Edit	View Go Capture Ana	lyze Statistics	4							🚄 Wireshark - Coloring Rules ARP		×
	Сору		Wireshark - Configuration Profiles			(vie	w Go Capture Analyze		etistics lelephony		-	
Q.	Find Packet	Ctrl+F				~	Main Toolbar			Name	Filter	
	Find Next Find Previous	Child	Search for profile			1	Filter Toolbar			ARP Request	arp.opcode==1	
		COI+IN								ARP Reply	arp.opcode==2	
		Ctrl+B	Profile		Type Default	~	Status Bar			HSRP State Change	harpistate III 8 888 harpistate III 16	/
	Mark Blownak Dashat	Child	Default							Spanning free topology Chang	e stp.type == 0:80	/
	mark/Unmark Packet	COI+M	A90		Personal	~ ~ ~	Full Screen	F11	ICMP errors	icmp.type.eq.3 icmp.type.eq.4 icmp.type.eq.5 icmp.type	eq 11 icmpv6.ty	
	Mark All Displayed	Ctrl+Shift+M	DUCD							ARP	arp	
	Unmark All Displayed	Ctrl+Alt+M	UNCP		Personal		Packet List			ICMP	icmp icmpv6	
	Next Mark	Ctrl+Shift+N	Bluetooth		Global		Packet Details			SCTP ABORT	sctp.chunk twee of ABORT	
	Demoisture Manek	Childhall	Classic				Darket Duter			TTL low or unexpected	(1 ip.dst == 224.0.0.0/4 & & ip.ttl < 5 & & ipim & & lospf) (ip	dst == 224.0.0.0/
	revious men. 50	CON+Jenne+D	No Reassembly		Global	-	Packetbytes			Checksum Errors	eth.fcs.status=="Bad" ip.checksum.status=="Bad" tcp.che	cksum.status==*
	Ignore/Unignore Packet	Ctrl+D					Packet Diagram			E ME	smb nbss nbns netbios	
	Janara All Diceland	Child Shift a D								DCERPC	dcerpc	
	ignore Mit Displayes	CON+Jamie+D					Time Display Format		,	Routing	hsrp eigrp ospf bgp cdp vrrp carp gvrp igrnp i	imp
	Unignore All Displayed	gnore All Displayed Ctri+Adt+D					Name Resolution		,	TCP SYN/FIN	tcp.fiags & 0x02 tcp.flags.fin == 1	
	Catelland Tone Beforence	04.7									tcp udo	
	Set/Unset time Reference	C01+1					Zoom		,	Broadcast	eth[0] & 1	_
	Unset All Time References	Ctrl+Alt+T								System Event	systemd_journal sysdig	
	Next Time Reference	Ctrl+Alt+N					Expand Subtrees		Shift+Right			
	Previous Time Reference	Ctrl+Alt+B					Collapse Subtrees		Shift+Left			
	Time Shift	Ctrl+Shift+T					Expand All		Ctrl+Right			
	Packet Comments	•					Collapse All		Ctrl+Left	¢		>
			+ - 9	taiRoamingiWireshark		-			Double click to edit. Drag to more. Rules are,	enceased in order until a match is finand.		
	Delete All Packet Comments					Help	Colorize Packet List		+ - R 🐻	C:[UnersineRAccData]Roaming[Winashark]profi	haj#iFicolorfitura	
	Configuration Profiles	Ctrl+Shift+A	OK	Import • Export • Cano	sei Help		Coloring Rules			OK	Copy from V Cancel Import Export	Help

To get to profiles in Wireshark, click Edit and Configuration Profiles. Inside the Configuration Profiles window we can choose profile for current use. We can also add, import, or export profiles. On the bottom of the window is the path that personal profiles are stored. I would not suggest altering the Default Profile, always make a new one. You can find Coloring Rules under View menu. Coloring rules can help you identify specific traffic in a profile by site. Always leave the Bad TCP filter on top.

