# Analiza alata i metoda za ispitivanje sigurnosti

**Šostarec, Marta**

**Master's thesis / Diplomski rad**

**2021**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Organization and Informatics / Sveučilište u Zagrebu, Fakultet organizacije i informatike**

*Permanent link / Trajna poveznica:* https://urn.nsk.hr/urn:nbn:hr:211:996599

*Rights / Prava:* Attribution 3.0 Unported/Imenovanje 3.0

*Download date / Datum preuzimanja:* **2024-03-29**

SVEUČILIŠTE U ZAGREBU
FAKULTET ORGANIZACIJE I INFORMATIKE
VARAŽDIN

*Repository / Repozitorij:*

Faculty of Organization and Informatics - Digital Repository

zir.nsk.hr

DIGITALNI AKADEMSKI ARHIVI I REPOZITORIJI

**SVEUČILIŠTE U ZAGREBU**

**FAKULTET ORGANIZACIJE I INFORMATIKE**

**V A R A Ž D I N**

**Marta Šostarec**

# Analiza alata i metoda za ispitivanje sigurnosti

**DIPLOMSKI RAD**

**Varaždin, 2021.**

**SVEUČILIŠTE U ZAGREBU**

**FAKULTET ORGANIZACIJE I INFORMATIKE**

**V A R A Ž D I N**

**Marta Šostarec**

**Studij: Informacijsko i programsko inžinjerstvo**

# Analiza alata i metoda za ispitivanje sigurnosti

**DIPLOMSKI RAD**

**Mentor/Mentorica:**

Doc. dr. sc. Petra Grd

**Varaždin, rujan 2021.**

*Marta Šostarec*

**Izjava o izvornosti**

Izjavljujem da je moj završni/diplomski rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

*Autor/Autorica potvrdio/potvrdila prihvaćanjem odredbi u sustavu FOI-radovi*

**Sažetak**

Tema se bavi sigurnosnom revizijom, procjenom ranjivosti, penetracijskim testiranjem i analizom metoda i alata pomoću kojih se ta testiranja provode. U sklopu menadžmenta sigurnosnog rizika objasniti će se i prikazati potreba i važnosti procjene sigurnosnog rizika i sigurnosnog testiranja te će se objasniti ostale faze menadžmenta sigurnosnog rizika. U temi će se prikazati korištenje metoda i alata koji se najčešće primjenjuju pri sigurnosnim provjerama te koje su njihove prednosti i mane. Tema također prikazuje na koji način bi se opisane metode mogle unaprijediti kako bi davale efikasnije rezultate. Također su prikazani najveći i najznačajniji sigurnosni napadi koji daju stvarnu sliku opasnosti ne provođenja procjene sigurnosnog rizika i ispitivanja sigurnosti. Na primjeru testiranja realnog informacijskog sustava prikazano je sigurnosno testiranje te korištene metode koje su zajedničkim djelovanjem došle do rezultata te prikazale pronađene sigurnosne ranjivosti. Sama demonstracija izvođenja sigurnosnog testiranja je prikazala važnost procjene sigurnosnog rizika koja usmjerava i definira fazu sigurnosnog rizika.

**Ključne riječi:** ispitivanje sigurnosti, sigurnosna procjena, penetracijsko testiranje, informacijska sigurnost, mrežno skeniranje, skeniranje na ranjivosti.

# Sadržaj

Sadržaj ..... iii
1. Uvod ..... 1
2. Sigurnosni napadi u prošlosti ..... 2
3. Menadžment sigurnosnog rizika ..... 5
3.1. Procjena sigurnosnog rizika ..... 5
3.2. Ispitivanje sigurnosti ..... 6
3.3. Popravljanje rizičnih područja ..... 6
3.4. Operacijska sigurnost ..... 7
4. Ispitivanje sigurnosti ..... 8
4.1. Što je ispitivanje sigurnosti ..... 8
4.2. Kada je potrebno provesti ispitivanje sigurnosti ..... 10
4.3. Tipovi sigurnosnog ispitivanja ..... 11
4.3.1. Skeniranje mreže ..... 11
4.3.2. Skeniranje ranjivosti ..... 15
4.3.3. Probijanje lozinke ..... 18
4.3.4. Pregled dnevnika ..... 21
4.3.5. Provjera integriteta ..... 22
4.3.6. Detekcija virusa ..... 23
4.3.7. War Dialing ..... 26
4.3.8. War Driving ..... 27
4.3.9. Penetracijsko testiranje ..... 28
5. Sigurnosno testiranje na realnom informacijskom sustavu ..... 30
6. Zaključak ..... 38
Popis literature ..... 39
Popis slika ..... 41

# 1. Uvod

Sigurnosno testiranje je jedan od najvažnijih procesa za kontinuirano poboljšanje internetske sigurnosti te prikaz i analizu cyber prijetnji koje mogu rezultirati ozbiljnim napadima te gubitkom internih podataka, iskorištavanjem podataka korisnika sustava te gubitkom kontrole rada aplikacija. Često se provodi kada se već počinju primjećivati sigurnosni napadi te je potrebna brza procjena rizika, no vrlo često tada je već kasno.

Mishra (2021, str.2) navodi da financijski i legalni gubitci unutar organizacije koja nije na vrijeme zaštitila podatke mogu biti masovni. Te da implementacija procjene sigurnosnog rizika i sigurnosnog testiranja osigurava da problemi budu minimalni. No, kako raste svjesnost i znanje o sigurnosnim napadima tako se moraju razvijati i obrambeni mehanizmi za sigurnosne napade. A sigurnosno testiranje se mora provoditi kontinuirano.

Ono može spriječiti legalne probleme, tužbe, financijske probleme te gubitak klijentele i korisnika, no unatoč tome, tvrtke često ne provode sigurnosna testiranja, te su slabo zaštićena od internet napada. Za najbolju prevenciju napada najbolja mjera je pravovremena procjena sigurnosnog rizika te redovito provođenje sigurnosnih testiranja.

Landoll (2006, str. 34) navodi da sigurnosno testiranje često može imati nekoliko sinonima te se miješati s pojmovima kao što su sigurnosna revizija i sigurnosna procjena rizika. No zapravo ti pojmovi imaju bitne razlike te je bitno jasno razlikovati ih kao različite procese. Svi ti procesi su dio menadžmenta sigurnosnog rizika kojeg je definiran kao „upravljanje procesima koji osiguravaju da je razina rizika unutar organizacije unutar prihvatljivih granica koje je definirao viši menadžment. On se provodi u četiri faze: procjena sigurnosnog rizika, sigurnosno testiranje i pregled rezultata, popravljanje rizičnih područja, operacijska sigurnost“.

Sigurnosno testiranje je druga faza te se definira kao: „ Točna mjeru sigurnosti na operacijskom nivou te skup metoda koje mjerljivo, konzistentno i ponovljivo određuju sigurnosnu mjeru“ (Barcelo, Herzog, 2010, str. 196)

# 2. Sigurnosni napadi u prošlosti

Od kako postoji internet i internet aplikacije bilo je mnogo sigurnosnih napada, no neki od njih su bili istaknutiji s obzirom na masovne gubitke korisničkih podataka te tužbama koje su rezultirale ogromnim novčanim iznosima koje su tvrtke morale platiti.

Prema Vijayan (2010) 2008. godine je napadnuta tvrtka Heartland Payment Systems te je ugroženo oko 134 milijuna kreditnih podataka korisnika. Napad je otkriven 2009. godine od strane Vise i MasterCarda nakon nekoliko sumnjivih transakcija koji su pokrenuli istragu gdje su uhvaćeni A. Gonzalez te njegova dva neimenovana pomagača koja su uspjeli pronaći rupu u sustavu te izvesti SQL injection napad. Napadači su kažnjeni s 20 godina zatvora, a tvrtka je proglašena nesposobnom da podrži sigurnosne standarde za podatke nakon čega su morali platiti oko 140 milijuna dolara odštete kako bi nadoknadili sve nevažeće transakcije, a narednih godinu dana tvrtka nije imala odobrenje za rad sa financijskim transakcijama.

Prema TrendMicro (2016) 2012. godine je napadnut LinkedIn – društvena mreža za poslovne ljude. U toku napada ugroženi su podaci od 165 milijuna korisničkih računa, iako prve procjene štete nisu bile ni približno precizne te je napadač kasnije uhvaćen kako prodaje korisničke podatke ta 2000 dolara. LinkedIn je resetirao lozinke te nije pretrpio ozbiljnije posljedice kao što su plaćanje odštete ili udar na poslovanje.

Prema Weise MySpace je napadnut 2013. godine kada su napadači uspjeli doći do podataka od oko 360 milijuna korisnika. Napadači su uspjeli dešifrirati lozinke kriptirane pomoću SHA-1 algoritma. Napad je otkriven tek 3 godine kasnije kada su ukradeni podaci stavljeni na prodaju na dark web-u.

Prema BBC (2013) 2013. godine dogodio napad na tvrtku Adobe te se pretpostavlja da su bile ugrožene informacije od oko 150 milijuna korisničkih podataka. Iako je Adobe najprije objavio da su hakeri ukrali oko 3 milijuna korisničkih zapisa te nešto zapisa o kreditnim karticama, kasnija istraživanja o posljedicama napada dala su drugačije estimacije puno većih razmjera. Napadači su uspjeli izvući informacije o korisnicima kao što su imena, prezimena, lozinke, te čak informacije o kreditnim karticama korisnika. Adobe se kasnije nagodio, te je određeno na mora platiti milijun dolara za troškove suđenja te za odštetu korisnicima zbog krađe korisničkih podataka.

Prema O'Donnel (2018) Marriott International je napadnut 2014. te je ugroženo oko 500 milijuna korisnika nakon što su napadnuti sistemi koje je Marriott International koristio. Napad je otkriven 4 godine kasnije, te je otkriveno da su ukradeni osobni podaci korisnika, no nije otkriveno dali su napadači uspjeli dešifrirati kreditne podatke više od 100 milijuna korisnika.

Napadači nikada nisu otkriveni, no prema nekim izvorima za napad je zadužena kineska obavještajna služba koja je prikupljala podatke o američkim građanima.

Slika 1. Hotel Mariott (Izvor: https://media.threatpost.com/wp-content/uploads/sites/103/2018/11/30083518/bmimc-exterior-0001-hor-feat-e1543584929332.jpg )

Prema Reuters CNBC (2014) Ebay je također imao veći napad 2014. godine gdje je ugroženo oko 145 miliona korisničkih računa s informacijama o imenu i prezimenu, adresi i korisničkim lozinkama. Hakeri su pomoću računa zaposlenika dobili pristup mreži i neprimijećeno krali informacije 229 dana. Iako je tvrtka obavijestila korisnike kako bi pravovremeno promijenili svoje korisničke informacije, tvrtka je bila metom mnogih kritika kako nisu dobro reagirali i dovoljno obavještavali korisnike, iako financijske informacije nikada nisu bile ugrožene s obzirom na drugačiji proces spremanja i upravljanja.

Slika 2. Ebay logo (Izvor: https://www.trefis.com/stock/xlk/articles/240829/ebay-suffers-hack-attack/2014-05-28)

Prema BBC (2015) NetEase je napadnut 2015. godine te je ugroženo oko 235 milijuna korisničkih računa. Tvrtka se bavi pružanjem email servisa, te ukradeni podaci sadrže email adrese i lozinke koji su kasnije prodavani na dark web mjestima poput DoubleFlaga. Napad je

bio dio većeg napada gdje su ukradeni podaci od nekoliko pružatelja email servisa, no unatoč tomu tvrtka NetEase je odbila sve optužbe u vezi napada i nije sanirala štete. U ovom poglavlju treba opisati koje će metode i tehnike biti korištene pri razradi teme, kako su provedene istraživačke aktivnosti, koji su programski alati ili aplikacije korišteni.

