Simulacre Prova final UF2 Planificació i administració de xarxes

Preguntes teòriques relacionades amb el Nivell 7 del model TCP/IP:

1.-a) ¿Quina diferència principal hi ha entres els protocols HTTP i HTTPS? ¿En quin port escolta per defecte un servidor de cada tipus? En aquest sentit, ¿per a què serveix el protocol TLS (antic SSL) i entre quins dos nivells TCP/IP se situa?

b) ¿Per a què serveix principalment el protocol SSH? ¿I el VNC? ¿En quin port escolta per defecte un servidor de cada tipus?

c) ¿Per a què serveixen, respectivament, els protocols SMTP i POP (o IMAP)? ¿En quin port escolta per defecte un servidor de cada tipus? ¿En quin port escolten per defecte els servidors SMTPS, POPS i IMAPS?

d) Digues dos protocols del nivell d'aplicació que funcionin sobre UDP i digues en quin port (UDP) escolten els servidors respectius per defecte.

e) ¿Quina informació proporciona l'arxiu "/etc/services"?

2.-a) ¿Quin tipus d'informació en general transmeten les capçaleres HTTP de petició? Posa'n un exemple

b) ¿Quin tipus d'informació en general transmeten les capçaleres HTTP de resposta? Posa'n un exemple

c) ¿Què significa que la capçalera HTTP de resposta "Content-Type" valgui "image/png"? ¿Què significa que la capçalera de petició HTTP POST "Content-Type" (en aquest cas dóna la casualitat que hi ha una capçalera de petició i una altra de resposta homònimes, però no és l'habitual) valgui "text/xml"? ¿Què són, en general, els tipus MIME?

d) Si un servidor web retorna el codi de resposta 200, ¿qué significa? ¿I el codi 403? ¿I el 501? ¿I el 301?

e) ¿Per a què serveix el mètode HTTP anomenat DELETE?

f) Si tinguessis funcionant un servidor HTTP (Apache, per exemple) al port 22 i des d'una altra màquina volguessis connectar-t'hi fent servir el client SSH, ¿què creus que passaria? Per què?

3.-a) ¿Què li passaria a un ordinador si té una IP, màscara i porta d'enllaç però no té cap DNS assignat?

b) ¿Quina informació emmagatzemen els 14 servidors DNS "root" del món? ¿I els diferents servidors DNS "TLD"? ¿Per què com a usuaris ens és igual assignar a la nostra màquina el servidor DNS 8.8.8.8 (de Google) o el 192.168.12.10 (del centre), entre molts altres? Digues la IP d'un altre servidor DNS públic qualsevol disponible

c) ¿Quin avantatge té posar en marxa a la nostra xarxa local un servidor DNS privat que sigui (només) rèplica d'un public? ¿I si el teu servidor DNS privat no fos rèplica de cap, tindria encara algun sentit implementar-ho? En aquest sentit, ¿què és un servidor DNS "autoritatiu" i un servidor DNS "de catxé"?

d) ¿Sempre es realitza una petició DNS abans d'executar qualsevol comanda/programa que faci ús de la xarxa (des d'un simple *ping* fins un navegador)?

e) Abans d'inventar-se el protocol DNS ja existia l'arxiu "/etc/hosts". ¿Per què creus que es va haver d'inventar el protocol DNS? O dit d'una altra manera: ¿quins problemes té haver de fer servir aquest arxiu per la resolució de tots els noms d'Internet?

f) El contingut de l'arxiu /etc/hosts d'un ordinador amb direcció IP (de classe B) 192.168.6.34 és:

127.0.0.1 localhost
192.168.6.35 pepe

I el contingut del mateix arxiu en un ordinador amb direcció IP (de classe B) 192.168.6.35 és:

127.0.0.1 localhost
192.168.6.34 pepe

¿Què passarà si s'executa la comanda *ping -c 4 -i 1 pepe* al 1r ordinador? ¿I si s'executa al 2n?

