Akademija tehničko-umetničkih strukovnih studija Beograd

Odsek Visoka škola za informacione i komunikacione tehnologije

**Cloud Sigurnost - AWS**

Bezbednost informacionih sistema

Natalija Vugdelija

Ilija Krstić 155/21

Đorđe Knežević 255/22

10.10.2023. Beograd

Table of Contents

[SHARED RESPONSIBILITY MODEL 3](#__RefHeading___Toc2367_1971068212)

[Opis 3](#__RefHeading___Toc2369_1971068212)

[Podela odgovornosti 3](#__RefHeading___Toc2375_1971068212)

[Primeri podele odgovornosti 4](#__RefHeading___Toc2377_1971068212)

[EC 2 (IaaS) 4](#__RefHeading___Toc2379_1971068212)

[Elastic Beanstalk (PaaS) 4](#__RefHeading___Toc136_2588892014)

[Workmail (SaaS) 4](#__RefHeading___Toc136_2588892014_Copy_1)

[AWS SECURITY BEST PRACTICES 5](#__RefHeading___Toc2367_1971068212_Copy_1)

[Opis 5](#__RefHeading___Toc2369_1971068212_Copy_1)

[Korišćenje root naloga 5](#__RefHeading___Toc2369_1971068212_Copy_2)

[Princip najmanjih privilegija 5](#__RefHeading___Toc2369_1971068212_Copy_3)

[Security grupe 5](#__RefHeading___Toc2369_1971068212_Copy_4)

[NACL 6](#__RefHeading___Toc586_790464588)

[DDOS I ZAŠTITA NA CLOUDU 7](#__RefHeading___Toc2367_1971068212_Copy_2)

[Opis 7](#__RefHeading___Toc2369_1971068212_Copy_5)

[DOS 7](#__RefHeading___Toc2369_1971068212_Copy_6)

[DDOS 7](#__RefHeading___Toc2369_1971068212_Copy_7)

[Podela po sloju 7](#__RefHeading___Toc2369_1971068212_Copy_6)

[Napadi na sloj infrastrukture 8](#__RefHeading___Toc2369_1971068212_Copy_8)

[UDP reflekcija 8](#__RefHeading___Toc2379_1971068212_Copy_1)

[SYN poplava 8](#__RefHeading___Toc2379_1971068212_Copy_2)

[SLIKE 9](#__RefHeading___Toc1383_2526964257)

[LITERATURA 10](#__RefHeading___Toc1385_2526964257)