Prema Goud (2018) U veljači 2018. godine napadnut je MyFitnessPal, zajedno s još 15 stranica kao dio velikog napada i otimanja informacija te kasnije i prodavanja tih istih informacija. Procjenjuje se da je 617 milijuna korisničkih podataka napadnuto te kasnije i stavljeno na prodaju. MyFitnessPal je javno objavio informacije o napadu a korisnici su morali promijeniti svoje lozinke, no kako je došlo napada ili tko je odgovoran, nije poznato.

Prema Dutta (2020) Tvrtka „Canva“ je 2019 godine imala sigurnosni napad gdje je bilo ugroženo oko 137 milijuna korisničkih računa. Napadači su uspjeli ukrasti podatke imena, prezimena, email adresa, korisničkih imena, i lozinka. Lozinke su bile hashirane pomoću „bcrypt“ algoritma. Napadači su navodno uspjeli samo pogledati no ne i ukrasti informacije o kreditnim karticama korisnika, te su uspjeli doći u posjed OAuth tokena usera koji se logirali pomoću svojih google računa. Tvrtka je uspjela na vrijeme detektirati napad te je ugasila server te time spriječila daljnje štete i curenje informacija. Tvrtka je o napadu obavijestila korisnike kako bi pravovremeno promijenili lozinke, a svi računi koji nisu na vrijeme promijenili korisničke podatke su proglašeni kao nesigurni.

# 3. Menadžment sigurnosnog rizika

Landoll (2006, str. 34) navodi da menadžment sigurnosnog rizika definira proces pomoću kojeg se može osigurati da je mjera sigurnosnog rizika unutar organizacije unutar prihvatljivih okvira koje je viši menadžment odobrio. Proces se odvija u četiri faze, te započinje s procjenom sigurnosnog rizika, zatim slijedi ispitivanje sigurnosti, potom slijedi popravljanje rizičnih područja, a potom i faza operacijske sigurnosti

## 3.1. Procjena sigurnosnog rizika

Landoll (2006, str. 34) navodi da je procjena sigurnosnog rizika objektivna analiza efektivnosti trenutnih sigurnosnih kontrola koje su uspostavljene nad organizacijskim resursima te vjerojatnost gubitka i štete nad tim resursima. Procjena sigurnosnog rizika procjenjuje stupanj prijetnji unutar organizacije, vrijednost resursa organizacije, kritičnost sistema, ranjivost sigurnosnih kontrola, utjecaj štete na organizaciju, te preporuke i uređivanje dodatnih kontrola koje bi pomogle da se sigurnosni rizik smanji na prihvatljivu razinu.

Viši menadžment sudjeluje u ovom procesu te na temelju rezultata procjene sigurnosnog rizika procjenjuje dali su dodatne sigurnosne kontrole potrebne ili su trenutne sigurnosne kontrole dovoljne. Jhala N. (2014. str. 12) navodi da postoji nekoliko segmenata čiji sigurnosni rizici se moraju procijeniti kako bi analiza i procjena sigurnosnog rizika informacijskog sustava bila kompletna. To su:

- Kontrola pristupa: Procedura koja određuje dopuštenja korisnika za pristup dijelocima sustava.
- Zaštite aplikacija i podataka: Na koji način su implementirane zaštite te u kojoj mjeri su ranjive.
- Zaštita platforme: Treba se procijeniti u kojoj mjeri je hardware zaštićen od krađe, vremenskih nepogoda, neovlaštenog pristupa itd.
- Zaštita mreže: Procjenjuje se u kojoj mjeri je mreža zaštićena od napada te koje sposobnosti obrane ima.

Sigurnosne kontrole mogu biti određene unutar organizacije, unutar države ili unutar većih geografskih cjelina. Najvažniji kriteriji koji propisuju upravljanje informacijama su „Opća uredba o zaštiti podataka“ koje se moraju pridržavati sve članice Europske Unije, te „Health Insurance Portability and Accountability Act“, „Sarbanes-Oxley Act“ i „the California Security Breach Information Act“ koji propisuju kriterije za upravljanje informacijama unutar SAD-a.

Prema OWASP (2014. str 18) Cilj procjene sigurnosnog rizika je detaljno proučavanje i definiranje testiranja sustava, kako bi se dobio sistematski pregled ranjivosti i slabosti sustava, te ocjena otpornosti i izdržljivosti sustava na određene vrste napada što rezultira sigurnosnim politikama i strategijama koje omogućavaju tvrtki da kreiraju adekvatne edukacije i obuke, prate sigurnosne standarde i preporuke, te zaštite korisnike sustava.

Procjenu sigurnosnog rizika treba raditi kako bi se definirale interne sigurnosne police te pravila i upute vezane uz ispitivanje sigurnosti. Procjena sigurnosnog rizika se također treba raditi kako bi se uspostavili osnovni sigurnosni standardi unutar organizacije te na taj način identificirali sigurnosni problemi i mane unutar sustava.

Prema OWASP (2014. str 16) U sklopu procjene sigurnosnog rizika se prvo trebaju dogovoriti smjernice i ciljevi za sigurnosno ispitivanje kako bi bilo jasno koji tip sigurnosnog ispitivanja će se provoditi i koje sve provjere će se uključiti. Vrlo važno je imati jasnu sliku koji dijelovi sustava i tvrtke će se revidirati – dali to uključuje opremu za rad, mrežni promet, softver na računalima itd.

## 3.2. Ispitivanje sigurnosti

Sljedeći korak je provođenje sigurnosnog ispitivanja te identifikacija potencijalnih ranjivosti i prijetnji.

Landoll (2006, str. 34) navodi da je sigurnosno testiranje pregled sigurnosnih kontrola i njihovih djelovanja prema sigurnosnim zahtjevima.

Sigurnosne kontrole se utvrđuju tokom faze procjene sigurnosnog rizika i testiraju se tokom druge faze, tj. sigurnosnog testiranja koje se ponavlja češće. Dalje će se objasniti i prikazati sigurnosno testiranje te će se analizirati i prikazati uporabe različitih alata i metoda za provedbu sigurnosnih testiranja.

## 3.3. Popravljanje rizičnih područja

Prema Scarfone, Souppaya, Cody, Orebaugh (2008). u trećoj fazi se implementiraju rješenja za pronalaske i analize iz prijašnjih faza a prijetnje pronađene pri ispitivanju sigurnosti se koriste kako bi se uvele nove police i kontrole koji bi riješile problem rizičnih područja.

Viši menadžment također sudjeluje u ovoj fazi, gdje na temelju rezultata ispitivanja sigurnosti može donijeti odluke koje će pridonijeti smanjenju rizika te se uvode nove sigurnosne police i kontrole te se postojeće uređuju kako bi se pronađeni problemi efektivno riješili.

Prema Stewart, Chapple, Gibson (2015, str.) popravljanje rizičnih područja se bavi implementacijom sigurnosnih zaštita i mjera koje eliminiraju ranjivosti ili blokiraju prijetnje. Prioritete određuje viši menadžment, no najčešće su to one kontrole koje su najučinkovitije. Popravljanje rizičnih područja ne uključuje uvijek popravke, već ponekada to znači zamjena tehnologije prihvatljivijom tehnologijom s manjim brojem sigurnosnih rizika.

Popravljanje rizičnih područja se ponavlja nakon svakog sigurnosnog ispitivanja.

## 3.4. Operacijska sigurnost

U sljedećoj fazi se provode operacije za implementaciju novih sigurnosnih kontrola i polica koje se uvedene i uređene u fazi popravljanja rizičnih područja.

Prema Gregg (str. 495), to uključuje sve od praksa i politika za edukaciju i zapošljavanje zaposlenika pa sve do izdavanja patch verzija aplikacija s popravljenim sigurnosnim propustima do provođenja sigurnosnih edukacija i treninga koji će osigurati da se nove sigurnosne kontrole i police adekvatno provode.

# 4. Ispitivanje sigurnosti

## 4.1. Što je ispitivanje sigurnosti

Prema Felderer, Johns, Buchler, Breu (2016, str.2) ispitivanje sigurnosti je sistematsko procjenjivanje i mjerenje koliko točno sustav djeluje na temelju propisanih uvjeta, demonstrira njihovu točnost i sposobnost obrane i otpornost na niz određenih sigurnosnih napada te kolike su mu performanse u odnosu na niz kriterija koji su definirani u procjeni sigurnosnog rizika.

Prema OWASP (2014. str 12) veliki problem je dok tvrtke smatraju da je nepotrebno raditi sigurnosno testiranje sustava, a tek nakon što nastane problem i štete budu velike kreće se s procjenom ispitivanja sigurnosti i sigurnosnim testiranjem. Sigurnosno testiranje zato treba raditi na vrijeme te ponavljati u određenim vremenskim intervalima kako bi se štete i gubitci spriječile pravovremeno.

Ispitivanje sigurnosti treba raditi kako bi se identificirali podatci koji su redundantno vraćeni unutar odgovora http zahtjeva, te kako bi se provjerile zaštite kritičnih podataka, identificirali problemi pri obradi i spremanju podataka. Ispitivanje sigurnosti također treba raditi kako bi se provjerilo dali sigurnosni sustavi prate upute sigurnosnih polica i uputa definiranih unutar faze procjene sigurnosnog rizika

Sigurnosna ispitivanja uključuju provjeru različitih točaka unutar cjelokupnog sustava, bilo da se radi o ispitivanju provedbe različitih politika i policama ili o provjeri arhitekture sustava svaka točka može otkriti ranjivosti, te je potrebno detaljno isplanirati plan testiranja te vrsta testiranja unutar procjene sigurnosnog rizika kako se potencijalna opasnost ne bi propustila provjeriti.

Sigurnosne police i pravila su neizostavni dio faze procjene rizika te je u sklopu sigurnosnog ispitivanja potrebno provjeriti dali sustav djeluje na propisani način. To može uključivati provjeru sigurnosnih preporuka iz GDPR-a - dali je korisniku omogućeno brisanje vlastitih informacija? Dali se informacije skladište na propisan način? Dali se informacije skladište na propisani vremenski period itd. Provjera upravljanja, skladištenja i obrade informacija treba biti izrazito temeljita s obzirom na veliku važnost koju predstavlja kada pričamo o sigurnosti sustava. Također se provjeravati dali je digitalna oprema u posjedu firme ili se radi o osobnim uređajima? Dali zaposlenici koriste ne dopuštene aplikacije koje bi mogle naštetiti sustavu itd.

Sigurnosno ispitivanje također uključuje provjeru enkripcije korištene za podatke. Provjerava se sigurnost kritičnih informacija koje nipošto ne smiju biti otkrivene poput lozinka

ili kreditnih informacija. Naravno opće poznato je da je neke algoritme lakše dekriptirati od drugih, stoga provjera enkripcije može uključiti i provjeru „kvalitete“ algoritma, npr. dali se koristi SHA-1 algoritam ?

Scarfone, Souppaya, Cody, Orebaugh (2008, str.15). navode da se pri sigurnosnom testiranju najviše izbjegavaju koristiti metode koje imaju utjecaja na mrežu, djelomice zbog toga sto svaki utjecaj na mrežu može imati negativan utjecaj na korisnike sustava a djelomice zbog toga što se takvi napadi moraju strogo regulirati i analizirati u fazi procjene sigurnosnog rizika, te iziskuju puno vremena za provedbu. Napadači nažalost imaju neograničeno vrijeme kako bi provodili mrežne napade te je zbog toga često obrana od mrežnih napada najslabije testirana.

Mrežne ranjivosti su kritična točka koju treba pregledati i provjeriti u sklopu sigurnosnog ispitivanja. Potrebno je provjeriti može li napadač na bilo koji način iskoristiti pristup mreži kako bi pridobio kritične informacije ili dobio pristup ostalim dijelovima sustava. Potrebno je provjeriti mrežni promet, razmjene poruka i datoteka te pristupne točke, s obzirom da napadač može iskoristiti „phishing“ metodu te pokušati naći podatke koji bi mogao iskoristiti.

https://www.tessian.com/blog/phishing-statistics-2020/ navode da je u današnje vrijeme phishing najčešći tip napada, a o tome svjedoči i činjenica da je preko 75% sustava napadnuto phishing metodom. Ta metoda je vrsta socijalnog inženjeringa gdje napadač preusmjeri korisnike mreže na kopiju sustava ili na nesiguran način preko preusmjerivanja poruka otkrije kritične informacije.