HTTP

4.-a) Després de consultar a https://www.boredapi.com/documentation la documentació de l'API "Bored", digues com faries servir la comanda *curl* contra el seu servidor per mostrar al terminal només (és a dir, hauràs de fer neteja amb *jq*) la descripció d'una activitat aleatòria que sigui de franc i només per una sola persona

> NOTA: Un bon tutorial de *jq* el pots trobar aquí: http://www.compciv.org/recipes/cli/jq-for-parsing-json , però recorda que tens disponible un pdf en exclusiva on es mostren les seves opcions més importants

aII) Digues com faries servir la comanda *ncat* per obtenir la mateixa resposta que amb el *curl* de l'apartat anterior.

> NOTA: Probablement caldrà que facis servir el paràmetre *-C* del *ncat* Aquest paràmetre fa que els "Enter" que pulsem interactivament mentre estem construint la petició no siguin interpretats com "\n" (que és el que passa per defecte) sinó com a "\r\n", que és la combinació de caràcter estàndar a l'HTTP per delimitar línies.

b) Després de consultar (a https://dog.ceo/dog-api/documentation) la documentació de l'API "Dog", omple amb el valor adient els forats (indicats en groc) en la següent combinació de comandes *eog $(curl -s xxxx | jq -r "yyyy")* per tal de mostrar a pantalla la foto d'un gos qualsevol triat a l'atzar.

c) Després de consultar (a https://ip-api.com/docs) la documentació de l'API "IP-API", escriu la comanda curl correcta per obtenir en format JSON la ciutat i país on s'ubica l'adreça IP 1.1.1.1 I també la comanda curl per obtenir la mateixa informació però en format CSV

d) Vés a https://webhook.site i observa la URL única que se t'ha generat. Tot seguit, després de consultar (a https://docs.webhook.site/api/examples.html) la documentació de la seva API, escriu la comanda curl correcta per enviar al seu servidor (via POST) la següent dada JSON: *{'a':'1','b':'2'}* i comprova, a la plana web de "Webhook.site", que efectivament hi hagi arribat

Ntfy.sh (https://ntfy.sh) és un servei online gratuït que funciona com un "broker" de missatges: a ell se li poden enviar peticions HTTP de tipus POST missatges d'un determinat tema (i amb un determinat contingut) als quals altres clients poden subscriure's (via una petició GET que roman oberta permanentment) per, a partir de llavors, rebre'ls de forma síncrona en el moment que el client publicador en faci un enviament al "broker".

> NOTA: El "broker" no cal que sigui la web "ntfy.sh" sinó que podem implementar-ne un "self-hosted" en els nostres propis servidors seguint els passos descrits a https://docs.ntfy.sh/install

5.-a) Després de consultar (a https://docs.ntfy.sh/subscribe/api) la documentació de l'API "Ntfy" relativa a la subscripció de missatges d'un determinat tema al seu "broker" online, escriu la comanda *curl* correcta per anar mostrant en temps real a la pantalla tots els missatges que vagi publicant (cada cop que s'executi) el client emissor de l'apartat següent sobre un tema concret

> NOTA: A l'igual que per fer enviaments, Ntfy proporciona diferents formes de subscriure's a un tema i, per tant, de visualitzar els missatges corresponents un cop rebut que no són amb la comanda curl (per exemple, via app de mòbil, via navegador integrat amb el sistema de notificacions de l'escriptori, via una aplicació de terminal específica, etc)

b) Després de consultar (a https://docs.ntfy.sh/publish) la documentació de l'API "Ntfy" relativa a l'enviament de missatges al seu "broker" online, escriu, en un altre terminal diferent de l'obert a l'anterior apartat (que ha de romandre obert), la comanda *curl* correcta per enviar un missatge (de tipus JSON o RAW, com vulguis) amb el tema que vulguis però amb el contingut "S.O.S estem en un vaixell fantasma".