A KP byairection.pcapng – L X												
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help												
🚄 🔳 🔬 🛞 📙 🔚 🔀 🔂 🔍 👄 🖻) 🕾 T 🕹 📃 🗏 Q Q	Q. 🎹	6	S								
arp 1 Opcode Regiver Opcode Regiver												
No. Time Delta Time Source	e Destination	Protocol	Length Info	Opcode	^							
6 0.749658 0.000000 Ubiq 24 3.514575 2.764917 Ubiq 52 6.265989 2.751414 Ubiq 53 7.290021 1.024032 Ubiq 81 12.956141 5.666120 VWwa 82 12.962868 0.006727 ARRIS	<pre>iti_20:20:94 Broadcast iti_20:20:94 Broadcast iti_20:20:94 Broadcast uiti_20:20:94 Broadcast uiti_20:20:94 Broadcast re_6a:70:44 ARNISGro_6c: SGro_6c:68:64 VMware_6a:70</pre>	ARP ARP ARP 68:dd ARP 9:4d ARP	60 Who has 192.168.0.102? Tell 192.168.0.198 60 Who has 192.168.0.102? Tell 192.168.0.198 60 Who has 192.168.0.102? Tell 192.168.0.198 60 Who has 192.168.0.102? Tell 192.168.0.198 42 Who has 192.168.0.1? Tell 192.168.0.106 60 192.168.0.1 is at 6c:63:9c:6c:68:dd	request request request request request request								
Expand Subtrees Collapse Subtrees Expand All Collapse All Apply as Column Ctrl+Shift+1	Destinati 20120:94 Broadc _a6:d8:39 Broadc _a6:d8:39 Broadc 20120:94 Broadc _a6:d8:39 Broadc _a6:d8:39 Broadc											
3 Apply as Filter	Apply as Filter: arp	lf you no	f you notice above in the Display Filter box (1), we									
Prepare as Filter Conversation Filter Colorize with Filter Follow	Selected 4 Not Selected and Selected or Selected	have limited the traffic for this capture to only ARP traffic. It is easy to apply filter by selecting a packet of interest from traffic window above and right										
Сору	and not Selected											
Show Packet Bytes Ctrl+Shift+O Export Packet Bytes Ctrl+Shift+X	or not Selected 254 192.16 254 192.16	clicking a protocol like (2) ARP on the left and choosing (3) Apply as Filter and (4) Selected, You										
Wiki Protocol Page Filter Field Reference Protocol Preferences	.254 192.16 0.106 13.107 0.106 13.107 0.106 13.107 0.106 13.107	can easily add a display filter button (5) for commonly used filters, by clicking the + (6) and										
Decode As Ctrl+Shift+U Go to Linked Packet 2 Show Linked Packet in New Window Address Resolution Protocol (request)	s captured (480 bits) :20:20:94), Dst: Broac	giving th	ving the button a Label Name and click Okay.									

Core Network Protocols: UDP, TLS, IPv6, DNS, ARP, IP, ICMP. These protocols make up the majority of the traffic on your network. They are the most useful in finding issues on your network.

Address Resolution Protocol (ARP) bridges layer 2 and layer 3. ARP helps devices on the network keep track of which Media Access Control (MAC) addresses have which IP addresses, in order to complete their packets and send. ARP request are broadcast and that is why you usually see it anywhere you do a capture. ARP reply is unicast and usually only the requesting client sees those packets. You can identify scanning or unwanted traffic with ARP if one machine is sending ARP request for every IP address on the

network. Most clients don't do peer-to-peer traffic, so finding machines that are requesting IP addresses of other workstations is a good indication of infection propagating or malicious activity.

Devices keep an ARP table that it can reference, so that it doesn't have to do an ARP request every time it needs to communicate with another device. Devices will keep the most recent ARP reply that they receive. So if two devices reply to a request, the last reply that it gets will be the one it keeps.

Internet Protocol (IP) is the network layer communication method. It allows end to end communication rather than point to point like Ethernet. IP Header Structure:



(1) Version-Usually 4

(2) Internet Header Length-Length of IP header in 4-byte (32-bit) units.

(3) Differentiated Services Codepoint (DSCP)- How a packet priority is marked. QoS uses this Byte to prioritize traffic. Explicit Congestion Notification (ECN) is an optional feature used by ECN-aware devices to notify devices of network congestion to avoid dropped packets. Standard machine traffic is Non ECN-Capable Transport (Non-ECT).

(4) Total Length- Length of the whole packet including the header in bytes.

(5) Identification- 16-bit number in each packet to help the destination host reassemble packet fragments.

(6) Flags- Only first 3 bits are defined. Bit 1 is reserved and always 0. Bit 2 tells if the packet can be fragmented. Bit 3 is set to 0 if it is the last fragment and 1 if more fragments follow. Fragment offset tells how many 8-byte blocks are in the packet fragment.