Prema Felderer, Johns, Buchler, Breu (2016, str.2) provjera softverskih sustava je također izrazito kritični dio sigurnosnog ispitivanja gdje je u prvom planu provjera autentifikacije i autorizacije kao prve linije obrane protiv napadača, također je važno provjeriti da sigurnosno testiranje provjerava povjerljivost podataka, integritet podataka te dostupnost podataka i njihova točnost. Provjerava se integritet prikazanih podataka, funkcioniranje sustava, te komunikacija između više vrsti sustava koji čine softverski ekosustav tvrtke.

Arhitektura sustava je sljedeća kritična točka sigurnosnog ispitivanja. Nepravilna i loše postavljena arhitektura može biti uzrok masivnih šteta i napada, ili možete biti uzrok minimalne štete ukoliko je arhitektura sustava dobro promišljena i odgovara potrebama sustava.

## 4.2. Kada je potrebno provesti ispitivanje sigurnosti

Ovisno o vrsti i namjeni sustava ovisi i učestalost i potreba za sigurnosnim ispitivanjima. Sustavi koji se bave skladištenjem i obradom kritičnih podataka kao što su sustavi koji spremaju kartične informacije, ili sustavi koje skladište podatke vezane uz zdravstveni sustav trebaju češća i oštrije regulirana sigurnosna ispitivanja, a broj sustava i aplikacija koje međusobno komuniciraju dodatno kompliciraju situaciju te stvaraju potrebu za detaljnijim i pomnije planiranim sigurnosnim ispitivanjima.

Zakoni i politike unutar države također mogu nametati potrebu za provođenjem sigurnosnih ispitivanja. Primjerice uvođenje GDPR-a ili „General Dana Protection Regulation“ unutar država europske unije propisuje načine i pravila spremanja i skladištenja podataka.

Prema Wack, Tracy, Souppaya (2003, str.41) preporuča se da se sigurnosna ispitivanja prosječnih sustava rade svakih 6 do 12 mjeseci, no ovisno o vrsti sustava prema Stewart, Chapple (2015, str. 568). sigurnosna testiranja se mogu provoditi i svaki mjesec. Ukoliko se radi o složenijim sustavima kojima je bitan integritet podataka, dobra preporuka je da se sigurnosna ispitivanja rade svaki mjesec s obzirom na to da se na taj način na vrijeme mogu uočiti ranjivosti i kritične točke.

Prema OWASP (2014. str 12) češće provođenje sigurnosnih ispitivanja može pronaći i teže uočljive ranjivosti koje se rezultat nekog uzorka ponašanja. Najbolje mjerilo za određivanje frekvencije sigurnosnih ispitivanja je kompleksnost samog sustava i važnost podataka koji su korišteni u sustavu. Ukoliko je kompleksnost sustava manja, sigurnosna ispitivanja će se provoditi rjeđe.

Pravilo je međutim, da se sigurnosna ispitivanja obavezno provode nakon izdavanja nove verzije sustava, pri tome se misli na „major“ verzije, a nakon izdavanja „minor“ verzija sustava nije imperativ da se sigurnosna ispitivanja provode.

Također ukoliko se provode migracije podataka, ili se bitno mijenjaju korištene tehnologije (primjerice promjene vrste baze podataka, prelazak na Angular 10 s Angulara 8, itd. ) također se preporučava izvođenje sigurnosnih ispitivanja s obzirom da sustav tada može biti podložan novim vrstama ranjivosti koje ranije nisu bile problematične.

Sigurnosna ispitivanja se također preporučuju provoditi nakon uvođenja novih regulatornih sredstava koju diktiraju spremanje i upravljanje podataka. Naravno, ukoliko dođe do napada na sustav, sigurnosna ispitivanja, kao i procjena sigurnosnog rizika će se trebati provesti, i to detaljnije, te eventualno može biti fokusirana na određeni dio sustava kako bi se odredilo koji je točno uzrok i posljedica napada.

## 4.3. Tipovi sigurnosnog ispitivanja

Prema Wack, Tracy, Souppaya (2003) postoji nekoliko tipova sigurnosnog testiranja te se često nadopunjuju i koriste zajedno ili pojedinačno ovisno o prigodi. Navode kako postoje: skeniranje mreže, skeniranje ranjivosti, probijanje lozinke, pregled dnevničkog zapisa, provjera integriteta, detekcija virusa, War Dialing, War Driving i Penetracijsko testiranje

### 4.3.1. Skeniranje mreže

Jhala N (2014, str. 1) navodi da je skeniranje mreže kritična metoda testiranja što se tiče sigurnosnog testiranje te da je osnova za prikupljanje informacija o sustavima i mreži.

Skeniranje mreže je proces gdje se analizira internet mreža koja se koristi unutar organizacije kako bi se pronašle ranjivosti koje ugrožavaju sigurnost, privatnost i integritet podataka i aplikacija na mreži. Skeniranje mreže se obavlja pomoću raznih alata koji skeniraju i detektiraju mrežni promet, servise na mreži, portove koji su korišteni i tko ih koristi, te uređaje koji su spojeni na mrežu (PC-ovi, laptopovi, smart tv, smart printer, ruteri).

„Port skeneri“ su alati koji se koriste za skeniranje mreže, a najpoznatiji su SolarWinds, ManageEngine, NMap, Advanced Port Scanner, Angry IP Scanner, NetCat, Unicornscan, MiTeC Scanner itd.

„Port skeneri rade na principu da prvo identificiraju poslužitelje unutar raspona adresa koje specificira korisnik pomoću TCP/IP (Transport Control Protocol/Internet Protocol) ICMP (Internet Control Message Protocol) i ICMP ECHO_REPLY paketa. Kada se aktivni poslužitelji skeniraju i identificiraju, tada se skeniraju njihovu otvoreni TCP i UDP portovi pomoću kojih će se moći identificirati mrežni servisi na tom poslužitelju. Različiti mrežni skeneri imaju različite mogućnosti i svojstva“ (Wack, Tracy, Souppaya, 2003, str. 22)

Prema OWASP (2014. str 49) iz informacija o otvorenim portovima i otvorenim internet servisima se može izvući mnogo različitih informacija, te ovisno o vrsti mrežnog skenera može se napraviti fingerprinting, tj. dobiti informacije o operacijskom sustavu. Informacije o operacijskom sustavu treba interpretirati stručnjak jer sami skeneri i alati nemaju informacije koji je točno operacijski sustav u pitanju već samo daju informacije o vrsti, broju porta, broju paketa i vremenu odgovora na ICMP pakete na temelju kojih se može zaključiti vrsta operacijskog sustava. Osim toga moguće je zaključiti i koje su vrste aplikacija pokrenute na kojem portu.

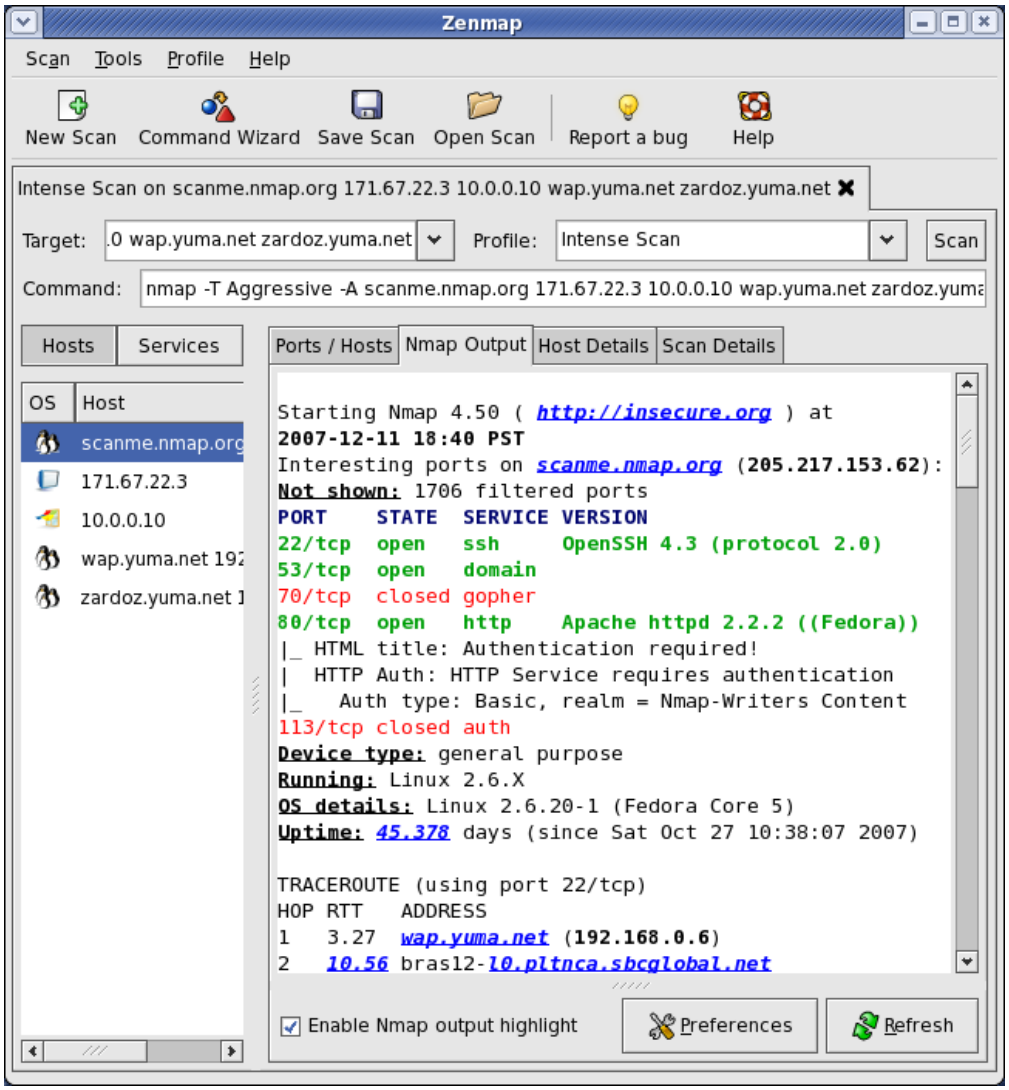
Same informacije neće imati previše značaja bez kvalitetne interpretacije stručnjaka koji će interpretirati vrstu aplikacije te usmjeriti analizu i procjenu sigurnosnih rizika na ranjivosti specifične za te aplikacije.

Na temelju tih informacija se dalje može usmjeriti plan procjene sigurnosnog rizika te adaptirati plan za metode testiranja koje će se koristiti. Nakon provođenja mrežnog skeniranja potrebno je analizirati podatke i dokaze koji su prikupljeni tokom testiranja, identificirati problematične poslužitelje kojima nije dopušteno da se spoje na mrežu, identificirati korištenje servisa koji nisu sigurni i imaju veliki broj sigurnosnih ranjivosti ili nedopuštenih radnji koje se prema sigurnosnim pravilima ili policama ne bi trebale odvijati.