6.-a) Consulta el manual de *curl* (i/o els apunts de classe) i digues amb les teves pròpies paraules:

-¿Per a què serveix el seu paràmetre *-F* i quin és el mètode HTTP que fa servir aquest paràmetre?
-¿Quin és el valor de la capçalera "Content-Type" que genera aquest paràmetre?
-¿Quina és la diferència entre usar els símbols "@" o "<" en indicar el valor d'aquest paràmetre?
-¿I si es combina algun dels dos símbols anteriors amb el símbol "-" ?
-¿Per a què serveix el paràmetre *-T* i quin és el mètode HTTP que fa servir?

b) A partir de la informació obtinguda a l'apartat anterior, dedueix què fan les següents comandes (les quals fan servir diferents serveis online similars excepte un...¿quin és?) i digues quina és la funció de cadascun dels paràmetres utilitzats (i el del seu valor indicat respectiu).

ps -e | curl -F "clbin=<-" https://clbin.com
curl -F "file=@a.yml" http://0x0.st
cat file.txt | curl -F "f:1=<-" http://ix.io
curl -F "f:1=@file.txt" http://ix.io
echo -n "pepito" | curl -F "-=<-" qrenco.de

DNS

7.-a) Digues tres formes diferents, totes mitjançant Systemd, de configurar els servidors DNS d'un sistema global o una tarja de xarxa. ¿Amb quina comanda podries confirmar quin dels eventuals múltiples servidors DNS configurats mitjançant aquestes diverses formes és el que efectivament té assignat una determinada tarja de xarxa?

b) Explica amb les teves pròpies paraules què significa el valor "Cache Hits" i el valor "Cache Misses" mostrats per la comanda *resolvectl statistics*

c) Utilitza la comanda dig per esbrinar, preguntant al servidor DNS configurat al teu sistema quina és la IPv4 associada al nom DNS "www.redtube.com" (és un lloc pornogràfic). ¿És diferent la resposta si realitzes la mateixa consulta però al servidor DNS 1.1.1.3? ¿Per què? D'altra banda, utilitza la comanda dig per esbrinar també quina és la IPv6 de "www.redtube.com"

d) Prova de fer una consulta DoT qualsevol al servidor 1.1.1.3 ¿És compatible aquest servidor amb aquest protocol? D'altra banda, ¿com hauríem de configurar el servei Systemd-resolved per a què per defecte utilitzés DoT en totes les seves consultes DNS (fetes per qualsevol programa del sistema)?

e) ¿Per a què serveix la línia de configuració *MulticastDNS=true* tant de l'arxiu "/etc/systemd/resolved.conf" com de l'arxiu "*.network" corresponent a una determinada tarja de xarxa, a més de la línia *LinkLocalAddressing=ipv4* d'aquest darrer arxiu també? ¿Què hi té a veure en tot plegat el contingut de l'arxiu "/etc/hostname"?

8.-a) En una màquina virtual (Ubuntu o Fedora, tant se val) que tingui almenys una tarja de xarxa en mode "adaptador pont", afegeix al seu arxiu "/etc/hosts" una línia que associï la IP 1.2.3.4 amb el nom "pepe.asix.com". Tot seguit, modifica la configuració del seu servei "Systemd-resolved" per tal de què escolti peticions provinents de la xarxa i per a què pugui actuar com a servidor autoritatiu de la "zona" definida a l'arxiu "/etc/hosts" local.

b) Executa a la màquina real la comanda *dig @ip.maq.virtual pepe.asix.com* ¿Què veus? ¿Per què, en canvi, la comanda *dig pepe.asix.com* no mostra res (o, el que és el mateix, ¿per què la comanda *ping pepe.asix.com* no funciona)? ¿Què hauries de fer llavors (suposant que tinguessis permisos de "root") per a què el servidor DNS implementat a la màquina virtual sigui l'utilitzat per defecte per la màquina real (mostra un exemple realista de configuració)?

