it all about the current state of "makingHfGreatAgain.

That particular demographic will thin itself out by getting arrested quickly because they didn't understand networks properly before they started doing illegal shit.
Stop bad shark, don't roast

Dr. Gonzo - August 2, 2017, 6:22 pm

(Aug 2, 2017 06:18 PM)Pronouncer Wrote: [\[CITE\]](#) Stop bad shark, don't roast
I would never ? ?

mutybyte - August 2, 2017, 9:30 pm

Superb superb and just superb tutorial. I didn't read entire but i will read this entire and will learn many things. I learned some new things from this post till i read. Rest will read soon.

LD_PRELOAD - August 3, 2017, 2:57 am

Quote:The next device we will be covering is the Router. The router allows you to separate different networks (Similar to if you're a tenant in a building and they offer internet connection). They will have to separate the network so that tenants cannot take advantage of each other. Each tenant is on their own network but they will all be connected to the same modem. Routers are used logically to separate networks.

Switches can also "seperate" connections on VLAN settings. It's not the main function of a router. The main function of a router is to route traffic.

Quote:also prevent internet traffic that you don't want. Firewalls are mostly a security device built into your modem or router. If you turn it on, it will block the service for the entire network, for example, if you block port 80 on the firewall, nobody in that network will be able to access the internet.

That's incorrect on multiple levels.

- 1) You won't be able to access services binding on port 80. That's not "the internet". You're already connected to the internet via the protocols you've discussed before such as DSL.
- 2) If you block outgoing packets with srcport 80, you're not blocking anything. Outgoing packets out of your PC have a randomized port which the OS provides. No packet will choose 80. Also most

traffic nowadays is 443 at any point.

Quote:Source / Port Dest./ Port / Total Length / Checksum / Data

That's not clear at all. What is "Source / Port Dest./ Port" ?

UDP has 4 headers. Source port, Destination Port, Length, Checksum, that's it. What's so complex?

Quote:5. PSH and URG -> Connection has Priority.

No.

if URG is set, the application will then utilize the Urgent Pointer (in the TCP headers) to point towards a offset specifying an important bit of data that must be read first. The connection does not have a priority. Application level errors might utilize this.

if PSH is set, the application will then receive the message as usual however will not hold things at the TCP buffer for further message if present, but push that information straight to the application as soon as it got it, so the application can work with it.

Quote:The most effective of the two is POST flooding, simply because it requests parameters that trigger complex processing on the server like accessing a database.

What do you think happens if the url looks like this?

<http://website.com/forum/messages/1>

You need to move on with your line of thought. RESTful APIs are basically everywhere now. You're not pentesting websites anymore, you're pentesting web services.

This tutorial alone does not stand well on it's own. Needs more work. In my opinion making a "networking tutorial" thread is a big of a long shot, how about making a series covering smaller stuff in better detail? You can't summerize networking on 3 posts, not without this many errors. The effort is good and well appreciated though.

SysAdmin - August 3, 2017, 3:03 am

(Aug 3, 2017 02:57 AM)LD_PRELOAD Wrote: [\[CITE\]](#) Switches can also "seperate" connections on VLAN settings. It's not the main function of a router. The main function of a router is to route traffic.

That's incorrect on multiple levels.

1) You won't be able to access services binding on port 80. That's not "the internet". You're already connected to the internet via the protocols you've discussed before such as DSL.

2) If you block outgoing packets with srcport 80, you're not blocking anything. Outgoing packets out of your PC have a randomized port which the OS provides. No packet will choose 80. Also most traffic nowadays is 443 at any point.

That's not clear at all. What is "Source / Port Dest./ Port" ?

UDP has 4 headers. Source port, Destination Port, Length, Checksum, that's it. What's so complex?

No.

if URG is set, the application will then utilize the Urgent Pointer (in the TCP headers) to point towards a offset specifying an important bit of data that must be read first. The connection does not have a priority. Application level errors might utilize this.

if PSH is set, the application will then receive the message as usual however will not hold things at the TCP buffer for further message if present, but push that information straight to the application as soon as it got it, so the application can work with it.

What do you think happens if the url looks like this?

<http://website.com/forum/messages/1>

You need to move on with your line of thought. RESTful APIs are basically everywhere now. You're not pentesting websites anymore, you're pentesting web services.