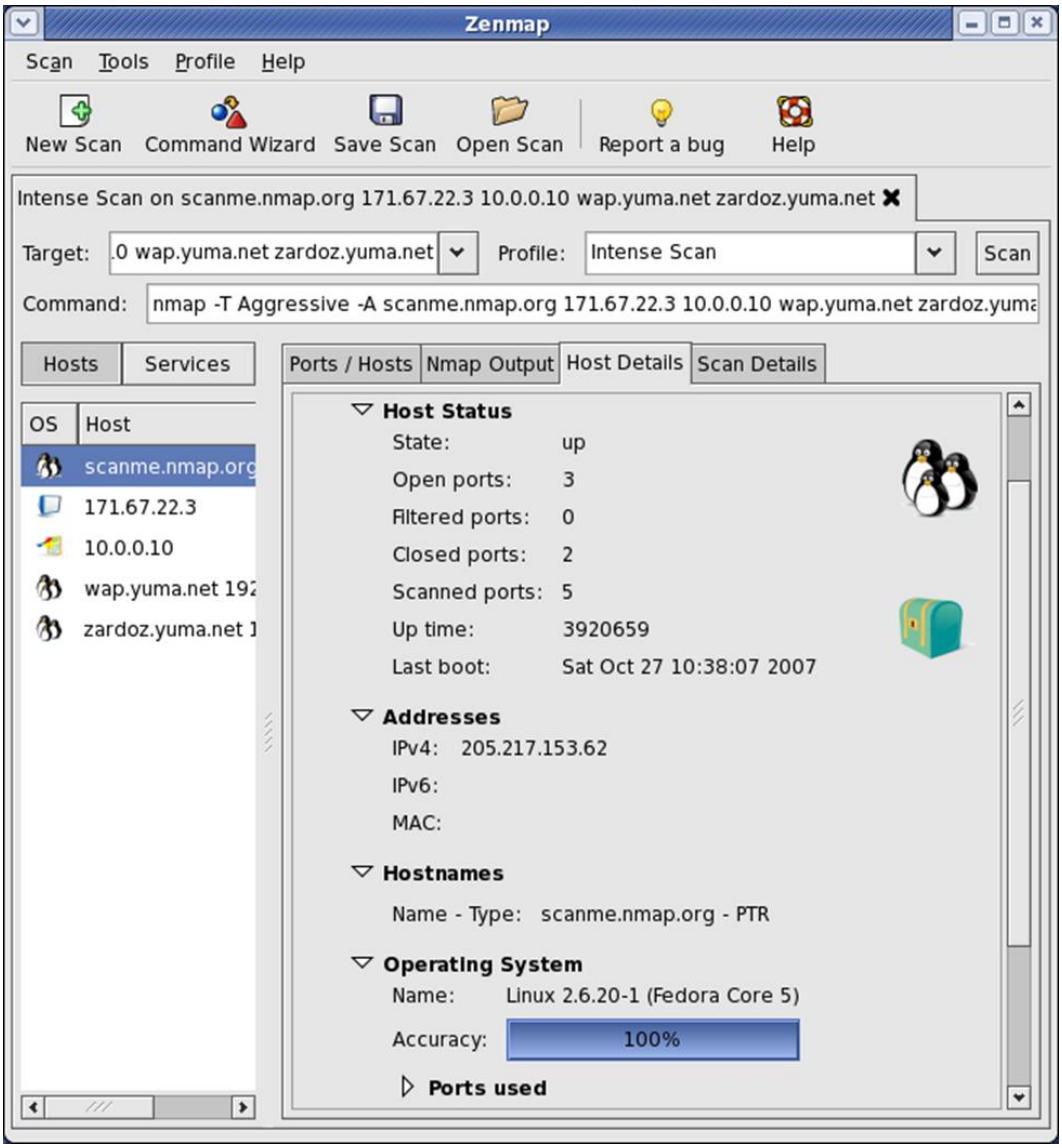
Također je potrebno napraviti analizu sigurnosne konfiguracije te provjeriti dali je konfiguracija u skladu s organizacijskim policama sigurnosti. Nakon što su se prikupile informacije o mreži i mrežnim aktivnostima može se provoditi penetracijsko testiranje, a prikupljene informacije se uveliko pomoći detaljnije definirati plan i metode koje će se koristiti u penetracijskom testiranju.

NMap je najpopularniji alat koji se koristi za „skeniranje mreže“, te nadgledanje i analiziranje mrežnog prometa. Prema (https://nmap.org/) Nmap koristi sirove ip pakete kako bi se utvrdilo koji poslužitelji su dostupni na mreži, koji servisi (aplikacija i verzija) su dostupni preko tih poslužitelja te na kojem su operacijskom sustavu te koji je tip firewalla koji se koristi. Alat je dizajniran kako bi brzo skenirao velike mreže, i može se pokrenuti na svim operacijskim sustavima. Nmap također ima i GUI verziju Zenmap.

Liao et al., (2020, str. 2) navode da nmap ima dualnu svrhu te da ga koriste oboje mrežni administratori i haker. Mrežni administratori ga koriste kako bi ocijenili performanse IDS sustava i pronašli sigurnosne ranjivosti na serveru dok ga hakeri koriste za prikupljanje važnih informacija o sustavu. Testeri i hakeri ga također koriste kako bi prikupili informacije o operacijskom sustavu tako što pronađu odgovarajuću IP/TCP odgovore unutar „fingerprint“ baze podataka.




Slika 3: Prikaz nmap ispisa nakon skeniranja (Izvor: https://nmap.org/zenmap/images/zenmap-no-648x700.png )



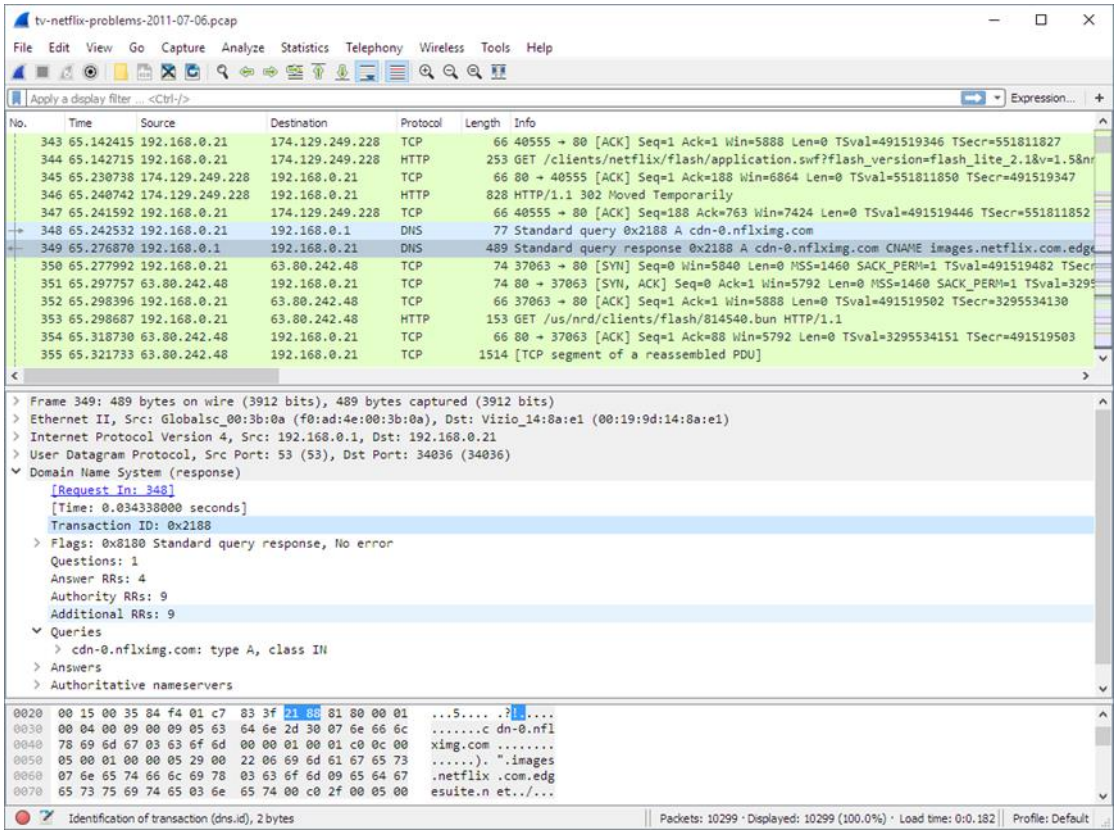

Slika 4: Prikaz detalja o poslužitelju (Izvor: https://nmap.org/zenmap/images/zenmap-hd-648x700.png )

Osim toga, također se koriste i tcpdump, Wireshark, Fiddler i Capsa koji imaju mogućnosti praćenja i analize mrežnih paketa.

Wireshark je jedan od popularnijih alata za analiziranje mreže.

Prema (https://www.wireshark.org/) Wireshark se koristi za analiziranje događanja na mreži na mikroskopskoj razini te možemo reći da je standard za analiziranje mreže, a koriste ga eksperti u raznim tvrtkama i organizacijama. Wireshark se koristi za duboku inspekciju stotine mrežnih protokola, „real time“ hvatanje mrežnog prometa i offline analizu, VoIP analizu, dekripciju za veliki broj protokola uključujući i IPsec, ISAKMP, Kerberos, SNMPv3, SSL/TLS, WEP i WPA/WPA2, te podršku za spremanje i obradu uhvaćenog prometa u veliki broj formata kao što su tcpdump (libpcap), Pcap NG, Catapult DCT2000, Cisco Secure IDS iplog, Microsoft

Network Monitor, Network General Sniffer® (compressed and uncompressed), Sniffer® Pro, and NetXray®, Network Instruments Observer, NetScreen snoop, Novell LANalyzer, RADCOM WAN/LAN Analyzer, Shomiti/Finisar Surveyor, Tektronix K12xx, Visual Networks Visual UpTime, WildPackets EtherPeek/TokenPeek/AiroPeek itd.




Slika 5: Prikaz Wireshark sučelja (Izvor: https://www.wireshark.org/docs/wsug_html/wsug_graphics/ws-main.png)

## 4.3.2. Skeniranje ranjivosti

Skeniranje ranjivosti je pouzdan i brz način za pronalaženje ranjivosti i mana u sustavu a provode ga sistemski i mrežni administratori te nije nužna interpretacija rezultata u tolikoj mjeri kao kod mrežnog skeniranja kako bi se dobile informacije o pronađenim ranjivostima već alati koji se koriste samostalno pružaju inicijalne informacije.

No iako se neke poznatije ranjivosti pronađu relativno lako, za pronalazak svih ranjivosti ova metoda nije dovoljna te je poželjno kombinirati skeniranje ranjivosti s ostalim metodama kako bi se dobila realna ocjena sigurnosnog rizika. Prema Landoll (2006, str. 20) Skeniranje ranjivosti nije dovoljno pouzdano za ocjenjivanje sigurnosnog rizika, djelomice zato što često ima rezultate koji lažno ocjenjuju kao ranjivosti iako to nisu.

Skeniranje ranjivosti ili “vulnerability scanning” djeluje na sličnom principu kao mrežno skeniranje, te također pruža detaljne informacije o ranjivostima pronađenim pri skeniranju te pružaj informacije kako popraviti rizične dijelove.

Prikladno je za automatizirano provođenje i rješavanje površinskih ranjivosti no nedostatak pouzdanosti i česta potreba za novom verzijom skenera su jedne od glavnih mana ove metode.