(7) Time to Live (TTL)- Number of seconds a packet can take to reach the destination. Routers decrease this value by a preconfigured amount, usually 1 and discard packets that arrive with field set to 0. Most networks interpret this field as a simple hop count between routers. TTL is usually set as 255, 128, or 64.So if a ping TTL return value is 60, we can be pretty sure that there are 4 hops to the destination.(8) Protocol: Tells what protocol section is next and Wireshark uses this field to determine which dissector to load next. TCP, ICMP, UDP, etc.

(9) Header Checksum- Used to verify that the header has not been altered.

Internet Control Message Protocol (ICMP): An ICMP packet pictured above (10), we can see it is not used just for ping. It can be used to send information about network traffic and alerts about network problems. It can send returns to devices for routing issues, port problems, network congestion, or TTL issues. If you watch a capture for ICMP return packet on Network Unreachable, it can give you a place to

start looking for potential problems. It will tell you if your traffic can't be routed because your packet needs to be fragmented.

User Datagram Protocol (UDP): Used for time-sensitive traffic like voice, video, and DNS lookups. By not establishing a connection like TCP, UDP is able to speed up data transfer. Because it doesn't establish a connection before sending data it is less reliable than TCP. If a datagram is lost in transit, there is no way for the sending device to know to resend it. So when you have a scratchy VOIP phone call you are experiencing datagram loss, but your conversation is not delayed like if it were being sent over TCP. This also makes UDP more susceptible to DDoS attacks. Attackers use the lack of TCP handshake to flood a target with UDP traffic and causes the denial-of-service for real traffic.



Dynamic Host Configuration Protocol (DHCP) is how we assign addresses dynamically to devices on the network from a single location. The DHCP procedure consist mainly of four packets. (1) The Discover packet is broadcast on the devices local segment to everyone. In the packet the device can ask for different options like DNS server, subnet mask, and default gateway. (2) The DHCP server responds with an offer from its ip address pool. (3) The device responds with a request that is broadcast again to everyone, so that if there were offers from other servers on the network, they know which offer was accepted. (4) The server responds with an acknowledgment and makes the IP official. The device then broadcast an ARP message with its MAC address and new IP address. This is how devices determine if they have a duplicate address. If it doesn't get a response that another device has that IP address, then it broadcast that it now has that IP address.

Domain Name System (DNS) is a naming system where names are resolved to the associated IP addresses and vice versa. It is similar to a phonebook, but for the internet or a local network. DNS is pretty simple it sends a request and the request is forwarded until a server has a cache of the requested name or it gets to an authoritative name server for that domain to be resolved and a response is sent back with the IP address for that domain name.

File Transfer Protocol (FTP) is a standard internet protocol used to transfer data from one device to another. The model it is built on, one devices acts as a server and the other devices as a client. It works really well with one big downfall, it is not a secure protocol and using Wireshark we can see login and password passed in plain text. Secure File Transfer Protocol (SFTP) addresses this issue by encrypting the login and transfer by a secure channel via Secure Shell (SSH). The other popular method of file transfer is Trivial File Transfer Protocol (TFTP) and is used mainly on a local network over UDP to push configuration information to devices like switches and VOIP phones. Hypertext Transfer Protocol Secure (HTTPS) or the secure version of HTTP using Secure Socket Layer (SSL) / Transport Layer Security (TLS). TLS is an updated and more secure version of SSL. TLS 1.3 is the most common method currently for encryptions. Some older websites might still use TLS 1.2. TLS uses mathematically generated keys that the client and server agree on to encrypt and decrypt data to be exchanged. The client sends a public private key pair that initiates the handshake and tells the server which cipher it is capable of, as well as the TLS version. The server responds with a generated random key, the cipher that it chose, and data encrypted with the key. With one round trip communication both sides can now encrypt and decrypt data being sent. Below we can seem an example of a Client Hello and the response of Server Hello. Everything after this is scrambled, because it is encrypted.