This tutorial alone does not stand well on it's own. Needs more work. In my opinion making a "networking tutorial" thread is a big of a long shot, how about making a series covering smaller stuff in better detail? You can't summerize networking on 3 posts, not without this many errors. The effort is good and well appreciated though.

While your effort is appreciated in correcting us, a cleaner, fixed version of a .pdf will be released when we have enough time to do so. Thanks for the corrections, we'll be sure to edit and correct any problems that are at fault before releasing the PDF File.

ida - August 3, 2017, 3:21 am

(Aug 3, 2017 02:57 AM)LD_PRELOAD Wrote: [\[CITE\]](#) What do you think happens if the url looks like this?

<http://website.com/forum/messages/1>

You need to move on with your line of thought. RESTful APIs are basically everywhere now. You're not pentesting websites anymore, you're pentesting web services.

What does that URL have to do with anything? POST parameters and values are sent in the body. Not sure what your point is.

LD_PRELOAD - August 3, 2017, 3:38 am

(Aug 3, 2017 03:21 AM)amid Wrote: [\[CITE\]](#) What does that URL have to do with anything? POST parameters and values are sent in the body. Not sure what your point is. Come to think of it I messed up the example and you're right. I was attempting to say that you can easily pass information in GET method and cause the same level of complexity and input being handled server side, although my example had little to do with it. Didn't like how strict the original wording was, there are alternative ways to pass information that sometimes happen on GET as well and can behave like POST. One common example I have is Protobuf on Google Maps. That's entire structures, dictionaries and key:value pairs that are being passed on GET.

ida - August 3, 2017, 4:05 am

(Aug 3, 2017 03:38 AM)LD_PRELOAD Wrote: [\[CITE\]](#) Come to think of it I messed up the example and you're right. I was attempting to say that you can easily pass information in GET method and cause the same level of complexity and input being handled server side, although my example had little to do with it. Didn't like how strict the original wording was, there are alternative ways to pass information that sometimes happen on GET as well and can behave like POST. One common example I have is Protobuf on Google Maps. That's entire structures, dictionaries and key:value pairs that are being passed on GET.

Oh, alright then. Yeah, there are a few explanations that could use a rewrite and you already pointed most of them out it seems.

However, I don't know if I agree with you entirely. Sure GET requests can be complex but I don't see why or how they could be near POST.

POST implementations include parameters that usually triggers some kind of complex processing on the server. Not that GET doesn't do this, but not to the same extent, as the use cases of GET are limited to serving simpler requests for retrieving information.

Unless of course you have a bad REST implementation, but that is an entirely different issue.

Pronouncer - August 3, 2017, 4:54 am

Please on the sections that are wrong and what you want it to say instead

I meant pm me the incorrect and the new correct

Pronouncer - September 17, 2017, 2:41 pm

Bumping this as it is still a great read, look out for our student scheme

Skull-13 - September 18, 2017, 8:50 pm

I really have read all this but I still got to watch all the videos. Great and neat job.

Pronouncer - September 21, 2017, 12:13 pm

Still a good read for beginners to be honest

Young Yeezy - September 25, 2017, 1:26 pm

Thanks! This really helped me out!

arvi - October 2, 2017, 1:26 pm

When is the Part 3 coming very excited for it.

@Pronouncer

Please add me to your Student program i have already filled in the details HF UID:3703743

I am a newbie in this field ready to learn and explore as i am very much passionate about hacking and pentesting, how the internet works, how computers communicate(as in this thread).....

Please Guide me I need help

CyberStrike - October 2, 2017, 4:17 pm

Excellent post well done.

arvi - October 21, 2017, 1:24 pm

(Sep 21, 2017 12:13 PM)Pronouncer Wrote: [\[CITE\]](#) Still a good read for beginners to be honest hey pronouncer you did PM me the link for the discord group but i missed the time limit of 24 hours can you please send it to me again.

unknowhero - October 27, 2017, 11:08 am

helped me a lot . thank you

Ashish882 - October 28, 2017, 5:43 am

Thanks you so interested thread you has devoted too much time in this thanks again

Pronouncer - December 23, 2017, 10:10 am

Still a good read for you guys out there

Spaark - February 3, 2019, 6:08 pm

Posting to come back later. Looks great but a bit too much to read right now :)

Voxi - February 4, 2019, 8:40 am

Very detailed, actually might help me with my first years exam :)

ActivateSeven - February 4, 2019, 8:51 am

Great job guys! So glad we could put this together!