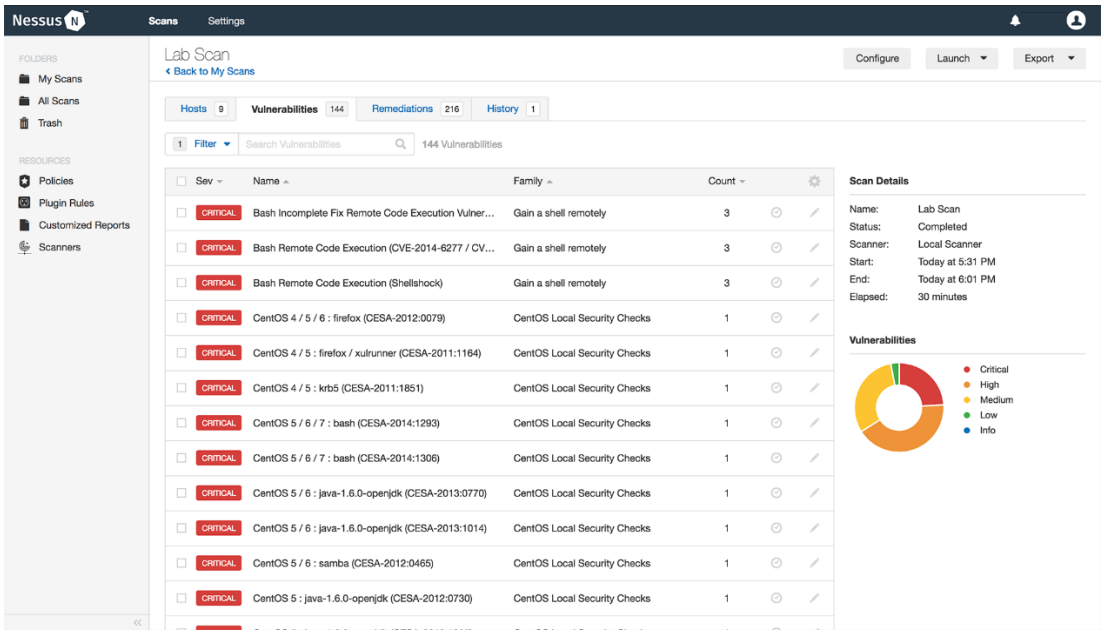
Prema Wack, Tracy, Souppaya (2003, str. 24) još jedna veća mana je što alati za skeniranje ranjivosti proizvode veće količine mrežnog prometa koji može imati negativne posljedice na mrežu ili poslužitelje. Također je problematično što alati za skeniranje ranjivosti pružaju mogućnost DoS napada koji može imati negativnu utjecaj i razne posljedice ukoliko nije proveden od strane iskusnog profesionalca.

Prema OWASP (2014. str 38) Alati koji rade skeniranje ranjivosti pružaju širok spektar informacija kao što su identificiranje starije verzije softvera ili verzije softvera koje sadrže poznate ranjivosti (te automatski preporučaju verziju softvera koja bi se trebala koristiti, te verziju softvera koja ima popravljene ranjivosti iz prijašnjih verzija).

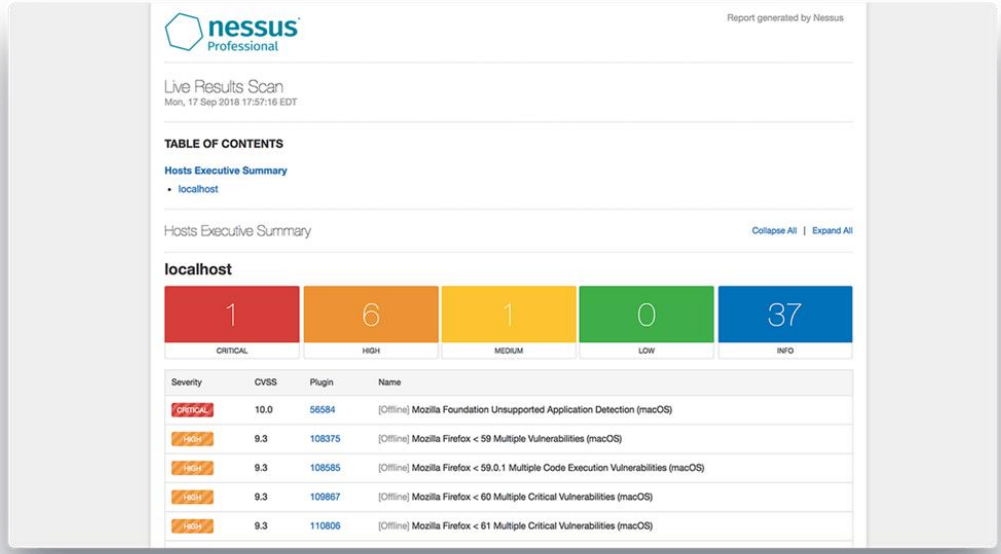
Alati uspoređuju sustav koji se skenira s bazom ranjivosti koju sadrže, te ukoliko identificiraju sustav mogu sve poznate ranjivosti tog sustava relativno lako riješiti. Naravno, nemaju svi alati sve navedene ranjivosti, stoga se često alati kombiniraju kako bi spektar ranjivosti koje se skeniranjem mogu pronaći i riješiti bude veći. Uobičajena praksa je da se uz komercijalni alat za skeniranje ranjivosti najčešće koriste dodatni alati za skeniranje ranjivosti koji su besplatni.

Skeniranje ranjivosti omogućuje sistemskim i mrežnim administratorima da uz pomoć gotovih alata identificiraju površinske ranjivosti prije nego ih hakeri uspiju iskoristiti. Skeniranje ranjivosti je brz i jednostavan način da se smanji broj površinskih ranjivosti i automatizirano pronađe rješenje za pronađene ranjivosti.

Wack, Tracy, Souppaya (2003, str. 25) također navode da nakon što se provede skeniranje ranjivosti treba popraviti sigurnosne ranjivosti u novoj ili patch verziji softvera. Ta verzija se tada treba „deployati“ na najsigurniji mogući način kako sustav nebi bio kompromitiran. Također je potrebno popraviti i urediti procedure kako bi sustav rutinski dobivao novije verzije i dodati rolu zaposlenika koju će pratiti upozorenja o najnovijim ranjivostima, te ih odgovarajuće pregledati i analizirati. Također je potrebno popraviti sigurnosne police, dokumentaciju i pravila kako bi se osiguralo da sustav rutinski dobiva najnovije verzije. Najpoznatiji alati koji se koriste su Nessus, SecureScanNX, Saint itd.




Slika 6: Nessus sučelje i rezultati skeniranja (Izvor: https://www.tenable.com/sites/all/themes/tenablefourteen/img/nessus/nessus-vulnerability-grouping-and-snoozing_large.png)




Slika 7: Nessus prikaz sadržaja (Izvor: https://media-s3-us-east-1.ceros.com/tenable/images/2018/12/04/9431d4e1871db487575704540178a97d/reporting-screen.png?imageOpt=1&fit=bounds&width=1249 )

### 4.3.3. Probijanje lozinke

Password cracking ili probijanje korisničkih lozinka se provodi kako bi se procijenilo vrijeme potrebno za otkrivanje korisničkih lozinka korisnika unutar sustava te dali korisnici koriste dovoljno „jake“ lozinke.

Prema OWASP (2014. str 78) Sigurnosne police i pravila propisuju dužinu i sastav znakova svake lozinke, te ukoliko se korištene lozinke drže propisanih pravila, ranjivost na brzo dešifriranje korisničkih lozinka je znatno manja.

Naravno, dužina trajanja dešifriranja lozinke ovisi i o vrsti algoritma koja se koristi. Lozinke šifrirane SHA-1 algoritmom će biti puno prije dešifrirane od lozinka šifriranih SHA-2 algoritmom, stoga je bitno da se pri testiranju i dešifriranju korisničkih lozinka uzme u obzir i vrsta algoritma te usporedi sa zahtjevima sigurnosnih polica kako bi se zahtjevi za valjanu lozinku mogli urediti a lozinke mogle promijeniti u dovoljno jake lozinke.

Kriptirane lozinke se pribavljaju različitim metodama: „sniffanjem“ mreže, sql injekcijom, pribavljanjem administratorskih prava ili čak social engineeringom, a nakon što su hashevi pribavljeni, alat za dekriptiranje korisničkih lozinka automatski generira hasheve sve dok ne pronađe odgovarajući par.

Prema OWASP (2014. str 77, 78, 79) Alat može koristiti jednu od nekoliko metoda: napad rječnikom, hibridni napad ili brute force napad. Napad rječnikom se temelji na tome da testira lozinke na poznate riječi iz raznih rječnika, dok se brute force napad temelji na tome da se testira ne povezana sekvenca znakova. Hibridni napad testira kombinaciju riječi iz rječnika s dodanim ne povezanim sekvencama znakova.

Iako je sve lozinke moguće probiti, ukoliko je lozinka složena od dovoljne sekvence znakova slova i brojeva može se dogoditi da treba više mjeseci da se takva lozinka probije pa se može ocijeniti kao dovoljno sigurna. Ova metoda provjerava upravo dali je vrijeme za probijanje lozinke prihvatljivo te dali je potrebno promijeniti pravila za korisničke lozinke ili osigurati edukacija korisnika.

Rudy J., Rodwald P. (2020, str. 2) navode zbog naprednih algoritama za hashiranje lozinki, jedan GPU nije dovoljan da u prihvatljivom vremenskom roku dođe do riješenje, stoga se trenutno populariziraju platforme za probijanje lozinki kojem imaju nekoliko GPU-ova te dolaze do rješenja u puno prihvatljivijem i realnijem roku.

Jedan takav primjer je alat Hashtopolis koji pomoću nekoliko GPU-ova probija lozinke u rekordnom roku. Alat je baziran na već postojećem alatu Hashcat. Rudy J. i Rowald P (2020, str.2) navode da ovaj alat ubrzava pronalaženje rješenja za minimalno 4% a u idealnom slučaju za čak 38%.

Ovakva vrsta testiranja se provodi vrlo često dok se ne uspostavi zadovoljavajući sustav autorizacije korisnika.

Jedan od najpoznatijih alata za probijanje lozina je „John the Ripper password checker“. Command line alat koji služi za sigurnosne testiranje lozinki i povratak izgubljenih lozinki, dostupan je na većini operacijskih sustava te dolazi i u Pro i Jumbo verziji koje imaju puno mogućnosti, no za testiranje lozinki dovoljna je besplatna verzija.

```
$ /usr/sbin/john
John the Ripper password cracker, version 1.8.0.6-jumbo-1-bleeding [linux-x86-64-xop]
Copyright (c) 1996-2015 by Solar Designer and others
Homepage: http://www.openwall.com/john/

Usage: john [OPTIONS] [PASSWORD-FILES]
--single[=SECTION]        "single crack" mode
--wordlist[=FILE] --stdin wordlist mode, read words from FILE or stdin
                  --pipe  like --stdin, but bulk reads, and allows rules
--loopback[=FILE]         like --wordlist, but fetch words from a .pot file
--dupe-suppression        suppress all dupes in wordlist (and force preload)
--prince[=FILE]           PRINCE mode, read words from FILE
--encoding=NAME           input encoding (eg. UTF-8, ISO-8859-1). See also
                          doc/ENCODING and --list=hidden-options.
--rules[=SECTION]         enable word mangling rules for wordlist modes
--incremental[=MODE]      "incremental" mode [using section MODE]
--mask=MASK               mask mode using MASK
--markov[=OPTIONS]        "Markov" mode (see doc/MARKOV)
--external=MODE           external mode or word filter
--stdout[=LENGTH]         just output candidate passwords [cut at LENGTH]
--restore[=NAME]          restore an interrupted session [called NAME]
--session=NAME            give a new session the NAME
--status[=NAME]           print status of a session [called NAME]
--make-charset=FILE       make a charset file. It will be overwritten
--show[=LEFT]             show cracked passwords [if =LEFT, then uncracked]
--test[=TIME]             run tests and benchmarks for TIME seconds each
--users=[-]LOGIN|UID[,..] [do not] load this (these) user(s) only
--groups=[-]GID[,..]      load users [not] of this (these) group(s) only
--shells=[-]SHELL[,..]    load users with[out] this (these) shell(s) only
--salts=[-]COUNT[:MAX]    load salts with[out] COUNT [to MAX] hashes
--save-memory=LEVEL       enable memory saving, at LEVEL 1..3
--node=MIN[-MAX]/TOTAL    this node's number range out of TOTAL count
--fork=N                  fork N processes
--pot=NAME                pot file to use
--list=WHAT               list capabilities, see --list=help or doc/OPTIONS
--format=NAME             force hash of type NAME. The supported formats can
                          be seen with --list=formats and --list=subformats
```

Slika 8: John the Ripper komandne naredbe (Izvor: https://miro.medium.com/max/777/1*JpTERXiDauT2VtuJv-BthA.png)

Hashsuit je alat za sigurnosno testiranje lozinki te se također vrlo često koristi. Podržava 13 najpoznatijih hash tipova: LM, NTLM, MD5, SHA-1, SHA-256, SHA-512, DCC, DCC2, SSHA, md5crypt, bcrypt, sha256crypt, sha512crypt.