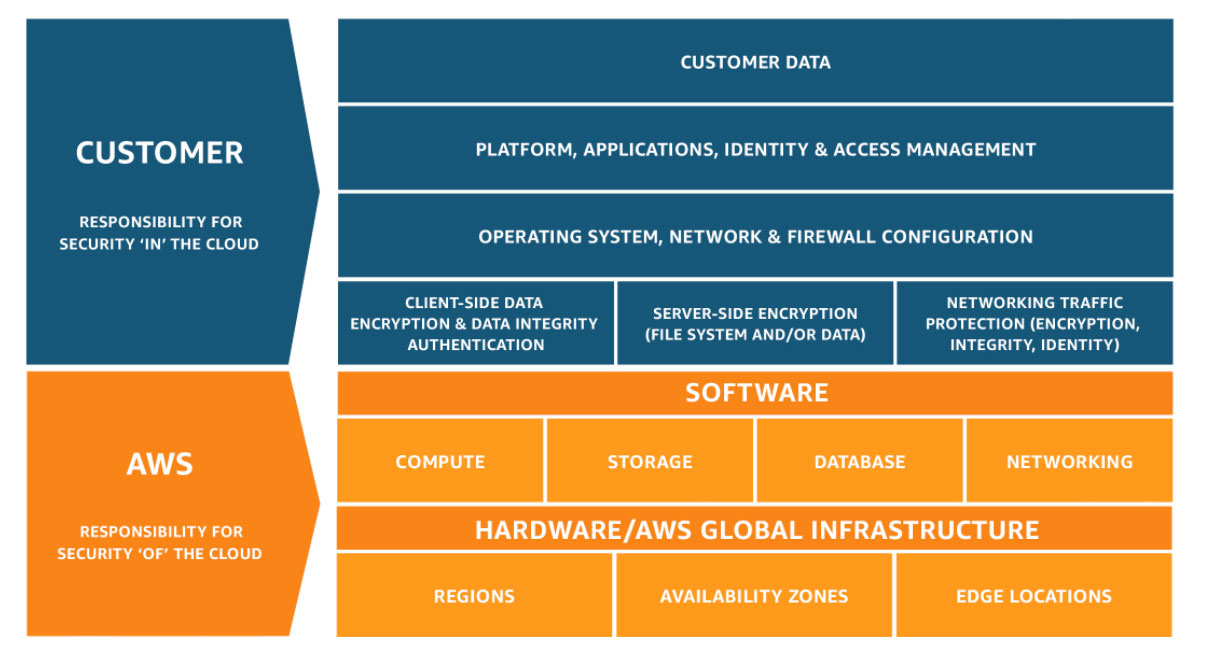
# SHARED RESPONSIBILITY MODEL

## Opis

Shared responsibility model je model po kojem se deli odgovornost za sigurnost (eng responsibility) između provajdera cloud usluga, u našem primeru amazona, i korisnika njihovih usluga i servisa.

Kako nije moguće da je više strana apsolutno odgovorna za sigurnost jedne stvari, odgovornosti se dele na manje delove, a nad delovima se stranama dodeljuje apsolutna odgovornost.

Amazon delove koji su rezultat podele grupišu kao sigurnost clouda (eng security of the cloud) i sigurnost u cloudu (eng security in the cloud) i preuzima odgovornost za prvu grupu i korisnicima predaje odgovornost za drugu[1].

Slika 1 - Opširna podela odgovornosti

## Podela odgovornosti

Podela odgovorosti često zavisi od specifičnog servisa o kome se radi, ali, najosnovije, cloud provider je odgovoran za zaštitu infrastrukture servisa koje nudi, čineći hardware, software, umrežavanje i objekte kod kojih se ovi servisi izvršavaju.

Odgovornost uveliko zavisi od vrste cloud servisa, od kojih postoje tri glavna[2].

Software as a service (SaaS) je model u kom provider pruža aplikaciju u cloudu korisniku za korišćenje, u ovakvom modelu korisnik preuzima najmanji mogući deo odgovornosti[2].

Platform as a service (PaaS) je model u kome se platforme mogu otkupiti od provajdera i koristiti za razvoj, hostovanje i upravljanje aplikacijama, u ovom modelu provider je odgovoran za hardware i software na kojem korisnici pokreću aplikacije, a korisnici su odgovorni za sigurnost aplikacija koje postavljaju na platformu[2].

Infrastructure as a service (IaaS) je model u kome provider pruža računarske kapacitete, tj komputacione resurse, korisniku, a on je odgovoran za sve što se na njima desi, samim tim, odgovoran je za operativni sistem i sve što dalje radi na njemu[2].

Minimalno, bez obzira na model, korisnik je uvek odgovoran za sigurnost podataka, kontrolu pristupa resursima i zakonitost sadržaja.

## Primeri podele odgovornosti

### EC 2 (IaaS)

Elastic Cloud Container je servis koji pruža amazon i koji prati IaaS model cloud servisa.

Amazon pruža razne vrste instanci koje predstavljaju komputacione resurse, optimizovane za različite namene, u formi virtualnih mašina.

Instance su hostovane na fizičkim mašinama, a podelu na virtualne mašine vrši hypervisor[3], sigurnost fizičkih mašina i potpuna razdvojenost između instanci su odgovornost providera.

Jedna instanca se inicijalizuje sa operativnim sistemom po izboru korisnika i od tada sve ostaje na korisniku.