e

ïme	Source	Destination	Protocol	Length	Info		Time	Source	Destination	Protocol	Length Info)		
.934754	192.168.0.105	13.107.21.200	TLS01.2	543	Client	Hello	6.977549	13.107.21.200	192.168.0.106	TLS01.2	1250 Seru	er Hello, Certificate, Certificate Status, Server Key Box		
 Transmo 	rt Laver Security						 Transpo Tis 	rt Layer Security	Handshake Protocol: Mult	inle Handshak	e Mettaret			
V TLS	w1.2 Record Laver: Ha	andshake Protocol: Client				100	Content Type: Hand	shake (22)	apac narraanan	e i kanalien				
	Content Type: Hands	hake (22)				Version: TLS 1.2 (8+8383)							
	Version: TLS 1.2 (8x8383) Length: 484							Length: 7831						
								Handshake Type	: Server Hello (2)					
~	Handshake Protocol: Client Hello							Length: 98						
Handshake Type: Client Hello (1)								Version: TLS 1	2 (0:49383)			_		
	Length: 480	(-/						> Random: 616da3 Session ID Len	66f1b466fd5b744ea6fe98a7 ath: 32	1106975da3cba	icee7a2c51e18df	9464126		
	Version: TLS 1.	2 (0:49393)						Session ID: 94	109999dc8af495b43f364aed	2676f983a448b	abf491d2bef91d	id269f85ee59		
	> Random: 616da36	6bde90ab2964c0326768bee56	875eabd125c	534587155	c2782f56	861e		5 Cipher Suite:	TLS_ECOHE_RSA_WITH_AES_2	56_0CM_SHA384	(8x0939)			
	Session ID Leng	th: 32						Compression Me	thod: null (0)					
	Session ID: ae2	eeeee5e592cd3f425cd1b53b3	fe1ca@de648	:46a53ae2	ee8e11ff	622903aa		> Extension: star	gun: 26 tus request (len≠0)					
	Cipher Suites L	ength: 38						> Extension: ses	sion_ticket (len⇒0)					
	✓ Cipher Suites (:	' Cipher Suites (19 Suites) Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_6CM_SHA384 (0x002c)					 Extension: application_layer_protocol_negotiation (len=5) 							
	Cipher Suit						Type: application_layer_protocol_negotiation (16)							
	L Cipher Suit	e: TLS_ECOHE_ECOSA_WITH_A	ES_128_GC/1_:	SHA256 (8	x002b)			ALPN Exten	sion Length: 3					
	Cipher Suit	e: TLS_ECOHE_RSA_NETH_AES	_256_GCM_SH	4384 (Өкс	0 3 0)			GY ALPN Proto	col _					
	Cipher Suit	e: TLS_BCDHE_RSA_WITH_AES	_128_GC/1_SH	4256 (Øxo	02f)			ALPN 5	tring length: 2					
	Cipher Suit	e: TLS_BCDHE_BCDSA_WITH_A	ES_256_CBC_5	SHA384 (e	к с 924)			> Extension: ed	ext Protocol: n2 ended master secret (len	∋ 9)				
	Cipher Suit	e: TLS_ECDHE_ECDSA_WITH_A	ES_128_CBC_5	SHA256 (8	×0923)			> Extension: ren	egotiation_info (len=1)	-/				
	Cipher Suit	e: TLS_ECDHE_RSA_WITH_AES	_256_CBC_SH	4384 (Өкс	928)			[3435 Fullstri	ng: 771,49288,5-35-16-23	-65281]				
	Cipher Suit	e: TLS_ECDHE_RSA_WITH_AES	_128_CBC_SH	4256 (Өжс	027)		~	[3435: a66ea56 Handshake Protoco]	8599a2f5c89eec8c3a0d69ce : Certificate	e]				
	Cipher Suit	e: TLS_BCDHE_BCDSA_WITH_A	ES_256_CBC_5	SHA (BOKOB	0a)			Handshake Type	: Certificate (11)					
	Cipher Suit	e: TLS_ECDHE_ECDSA_WITH_A	ES_128_CBC_5	SHA (exce	69)			Length: 4886						
	Cipher Suit	e: TLS_ECDHE_RSA_WITH_AES	_256_CBC_SH	4 (excel4)			Certificates L	ength: 4803 4897 heter)					
	Cipher Suit	e: TLS_ECDHE_RSA_WITH_AES	_128_CBC_SH	4 (0xc013	0			 Certificat 	e Length: 3423					
	Cipher Suit	e: TLS_RSA_NITH_AES_256_G	Ст_5НАЗВ4 (6	6×9999d)				7 > Certificat	e: 38828d5b38828b43a8838	2818282137f88	1963acd57a78ac	477809510000001963ac300d. (id-at-commonName≍www.bing.com)		
	Cipher Suit	e: TLS_RSA_WITH_AES_128_G	СM_SHA256 (е	8x999c)				Certificat	e Length: 1374					