Quick benchmark will finish in 10 min (spending 5 sec in each attack).

| Processor Name | Frequency | L1 | L2 | L3 | Ram | Other Information |
|---|---|---|---|---|---|---|
| Intel Core i5-4670 | 3.40GHz | 64KB | 256KB | 6MB | 4GB DDR3-1600 | Windows 8.1 Professional 64-bit |
| GeForce GTX 970 | 1.20GHz | 208KB | 1.8MB | 0KB | 4GB GDDR5-3505 | Driver 355.82 |

| Format | Threads | 1 | 10 | 100 | 1000 | 10000 | 65536 | 100000 | 1000000 |
|---|---|---|---|---|---|---|---|---|---|
| LM | 4 | 621M | 579M | | | | | | |
| | GPU | 7.51G | 7.25G | | | | | | |
| NTLM | 4 | 456M | 448M | | | | | | |
| | GPU | 22.6G | | | | | | | |
| Raw-MD5 | 4 | 301M | | | | | | | |
| | GPU | 10.9G | | | | | | | |
| Raw-SHA1 | 4 | 183M | | | | | | | |
| | GPU | 3.71G | | | | | | | |
| Raw-SHA256 | 4 | 93.0M | | | | | | | |
| | GPU | 1.32G | | | | | | | |
| Raw-SHA512 | 4 | 34.7M | | | | | | | |
| | GPU | 483M | | | | | | | |
| | | 1 | 4 | 16 | 64 | | | | |
| DCC | 4 | 293M | | | | | | | |
| | GPU | 5.27G | | | | | | | |
| SSHA | 4 | 153M | | | | | | | |
| | GPU | 3.37G | | | | | | | |
| MD5CRYPT | 4 | 379K | | | | | | | |
| | GPU | 5.63M | | | | | | | |
| DCC2 | 4 | 10.5K | | | | | | | |
| | GPU | 144K | | | | | | | |
| WPA-PSK | 4 | 12.9K | | | | | | | |
| | GPU | 172K | | | | | | | |
| BCRYPT | 4 | 5.44K | | | | | | | |
| | GPU | 5.75K | | | | | | | |

View saved benchmark

Stop

Slika 9: Hashsuit mogućnosti algoritama (Izvor: https://hashsuite.openwall.net/tt_benchmark.png )

Hash Suite 3.7 [64 Bits] [Pro]

Main View Params Hardware Reports Downloader Rules

Cache: OFF CPU: 14/16 GPU: 2/2 Style About

Start Stop Resume — Attack

NTLM — Format

Charset Phrases Wordlist DB Info Keyboard LM2NT — Key Providers

Rate: 16.9G Time: 00:00:44 Keys Tested: 742,912,080,729
Load: 739 End In: 00:00:00 Key Space: 742,912,080,729
Done: 100% All Time: 00:00:44 Last Save: 00:00:44 — Attack Status

ABC Current

Charset 0-6 [95 LUDS]

Key Provider Params

Charset Params
Minimum Size 0
Maximum Size 6
Use rules
Add new charset
Lower abcdefghijklmnopqrstu...
Upper ABCDEFGHIJKLMNOPQ...
Digit 0123456789
Symbol !@#$%^&*()-_+=~`[]{}|...
Wordlist Params
Keyboard Params
Phrases Params
DB Info Params
LM2NT Params
Rules

| Username | Hash | Cleartext |
|---|---|---|
| hcookis | F83C01861FDD23B4354465FE6D7F6402 | nurse |
| emcnees | C1A8D439E0906BBF241EBBC04BDB424C | ???????????????????????????????? |
| hpasceri | 486EA753DEF361967B1A5E9C5D65EC18 | ???????????????????????????????? |
| ddauria | 34643FD04B90614EF2E9A3E1DB738986 | ???????????????????????????????? |
| ncuellar | 34643FD04B90614EF2E9A3E1DB738986 | ???????????????????????????????? |
| lrothman | 85ECE392A6A543E03B0117632CCA706C | ???????????????????????????????? |
| mplance | D9D0062E7F69FBCE862E1109F034047F | ???????????????????????????????? |
| aliao | 475BD57348733E0CC223F97FCE352F30 | ???????????????????????????????? |
| glibby | F349E6F99BA4795044AF1143A7167909 | velvet |
| sganji | 7902E9B7EEF97019E29CD1979C007BC5 | 2000 |
| hmarmon | 494877A3209B0EF206A36248E245F2A2 | james |
| jdashno | 494877A3209B0EF206A36248E245F2A2 | james |
| ilanni | 1C6C3C6BAEF048DBF0C8C5646AC684B7 | ???????????????????????????????? |
| iellingboe | EE62EC5FDB60604066CAEFA22EB35483 | lizard |
| tsculley | B3EB90B3B58F1F98C784D6A8CFBB74B6 | gigi |
| trando | 5C852DAB7DCB3996D96041C4D23FC07C | violet |
| bstruve | AD65E2482DD702903EEFE4B721D3A494 | ???????????????????????????????? |
| umaclaurin | C535FCB39752F3FCD5E5F0B4799A64F2 | floyd |
| plingelbach | 1862BB03C91B597558DF7658F1284572 | bigman |
| mmihalchik | 3F5156E39D9C989C2609FD8329A46CA4 | abcde |
| df2 | 52008EE808983C65A34D4ACC56016EC6 | ???????????????????????????????? |
| aruberte | FEDB9A5965E27B5A0FA8344DF0249584 | teddy1 |
| pstank | FEDB9A5965E27B5A0FA8344DF0249584 | teddy1 |
| tbirckhead | FEDB9A5965E27B5A0FA8344DF0249584 | teddy1 |
| mvaragona | 13A89BF8D80D50185D89359F57160980 | parker |

Cracking ended. 1 / 20 Found: 620 Total: 1,359 Found: 45%

Slika 10: Rezultati nakon korištenja Hashshuite (Izvor: https://hashsuite.openwall.net/hash_suite_win10_v37.png )

### 4.3.4. Pregled dnevnika

Ma, D i Tsudik, G. (2008, str. 38) navode da su dnevnici rada kritična točka svakog sigurnog sustava zbog toga što bilježe sve važne događaje i aktivnosti, te se pomoću njih može pronaći autoritet za svaku od zabilježenih akcija, pronađe i rekonstruira slijed događanja, te uoči neautorizirani ulazak u sustav. To potvrđuju i Wack, Tracy i Souppaya (2003, str. 28) koji navode da je log review ili pregled dnevnika rada, je metoda testiranja gdje se pregledavaju dnevnički zapisi koji bilježe aktivnost sustava, aktivnosti firewalla ili aktivnost sustava za detekciju napada.

Pregled navedenih dnevničkih zapisa može donijeti jasnu sliku dali su sustavi, firewall i IDS (Intrusion Detection System, tj. sustav za detekciju napada) pravilno konfigurirani te dali pravilno provode svoju funkciju. Može se provjeriti dali blokirane aktivnosti zaista firewall blokira, te dali IDS prepoznaje sumnjive aktivnosti te dali ih na adekvatan način riješava.

Pregled dnevničkih zapisa također može donjeti jasnu sliku u kojoj mjeri sustav djeluje po propisanim sigurnosnim policama i pravilima. Ma, D i Tsudik, G. (2008, str. 39) napominju da iskusni napadači najprije osiguraju da dnevnik aktivnosti ne bilježi njihove akcije kako se u kasnijim koracima ne bi primijetile neautorizirane akcije te kako bi se sakrio identitet napadača. Stoga je važno da sigurnosne police sadrže mehanizme koje zaštićuju dnevnike rada.

Nakon pronalaska nepravilnosti u konfiguracijama treba se popraviti konfiguracija sustava kako bi djelovalo valjano, popraviti konfiguracija firewalla i IDS-a kako bi djelovali valjano te kako bi pratili preporuke sigurnosnih polica.

Oktay, T. i Sayar, A. (2017, str. 4) navode nekoliko mana metode testiranje dnevnika rada, najveća mana je što za detaljan pregled dnevničkih zapisa treba jako puno vremena i posvećenost detaljima . Jedan od glavnih problema je veliki broj različitih izvora generira puno sadržaja za puno različitih svrha. Sadržaj se također generira u različitim formatima i vremenskim formatima koje sve treba analizirati i prezentirati na jednaki način, što znači da se dodatno vrijeme troši na to izjednači format sadržaja. Vrlo često jako puno sadržaja se bilježi u vrlo kratkom vremenskom intervalu. Tako da se može zaključiti da iako ova metoda donosi puno zaključaka što se tiče sigurnosnih ranjivosti, zbog opsega i vremenskog perioda za analiziranje, ponekada je nemoguće analizirati i pronaći sve vremenske prijetnje.

### 4.3.5. Provjera integriteta

Prema Wack, Tracy, Souppaya (2003, str. 28) integrity checkers ili alati za provjeru integriteta datoteka računaju i pohranjuju „checksum“ tj. kontrolnu sekvencu te osiguravaju bazu za pohranjivanje i praćenje kontrolne sekvence za svaku datoteku od bitne važnosti.

Prema Landoll (2006, str. 265) checksum je kontrolna sekvenca koja garantira integritet datoteke te služi kao garancija za autoriziranu promjenu datoteke. Kontrolna sekvenca se može izračunati svaki puta kada za to postoji potreba i kada se želi provjeriti dali je datoteka eventualno promijenjena te dali je neautorizirano promijenjena.

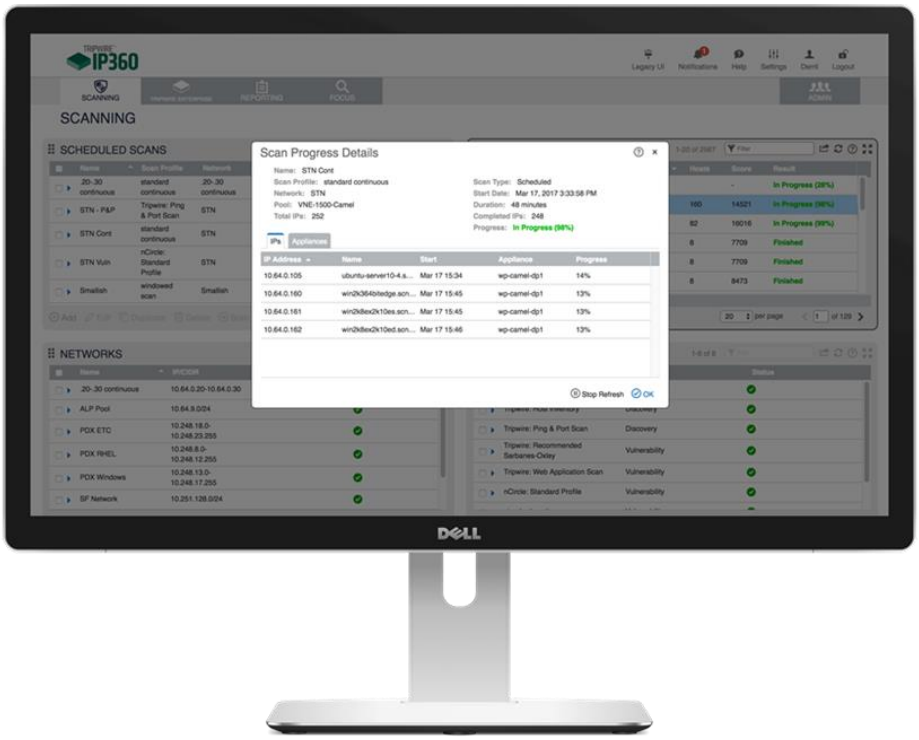
Ova metoda ima puno pozitivnih strana, no jedna negativna strana je ta što je baza kontrolnih sekvenci podložna sigurnosnim napadima te samim tim, ugrožava integritet daljnje provjere datoteka.