### Elastic Beanstalk (PaaS)

Elastic beanstalk pruža mogućnost olakšanog pokretanja aplikacija na cloudu koji od korisnika zahteva samo otpremanje koda aplikacije koju korisnik želi da pokrene.

Amazon preuzima odgovornost za bezbednost hardware-a i software-a na kome se aplikacije pokreću, a korisnik za bezbednost aplikacija koje pokreće.

Elastic beanstalk takođe može pomoći sa dostupnošću aplikacija pokrenutih na njemu zato što nudi i automatsko skaliranje resursa aplikacija po uviđenoj potrebi, što takođe može smanjiti troškove kada potreba za resursima manja[4].

Ovaj servis takođe pri skaliranju resursa može koristiti više zona dostupnosti (eng availability zones), što osigurava dostupnost aplikacije čak i pri pretnjama koje bi mogle da onesposebe jednu[4].

### Workmail (SaaS)

Workmail nudi privatan mejl servis potpuno podešen od strane amazona.

Amazon preuzima odgovornost za bezbednost hardware-a i software-a koji omogućavaju ovaj servis, a korisniku ostaje odgovornost o podacima, kontroli pristupa, i zakonitosti sadržaja i podataka[5].

# AWS SECURITY BEST PRACTICES

## Opis

U ovom poglavlju ćemo u kratko proći nekoliko najboljih praksa (eng best practice) kada se radi o bezbednosti na AWS cloudu.

## Korišćenje root naloga

Amazon strogo preporučuje da se root nalog ne koristi za svakodnevne zadatke, već da se koristi za izvršavanje samo onih zadataka koje samo root nalog može da izvrši, a da se za sve ostale potrebe napravi i koristi administrativni korisnik[6].

Preporučen je visok nivo pažnje pri radu i zaštiti root naloga, te korišćenje unikatnih šifara i multifaktorske autentikacije[6].

Takođe je preporučljivo praćenje pristupa i korišćenja root naloga, Amazon Cloudwatch nas može obavestiti o tome pristupu na root nalog a AWS Cloudtrail nam omogućava da pratimo sve aktiovnosti korisnika[6].

## Princip najmanjih privilegija

Princip najmanjih privilegija, ili na engleskom “least privilege principle” je princip po kojem se korisnicima daju samo one privilegije bez kojih oni ne bi mogli da izvrše svoje dužnosti i ni jednu privilegiju više od tih[7].

Praćenjem ovog principa se uspešno možemo zaštititi od pristupa podacima ili sistemima od strane korisnika za kojeg ne nameravamo da njima ima pristup, smanjujemo rizik u slučaju kompromizovanja korisnika i smanjujemo mogućnost slučajnih grešaka.

Bitno je da pri praćenju ovog principa pažljivo procenimo kojem korisniku su koje privilegije potrebne kako ih ne bi sprečili u izvršavanju njihovih dužnosti, dok im istovremeno ne bi omogućili nenameren pristup.

## Security grupe

Security grupe kontrolišu mrežni pristup određenom resursu, a jedan može pripadati više security grupa.

Security grupe su stateful što znači da pamte zahteve, u slučaju da se pošalje zahtev sa EC2 instance odgovoru će uvek biti dopušteno da stigne do nje bez obzira na pravila security grupa u kojima je, i isto tako će dozvoljenom zahtevu uvek biti dozvoljen i odlazni zahtev za odgovor[8].

Preporučljivo je napraviti najmanji mogući broj security grupa, a da se one koriste za pristup resursima koje imaju slične funkcije i sigurnosne zahteve[8].

Pri dodavanju pravila za ulazne zahteve za port 22 (SSH), dobro je dozvoliti ga samo određenom rasponu ip adresa, kako se ne bi omogućio pristup resursu tek bilo kome[8].

## NACL

NACL je akronim za network access liste, slične su dosta već poznatim ACL listama koje se mogu naći na linux sistemima, umesto kontrole pristupa fajlovima i folderima, vrše kontrolu pristupa resursa određene podmreže.