9.-Configura en una màquina virtual qualsevol amb escriptori gràfic el programa AdGuardHome (vist als exercicis) per:

* Activar totes les llistes possibles de dominis sospitosos que AdGuardHome ofereixi a l'apartat de filtres "DNS blocklists" i, de pas, actualitza-les.
*Prohibir la navegació al domini "microsoft.com" (i tots els seus possibles subdominis) des de l'apartat de filtres "Custom filtering rules"
*Blocar Facebook sencer des de l'apartat de filtres "Blocked services"
Redireccionar totes les peticions dirigides a ".marca.com" per a què vagin a "elpuig.xeill.net", des de l'apartat de filtres "DNS rewrites"

A partir d'aquí:

a) Configura en aquesta mateixa màquina de forma temporal amb la comanda *resolvectl* el servidor DNS que s'emprarà per defecte per a què sigui el servidor AdGuardHome.

b) Obre el seu navegador, visita les següents pàgines web que s'indiquen, i observa què passa:

*Pàgina "www.elpais.com" o "www.hola.com" o qualsevol altra web que se t'acudeixi.
*Pàgina "gecko.me" o "liadm.com"
*Pàgina "microsoft.com"
*Pàgina "www.facebook.com" o "www.facebook.es" o "facebook.com" o similar
*Pàgina "www.marca.com" o qualsevol subdomini relacionat

WIRESHARK

10.-a) Obre amb el Wireshark el fitxer "exercici1.pcapng" penjat a la web del centre i respon les següents preguntes:

*****¿Quin és el servidor DNS que està configurat a la màquina 192.168.1.4 (d'on se suposa que s'ha obtingut aquesta captura? ¿Quin és el nom de la màquina remota amb qui està contactant tota l'estona (després del tràfic DNS)?

*****¿Quin és la possible comanda que ha generat tot aquest trànsit? Pista: fixa't a quin port van dirigits els tots els paquets SYN enviats (per veure-ho millor pots fer servir el filtre "tcp.flags.syn")

*****¿Quin és el rang de ports amb els que s'ha volgut connectar aquesta comanda? Digues quin filtre de pantalla has hagut d'utilitzar per veure-ho

*****¿Quin/s port/s de la màquina remota són els que responen a l'enviament de paquets SYN amb un paquet [SYN,ACK]? ¿Quina creus que és la raó per la què hi ha ports que no en responen?

*****¿Quin sentit té que la màquina d'on s'ha obtingut aquesta captura, en rebre una resposta [SYN,ACK] estàndar de la màquina remota, li envïi llavors un paquet de tipus RST?

b) Obre amb el Wireshark el fitxer "exercici2.pcapng" penjat a la web del centre. Sabent que la informació que et mostra aquesta captura ha sigut generada per una conversa entre un servidor Netcat i un client Netcat i que aquesta conversa ha consistit en l'enviament d'un fitxer de text des del servidor fins el client, respon les preguntes:

*****¿Per què els valors de les MACs d'origen i de destí són 00:00:00:00:00:00?

*****¿Quin és el contingut -en ASCII- del fitxer de text enviat? ¿Com ho has sabut?

*****Sabent que els diferents flags d'un paquet TCP qualsevol es troben agrupats en un bloc de 12 bits, ¿és cert o fals que el valor numèric -en decimal- d'aquest grup de 12 bits seria 17 si el paquet fos de tipus [FIN,ACK]? ¿Com ho has sabut?

c) Obre amb el Wireshark el fitxer "exercici3.pcapng" penjat a la web del centre i respon les següents preguntes:

*¿Quin és el navegador que utilitza la màquina 192.168.0.101 (d'on se suposa que s'ha obtingut la captura)?

*Aplica el filtre de pantalla "tcp.stream eq 0" i explica la funció de cadascun dels 10 paquets que formen aquest fluxe

*¿Per què les peticions HTTP/S es fan a la màquina "dictionary.map.fastly.net" quan a l'inici la màquina 192.168.0.101 ha fet una consulta DNS per obtenir la IP de la màquina "www.dictionary.com"?

*¿Quina és la direcció IPv6 de "dictionary.map.fastly.net" i com ho has averiguat?

*¿Què veus si apliques ara el filtre de pantalla "tcp.stream eq 1"?

*Digues el nº del/s paquet/s que contenen la paraula "Undertow", digues quines opcions del Wireshark has hagut d'utilitzar per esbrinar-ho i digues en quina part del/s paquet/s apareix aquesta paraula.

d) Obre amb el Wireshark el fitxer "exercici6.pcapng" penjat a la web del centre. Sabent que la informació que et mostra aquesta captura ha sigut generada per una "escombrada" de peticions ARP a la LAN, respon:

*¿Quina és la direcció IP de la màquina que ha fet l'"escombrada"? ¿Amb quina comanda la podria haver fet?