Wack, Tracy, Souppaya navode da je ova vrsta testiranja dobra opcija za provjeru koje datoteke su napadači promijenili no treba imati na umu da svaka autorizirana promjena datoteke zahtjeva i promjenu kontrolnih sekvenci, u protivnom će svaka sljedeća provjera checksumova nakona toga biti ne valjan, te može imati lažno negativne rezultate.

Također se mogu dogoditi i lažno pozitivni rezultati ukoliko se kontrolne sekvence računaju nakon što je sustav već kompromitiran. Stoga je bitno da se kontrole sekvence ponovno računaju nakon izdavanja nove verzije sustava, nakon promjena važnih sistemski datoteka, te nakon uspostave datoteka kako bi provjera bila validna.

Ova metoda ne daje detaljne rezultate i analizu napada ukoliko se primijete nepravilnosti već samo upućuje na datoteke koje su kompromitirane, stoga je nakon toga potrebna detaljna analiza incidenta, a ukoliko se procjeni da je incident validan i pronađe se ranjivi dio sustava, tada se može krenuti sa saniranjem štete, prilagodbom sigurnosnih polica te ponovnim računanjem kontrolnih sekvenci.

Alati za provjeru integriteta datoteke su često dio IDS-a ili mogu postojati kao opcija pri konfiguraciji baze podataka, no postoje i samostalni alati kao što su Veracity, Tripwire.




Slika 11: Grafičko sučelje Tripwire alata (Izvor: https://www.tripwire.com/-/media/tripwiredotcom/images/product-screenshots/ip360_1.png?h=668&iar=0&w=1000&rev=857d7c56f02c47a2bf729c9953dc52b2&hash=968376276679E0D0C056C3789550694E)

### 4.3.6. Detekcija virusa

Prema Landoll (2006, str. 282) Detekcija virusa je vrsta sigurnosnog testiranja gdje za to predviđeni alati testiraju mrežu ili uređaje na viruse, crve i Trojance.

Prema Wack, Tracy, Souppaya (2003, str. 29) postoje dvije vrste alata za detekciju virusa, antivirusni alati na mrežnoj infrastrukturi i antivirusni alati na korisničkim uređajima. Bilo da se radi o antivirusnim programima na mrežnoj infrastrukturi ili na korisničkim uređajima, potrebno je često provjeravati i instalirati najnovije verzije alata kako bi se osiguralo da je dostupna obrana protiv najnovijih prijetnji.

Naravno antivirusne alate na mrežnoj infrastrukturi je puno lakše odražavati zato što stručnjaci mogu preuzeti brigu o verzijama i instalaciji, za razliku od antivirusnih alata na uređajima krajnjih korisnika, gdje su korisnici sami odgovorni da na vrijeme instaliraju nove verzije antivirusnih alata.

Za najbolju zaštitu protiv virusa, crva i Trojanac najbolje je kombinirati obje vrste antivirusnih alata. Virusi, crvi i Trojanci mogu ugroziti kritične informacije, stoga je bitno da budu prisutni na mrežnoj infrastrukturi te da djeluju zajedno s firewallom i alatima s provjerim integriteta datoteka kako bi zajedno dali maksimalne rezultate.

Prva linija obrane protiv virusa na mrežnoj infrastrukturi su definitivno mail serveri s obzirom na to mogu ukloniti prijetnje prije nego naprave štete a korisnike upozoriti na potencijalne nesigurnosti što će omogućiti korisniku da uoči opasnost i ne otvara mail.

Mrežni antivirusi pružaju mnogo opcija koje će pomoći pri detekciji i zaštiti od virusa i crva kao što su „Sandboxing“, „Host Intrusion Protection“, „Virtual Desktop“, „Rescue Disc“.

Prema Landoll (2006, str. 265) Sandboxing nadgleda i autenticira procese na serveru, te sprječava da virusi i crvi naprave štetu na sustavu tako što izolira potencijalne prijetnje unutar posebne sekcije gdje se prijetnje mogu analizirati i pregledati no ne mogu imati negativan utjecaj na sustav.

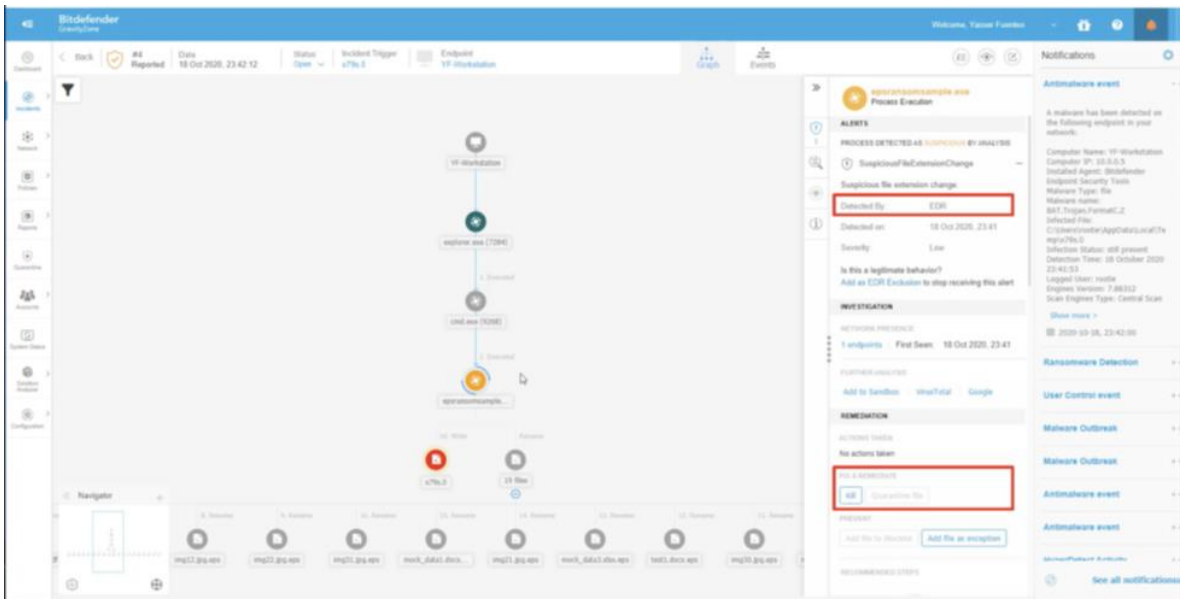
Host Intrusion Protection pruža zaštitu od neželjenih aktivnosti te automatski zaustavlja potencijalnu prijetnju prije nego što ona započinje s procesom.

Virtual desktop pruža posebno okruženje gdje se može testirati softver bez da ugrozi infrastrukturu i podatke. Rescue disc pruža mogućnosti oporavka bitnih informacija poput korisničkih računa i lozinki.

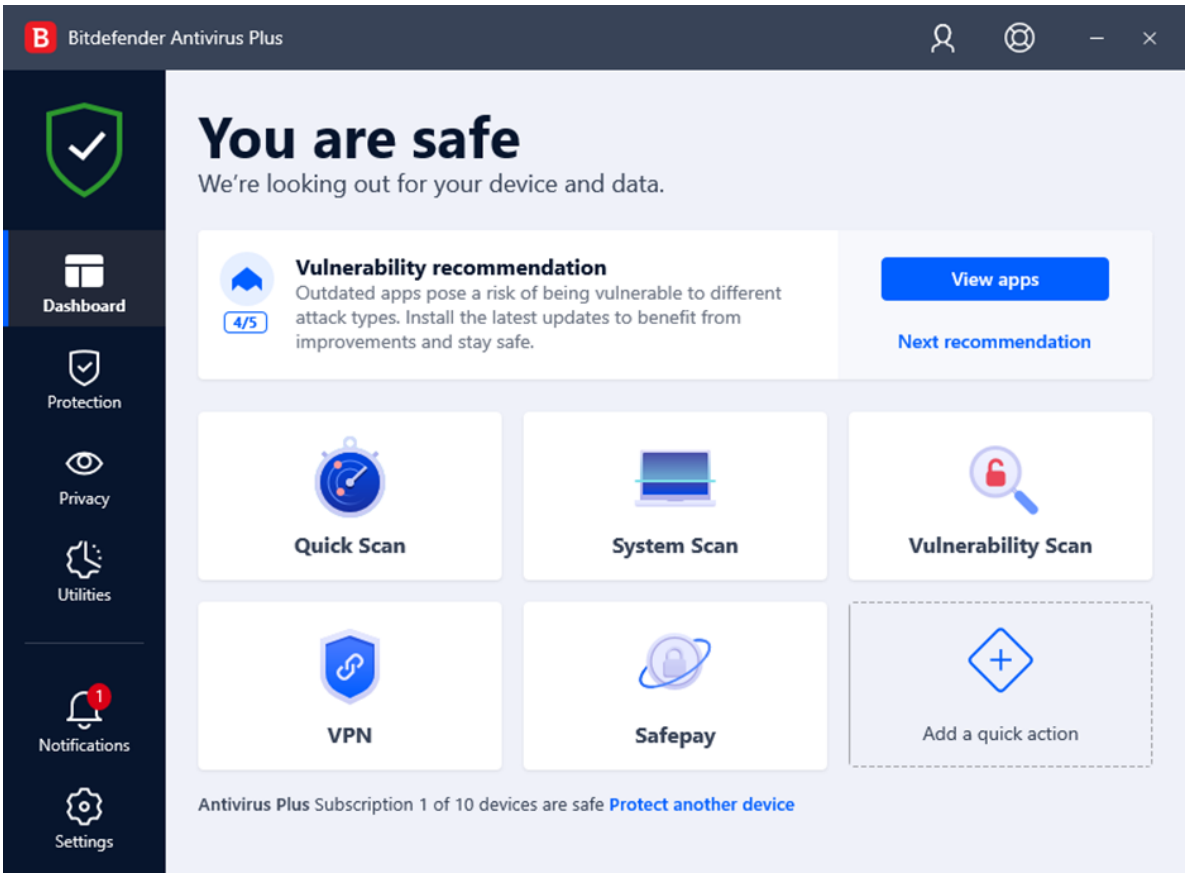
Za najbolju zaštitu i najbolje djelovanje preporučuje se na tjednoj bazi provjeravati verzije alata, te osigurati da alat uvijek radi u pozadini kako bi najbrže moguće mogao pronaći ranjivosti i ukloniti ih.

Naseer et al., (2021, str. 5) navodi mane metode detekcije virusa. Najveći izazov je definitivno veliki opseg podataka te sakupljanje podataka. Svakim danom nastaju nove vrste virusa u tolikoj mjeri da je teško fokusirati se na određen skup podataka, no proširenjem skupa podataka gubi se točnost i preciznost te raste broj lažnih pozitiva. Slijedeća mana je što se neke virusi uoče samo ukoliko se prezentiraju kao anomalija, a virusi koje se ne prezentiraju na taj način su teško uočljivi i često neprimjetni. Zbog velikog opsega podataka te brojnih vrsta virusa, treba osigurati i dovoljne resurse i spremišta koji će moći podržati tolike količine obrade i podataka.