Podmreža može u bilo kom trenutku da koristi samo jedan NACL, a jedan NACL se može koristiti od strane više podmreža[9].

Za razliku od security grupa, NACL su stateless, što znači da se podaci o zahtevima ne čuvaju, i da uspešan ulazni zahtev ne garantuje da će odlazni zahtev biti dozvoljen[9].

Preporučljivo je koristiti ih isključivo za zaustavljanje zahteva, a praviti ih na osnovu potreba i očekivanog ponašanja podmreže, te od svega ostalog se zaštititi.

# DDOS I ZAŠTITA NA CLOUDU

## Opis

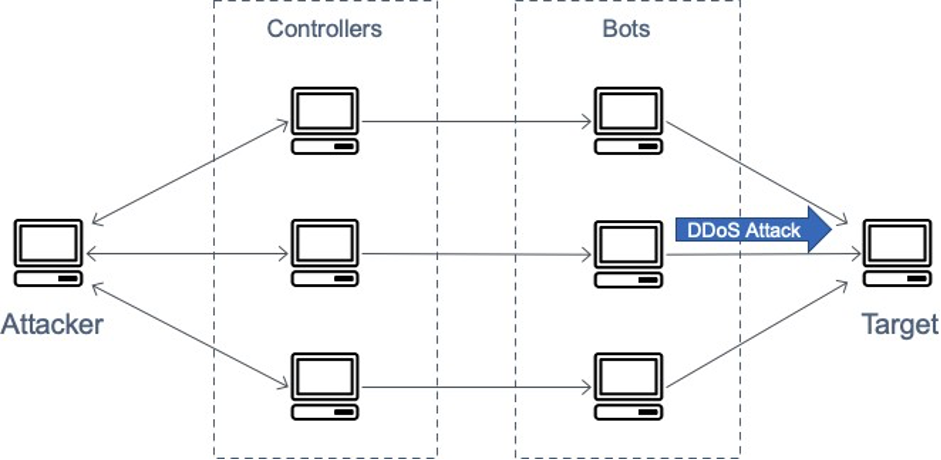
Postoji više vrsta distributed denial of service napada, u ovom poglavlju ćemo proći nekoliko njih, kao i načine zaštite.