	Cipher Suit	e: TLS_RSA_NETH_AES_256_C	BC_SHA256 (0	8×12993d)			~	Bandshake Protocol	e: 308205583082044280030 : Certificate Status	2818282188787	4722C53088C89T	58907017904838300066928. (10-80-00MMDANAMe="10"05070 KS4		
	Cipner Suit	e: TLS_RSA_WITH_AES_128_C	BC_SH4256 (6	\$92993C)				Handshake Type	: Certificate Status (22)				
	Cipher Suit	e: TLS_RSA_WLTH_AES_256_C	BC_SHA (BOABA THE CIVE (COLOR	435) NAG				Length: 1746						
	Cipher Suit	e: TLS_KSA_WLTH_ACS_120_C	.DC_SNA (BOBB	92T) 200-)				Certificate St	atus Type: OCSP (1) Leonathi 1747					
	Composition Wat	e: ils_ksa_klin_bucs_cuc_ bode Longth, 4	CDC_SNA (888	999a)				> OCSP Response	cengent 1742					
	 Compression Meth 	hods (1 method)					~	Handshake Protocol	: Server Key Exchange					
	But engines Lengt	hous (I mechou) Fhy 360						Handshake Type	: Server Key Exchange (1	2)				
	 Extension: serul 	er name (len=17)						✓ BC Diffie-Hell	man Server Params					
	Type: serve	с пате (8)						Curve Type	: named_curve (0x03)					
	Length: 17							Named Curv	e: secp384r1 (@@018)					
	Server Name	Indication extension						Pubkey Len	gtn: 97 e45c8f4e1a998face3ce86af	Sf3hbc1bb7538	9c2774754c1d4c	9544158/587798ff/a66		
	Server I	Name list length: 15						✓ Signature .	Algorithm: rsa_pss_rsae_	sha256 (0x200	4)			
	Server I	Name Type: host_name (0)						9 Signat	ure Hash Algorithm Hash:	Unknown (8)				
	Server I	Name length: 12						Signat Signature	ure Hash Algorithm Signa Length: 256	ture: SM2 (4)				
	Z Server I	Name: www.bing.com						Signature:	02e239b7991dc69aa566b82	115e1c26f a96 b	2e8ecf63bb7b2a	ie62618575d3d6c3b91da8c		
	 Extension: signat 	ure_algorithms (len=26)												
	Type: signatu	we_arBourtume (12)												
	Signature Has	h Algorithms Length: 24							Clienter	m d c -	المللة	o with (1) Ciphore that		
	🛛 Signature Has	h Algorithms (12 algorithm	(3					ine	e client se	inus a	Hello	5 with (1) Ciphers that		
	3 > Signature	Algorithm: rsa_pss_rsae_s	:ha256 (0x000	4)						14/-		· • • • • (2) • • • • • • • • • • •		
	🥣 > Signature	Algorithm: rsa_pss_rsae_s	:ha384 (0×080	5)				can	support.	wea	can se	e the (2) server this		
	> Signature	Algorithm: rsa_pss_rsae_s	:ha512 (0≫080	6)							1/2			
	> signature	Algorithm: rsa_pkcs1_sha2	55 (8%8481) 84 (8%8581)					Hel	lo is going	g to a	nd (3) Signature Hash that it		
	> signature	Aleorithm: rsa_pkcs1_snap	∿⊶ (enecen) (enecen)						0 0	-	. `			
	> Signature	Algorithm: ecdsa secp256r	1_sha256 (@v	8483)				can	support.	It de	clares	s the (4) http version		
	> cimmtum													

Signature Algorithm: Goda_copyrat_Isha324 (0x850) Signature Algorithm: ecdsa_secp324r1_sha324 (0x8503) Signature Algorithm: SH41 DSA (0x8203)

Signature Algorithm: rsa_pkcs1_sha512 (@x Signature Algorithm: ecdsa_secp521r1_sha512 (@v@603)

Extension: application_layer_protocol_negotiation (len=14)
 Type: application_layer_protocol_negotiation (16)

Extension: session_ticket (len=269)
 Type: session_ticket (35)

Length: 269

Length: 14 ALPN Extension Length: 12 ✓ ALPN Protocol ALPN string length: 2 ALPN Next Protocol: h2 ALPN string length: 8 ALPN Next Protocol: http/1.1

Data (260 bytes)

it can support. We can see the (2) server this Hello is going to and (3) Signature Hash that it can support. It declares the (4) http version that it can support.

The server responds with the chosen (5) Cipher, (6) http version, (7) Certificate, (8) Public Key, and (9) Hash.