*¿Quin filtre de pantalla faries servir per saber quines màquines de la LAN han respost a aquesta "escombrada"? Digues les seves direccions IP

11.-a) Llegeix el següent article: https://www.infosecmatter.com/detecting-network-attacks-with-wireshark i digues quin filtre hauries d'utilitzar per detectar si estàs essent víctima d'un atac "ARP-Spoofing". D'altra banda, explica el significat del filtre que detecta els atacs "ICMP Flood", "TCP Null Scan" i "TCP FIN Scan" i digues l'eina amb què es fa cadascun

b) Explica el que es descriu en aquest article: https://shantoroy.com/security/operating-system-fingerprinting

TSHARK

12.-Digues quina informació (columna a columna) mostren les següents comandes *tshark* a la pantalla, i també una possible comanda que hauries d'executar per tal de generar el tràfic adient que es veiés filtrat en aquesta pantalla:

a) *tshark -Y "not arp" -T fields -e frame.time_epoch -e frame.protocols -e ip.src -e ip.dst -e udp.dstport -e _ws.col.Protocol*

b) *tshark -Y "dns and dns.flags.response == 1" -E header=y -T fields -e frame.time_epoch -e dns.qry.name -e dns.a*

NOTA: En el filtre cal especificar explícitament "*dns and ...*" per no tenir en compte el tràfic MDNS

c) *tshark -Y "http.request" -T fields -e frame.time_epoch -e ip.dst -e http.request.method -e http.request.uri -e http.user_agent*

cII) *tshark -Y "http.response" -T fields -E separator=, -e http.response.code -e http.content_length -e text*

d) *tshark -Y "http or dns" -O http,dns*

e) *tshark -q -c 500 -z http,tree*

eII) *tshark -q -c 500 -z plen,tree*

eIII) *tshark -q -c 500 -z dests,tree*

13.-a) Escriu la comanda Tshark adient per veure per pantalla només el tràfic que sigui generat des de la MAC de la teva tarja de xarxa i que vagi dirigit al port TCP 80

b) Amb la comanda Tshark anterior funcionant, obre un altre terminal i, amb la comanda *curl*, vés a la web http://www.chappeliu.com ¿Què passa? ¿Veus algun paquet a la sortida del Tshark? ¿Per què?

c) Tanca la comanda Tshark anterior i torna a escriure una de nova per veure només el tràfic generat des de la MAC de la teva tarja de xarxa igual però ara dirigit al port UDP 53. Amb aquesta comanda Tshark en marxa, en un altre terminal torna a utilitzar la comanda curl per tornar a intentar connectar amb la mateixa web http://www.chappeliu.com ¿Veus ara algun paquet a la sortida del Tshark? ¿Per què?

OPENSEARCH

14.-A partir d'executar l'script *./tshark-os.sh "ip" "frame ip udp tcp icmp tls http dns ssh"* (vist a classe), implementa tot el necessari per aconseguir tenir els següents gràfics visibles en un panell Dashboards:

a) Histograma que compti la quantitat de paquets detectats al llarg del temps, classificats per protocol (ICMP, TCP, UDP, TLS HTTP, DNS i SSH)

b) Histograma que compti la quantitat de respostes HTTP al llarg del temps, classificades pel seu codi

c) "Pie chart" que compti la quantitat de paquets detectats classificats per port de destí i que s'autoactualitzi cada 5 s

cII) "Pie chart" que compti la quantitat de paquets detectats classificats per IP d'origen i que s'autoactualitzi cada 5 s

cIII) "Pie chart" que compti la quantitat de paquets detectats classificats per protocol i que s'autoactualitzi cada 5 s

d) Taula que sumi la quantitat de bytes dels paquets detectats classificats per la combinació de les següents dades (mostrades cadascuna en sengles columnes): IP d'origen, IP de destí, port d'origen, port de destí, protocol i "Info"