## DOS

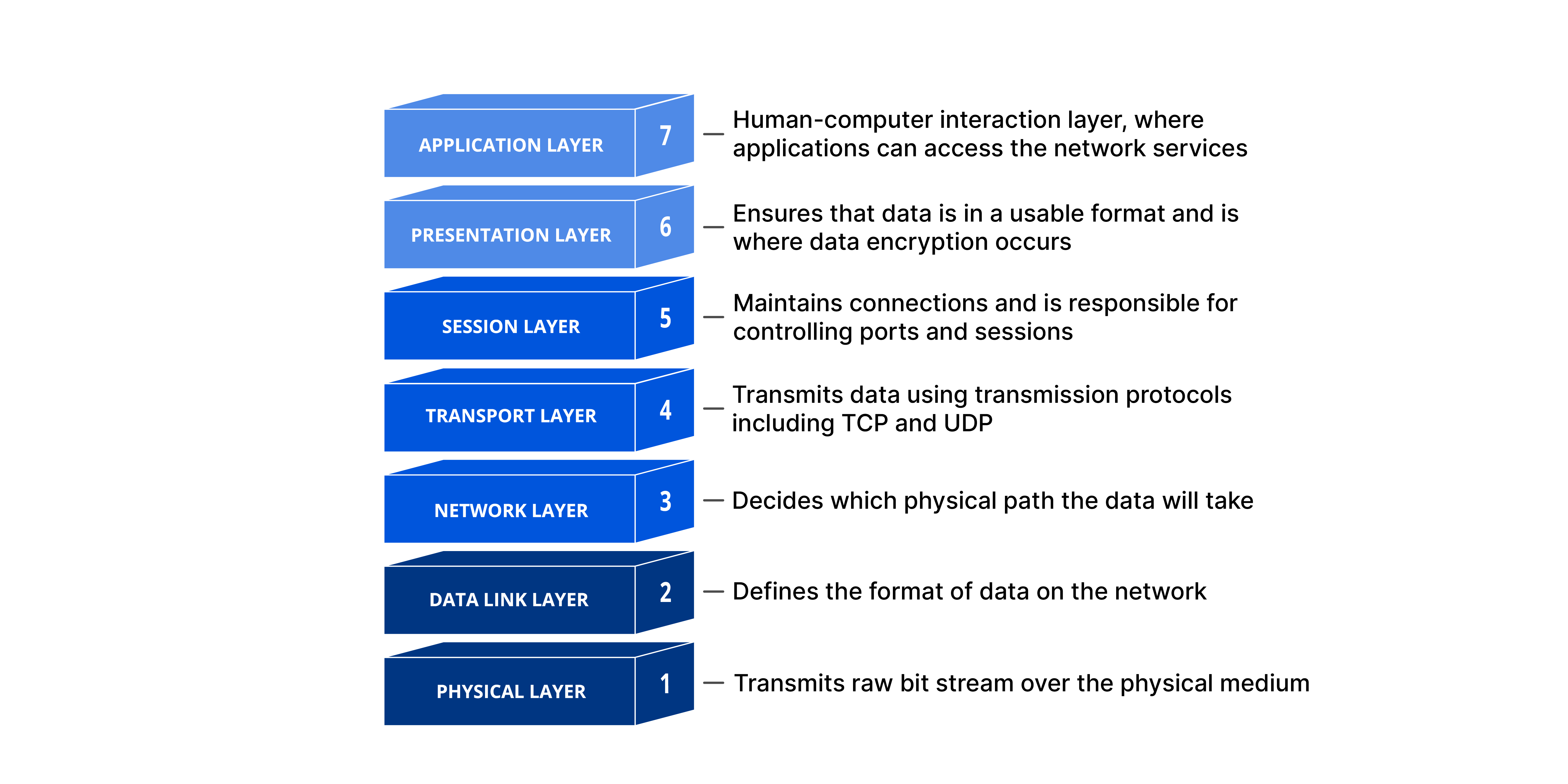
DOS (denial of service) napad je namerni pokušaj da se sajt ili aplikacija učini nedostupnom korisnicima, postoji više tehnika koje mogu trošiti visoke količine resursa, otežavajući pristup korisnicima[10].

## DDOS

DDOS (Distributed Denial of Service) ima isti cilj kao DOS, ali se za napad koristi više izvora koji mogu predstavljati kompromizovane računare, ruture i ostale uređaje povezane na mrežu[10].

Slika 2 - Diagram DDOS napada

## Podela po sloju

Slika 3 - Slojevi OSI modela

DDOS Napadi se mogu grupisati po sloju koji ciljaju[10], mi ćemo se baviti napadima na sloju infrastrukture, koji za naše potrebe predstavlja treći i četvrti sloj OSI modela, i na aplikacionom sloju, koji predstavlja šesti i sedmi sloj OSI modela.

## Napadi na sloj infrastrukture

Sluj infrastrukture je najčešća meta DDOS napada, ovi napadi generišu velike količine protoka sa ciljem da se sposobnosti ciljane mreže preplave ili da se određeni resursi poput servera ili firewalla drže zauzetim[11].

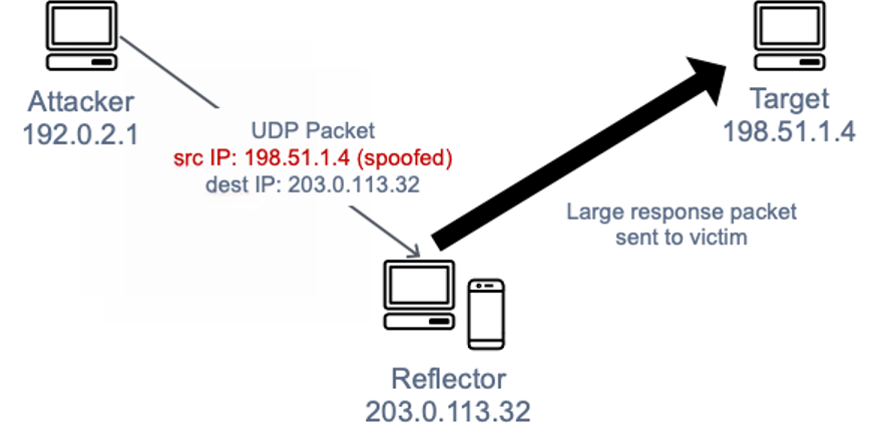
Dva najčešća napada na sloj infrastrukture su UDP reflekcija i poplava SYN paketima.

### UDP reflekcija

UDP reflekcija se služi činjenicom da je UDP stateless protokol, što, kao i za NACL, znači da se podaci o zahtevima ne čuvaju.

Napadači mogu da formiraju validan UDP paket, a da navedu IP adresu mete kao adresu izvora UDP paketa.

Falsifikovan paket se šalje od strane napadača posredovaćem serveru, koji onda svoj odgovor šalje meti umesto napadaču, posredovajući server se koristi jer je paket odgovora više puta veći nego paket zahteva, što značajno povećava protok poslat meti.

Slika 4 - Diagram UDP reflekcije

### SYN poplava

# SLIKE

(1) Amazon, dostupno na https://aws.amazon.com/compliance/shared-responsibility-model/

(2) Amazon, dostupno na https://docs.aws.amazon.com/whitepapers/latest/aws-best-practices-ddos-resiliency/introduction-denial-of-service-attacks.html

(3) Cloudflare, dostupno na https://www.cloudflare.com/learning/ddos/glossary/open-systems-interconnection-model-osi/

(4) Amazon, dostupno na <https://docs.aws.amazon.com/whitepapers/latest/aws-best-practices-ddos-resiliency/udp-reflection-attacks.html>

# LITERATURA

[1] Amazon – Shared responsibility model.

**Dostupno** na <https://aws.amazon.com/compliance/shared-responsibility-model/>\

[2] Crowdstrike – **Shared Responsibility Model.**

**Dostupno na** https://www.crowdstrike.com/cybersecurity-101/cloud-security/shared-responsibility-model/

**[3] Amazon – What is a hypervisor?**

**Dostupno na** https://aws.amazon.com/what-is/hypervisor/

**[4] Amazon – Elastic Beanstalk features.**

**Dostupno na** https://aws.amazon.com/elasticbeanstalk/details/

**[5] Amazon – Workmail.**

**Dostupno na** https://aws.amazon.com/workmail/

**[6] Amazon -** **Root user best practices for your AWS account**

**Dostupno na** <https://docs.aws.amazon.com/IAM/latest/UserGuide/root-user-best-practices.html>

**[7] Wikipedia – Principle of least privilege**

**Dostupno na** https://en.wikipedia.org/wiki/Principle\_of\_least\_privilege

**[8] Amazon – VPC Security grupe**

**Dostupno na** https://docs.aws.amazon.com/vpc/latest/userguide/vpc-security-groups.html

**[9] Amazon – Control trraffic to subnets using NACLs**

**Dostupno na** https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html

[10] Amazon – Introduction to denial of service attacks

Dostupno na <https://docs.aws.amazon.com/whitepapers/latest/aws-best-practices-ddos-resiliency/introduction-denial-of-service-attacks.html>

**[11] Amazon – Infrastructure layer attacks**

**Dostupno na** https://docs.aws.amazon.com/whitepapers/latest/aws-best-practices-ddos-resiliency/infrastructure-layer-attacks.html

**[12] Amazon –** **UDP reflection attacks**

**Dostupno na** <https://docs.aws.amazon.com/whitepapers/latest/aws-best-practices-ddos-resiliency/udp-reflection-attacks.html>