Najpoznatiji antivirusni alati na mrežnoj infrastrukturi su: Bitdefender GravityZone Endpoint Security, Comodo Server Antivirus, Avira Antivirus Server, Kaspersky Endpoint Security Cloud Plus, ESET Endpoint Protection Standard, AVG File Server Business Edition. Dok su na korisničkim uređajima najpoznatiji antivirusni alati: Bitdefender, Norton, Intego, Panda



Slika 12: Grafičko sučelje GravityZone alata




Slika 13: Grafičko sučelje Bitdefender alata (Izvor: https://i.pcmag.com/imagery/reviews/04rK8yQu2cIXBMdJVdlQRnz-29..1595951943.png )

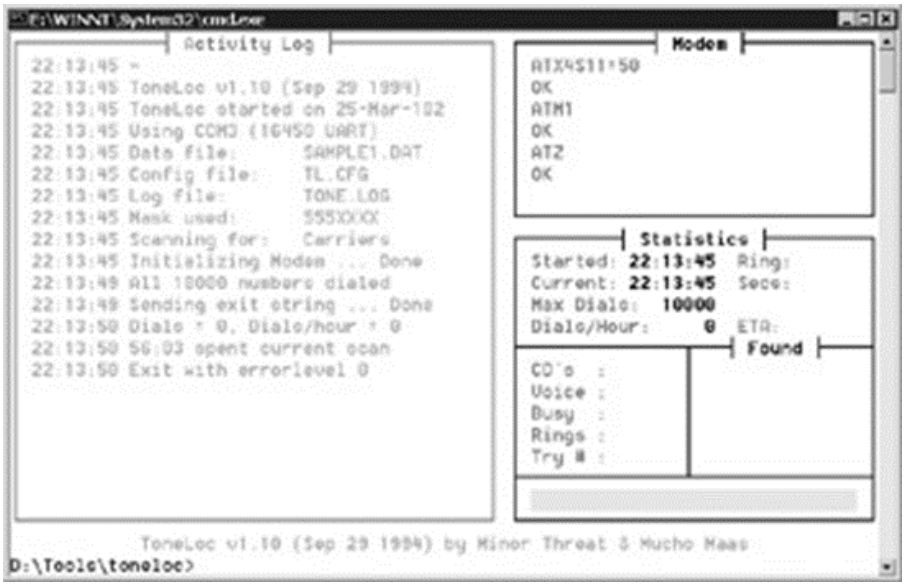
## 4.3.7. War Dialing

Prema Wack, Tracy, Souppaya (2003, str. 30) „War Dialing“ je proces gdje se testira veliki broj brojeva telefona organizacije kako bi se našao broj telefona koji se spojen s modemom te se na taj način može dobiti pristup mreži.

Velike organizacije često imaju određeni broj brojeva telefona koji počinju istim prefiksom, kada se uzme u obzir kombinacije broja može se dobiti skala brojeva telefona koja se potom testira. Prema Landoll (2006, str. 46) ukoliko je ijedan broj s te skale spojen na modem, napadači mogu pokušati pristupiti mreži a potom i sustavu.

Velike organizacije jako često zanemaruju ovu vrstu prijetnje te su podložne ovom načinu napada. Ukoliko je modem spojen na računalo i napadači dobiju pristup računalu cijelu sustav može biti kompromitiran iako je uspostavljena dobra sigurnosna praksa. Alati koji se koriste za war dialing su open source, te postoji nekoliko vrsta kao što su ToneLoc, PhoneTap, Blue Deep.

Slika 14: ToneLoc opcije za skeniranje (Izvor: https://flylib.com/books/3/85/1/html/2/images/fig18_05.jpg)




Slika 15: Rezultati Tone Loc skeniranja (Izvor: https://flylib.com/books/3/85/1/html/2/images/fig18_06.jpg)

### 4.3.8. War Driving

Singh (2014, str. 2) navodi da je cilj metode testiranja „war driving“ prikupljanje podataka i informacija o broju pristupnih točaka koje imaju implementiranu slabu sigurnosnu konfiguraciju ili uopće nemaju implementirane nikakve sigurnosne mehanizme.

Prema Landoll (2006, str. 282) testiranje bežičnog LAN-a, ili mapiranje bežične točke pristupa (engl. War Driving) se bazira na vožnji oko mogućnih meta te pokušaju da se spoje da otvorene točke pristupa.

Interes oko ovo metode testiranja je porastao zbog velikog broja pristupnih točaka, posebice zbog toga što je svake godine sve veći i veći broj pristupnih točaka, a svjesnost oko potrebne sigurnosne zaštite još uvijek nije na zadovoljavajućoj razini što omogućuje napadačima da pomoću ove metode krenu u „lov“ na nesigurne pristupne točke.

Singh (2014, str. 2) navodi da je metodu izmislio „Peter Shipley“ na „Defconu“ 2001. godine gdje je prezentirao „war driving“ metodu zajednici.

Napadači koriste bežične mrežne kartice, te dodatnu opremu kako bi pojačali signali do mrežnih kartica. Ovu vrstu ranjivosti je bitno detaljno testirati jer 802.11b ima mnogo poznatih ranjivosti, a često je loše i nestručno konfiguriran što doprinosi povećanju sigurnosnog rizika jer su mnoge metode za napad već dobro poznate.

Nakon što testiranje pomoću War Driving pokaže da je mreža podložna napadu, potrebno je popraviti mrežnu konfiguraciju uređaja te popraviti sigurnosne police kako bi preporuke za konfiguriranje drastično smanjile sigurnosni rizik.

Najpoznatiji alati za war driving su: Stumbler, InSSIDer, Kismet, KisMAC, NetSpot, NetStumbler itd.

### 4.3.9. Penetracijsko testiranje

Mishra (2021, str.2) navodi da sigurnosni napadi kontinuirano rastu a metode i alati postaju sve moćniji i snažniji, te mogu iskorištavati sve više vrsta različitih sigurnosnih napada.

Potreba za znanjem o penetracijskom testiranju raste, no potrebno je dobro analizirati i osmisliti metode za izvođenje penetracijskog testiranja. Potrebno je da metode budu razne, te da se koriste konvencionalne i nekonvencionalne metode te da pokriju sve najnovije vrste ranjivosti. Što je više različitih vrsta ranjivosti potrebno je koristiti više različitih alata koji imaju različite mogućnosti i opcije.

Prema Stewart, Chapple, Gibson (2015) penetracijsko testiranje je tip testiranja gdje je cilj identificirati ranjivosti sustava tako što se kroz ulogu napadača testiraju kritične točke sustava kroz koje bi se mogla preuzeti kontrola nad sustavom ili preuzeti kritične informacije sustava.

Ahmad, W. (2020, str.5) navodi da je penetracijsko testiranje autorizirani simulirani cyber napad na sustav, koji se izvodi kako bi se procijenila sigurnost sustava

Testeri koriste ilegalne alate i metode kako bi pokušali doći do kontrole sustava te na taj način identificirati ranjivosti sustava no sustav nikada neće biti potpuno siguran i neprobojan unatoč penetracijskom testiranju. Svaki napadač može imati svoje metode i alate kroz koje će pokušati pridobiti kontrolu nad sustavom te može imati drugačije rezultate od rezultata na sigurnosnom testiranju.

Penetracijsko testiranje je važno dobro isplanirati u fazi procjene sigurnosnog rizika te tažiti dopuštenje i odobrenja za sve korištene alate i metode. Treba se analizirati i isplanirati korištene identitet testera koji će provoditi penetracijsko testiranje, njegova ip adresa, metode, tehnike i alati.

Također je važno vrijeme provođenja penetracijskog testa te koliko dugo će se test provoditi, zato što se preporuča da to bude i vrijeme kada je najmanje zaposlenika na mreži, no sistemski administratori moraju biti obaviješteni kako bi mogli razlikovati stvarni sigurnosni napad od testiranja.

Ahmad, W. (2020, str.7) navodi da se svako penetracijsko testiranje izvodi u fazama: aktivnosti prije izvođenja napada (u toj fazi se definiraju metode, ciljevi i očekivanja), prikupljanje informacija (prikupljaju se informacije o vrsti sustava), enumeracija (u toj fazi se

provodi mrežno skeniranje i analiziraju rezultati), dobivanje pristupa, eskalacija privilegija, održavanje pristupa, prikrivanje napada, te izvještavanje o napadu.

Prema Stewart, Chapple, Gibson (2015) cilj penetracijskog testiranja je pronaći ranjivosti sustava, identificirati sposobnosti zaposlenika da identificiraju napad, te uspostaviti dodatne kontrole zaštite protiv napada.

# 5. Sigurnosno testiranje na realnom informacijskom sustavu

Informacijski sustav koji je potrebno testirati je sustav Hotela Blue Room koji je jedan u lancu hotelu. Hotel koristi desktop aplikaciju za izdavanje računa te evidenciju soba, gostiju i inventara. Svi podaci se na skladište na Microsoft Azure cloud.

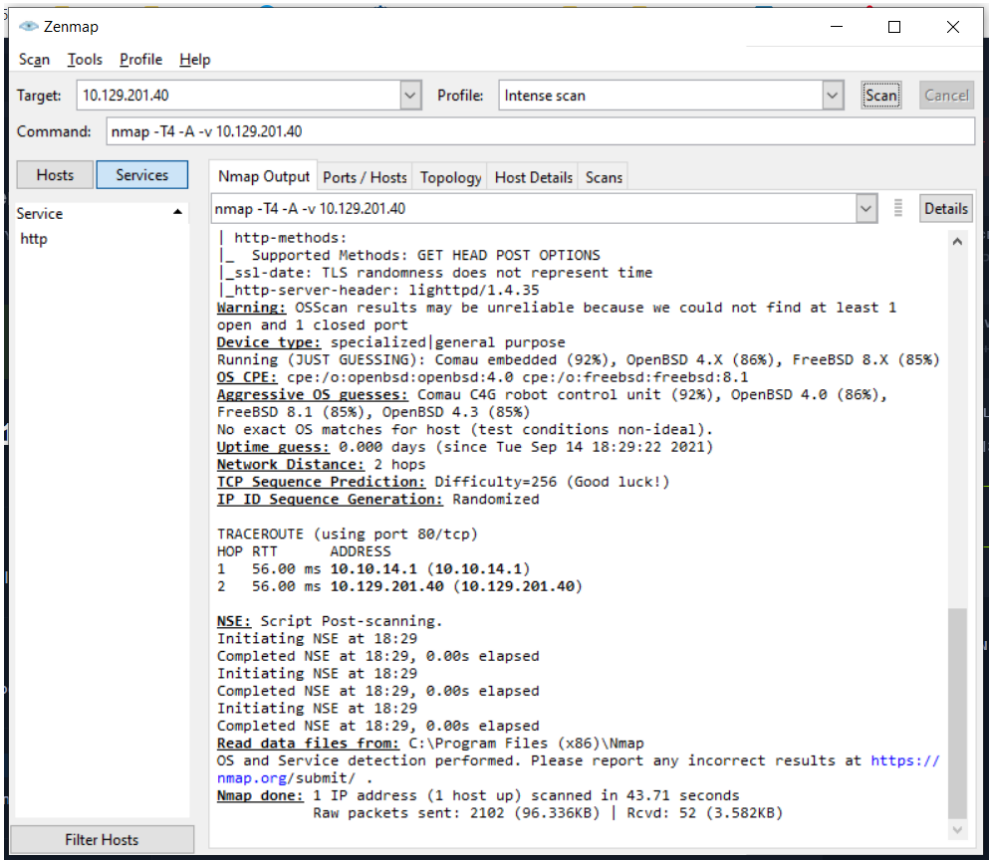
Gosti hotela imaju zasebnu mrežu no zbog prijašnjih sigurnosnih napada, tvrtka se odlučila na uvođenje menadžment sigurnosnog rizika kako bi poboljšala sigurnosno stanje u tvrtki. Zaposlenici su prošli edukacije o sigurnosti, a sigurnosne police i kontrole su uređene, no zbog sigurnosnih napada ponovno je rađena procjena sigurnosnog rizika kako bi se sigurnosne police i kontrole uredile.

U sklopu procjene sigurnosnog rizika procijenjeno je da je mrežni promet kritična točka. Poznato je da je implementiran firewall koji nadgleda i filtrira mrežni promet no napadači su nekoliko puta uspjeli zaobići firewall te provaliti u sustav.

Slijedeća faza je sigurnosno testiranje gdje će fokus biti mrežni promet te pronaći ranjivost tj. rupu u sustavu koju napadači iskorištavaju, kako bi se u slijedećim fazama mogli izvesti popravci sigurnosnih konfiguracija, prilagodba alata te dodavanje novih pravila unutar sigurnosnih polica.

Upravo zbog toga prvi korak će biti mrežno skeniranje i detaljna analiza prometa. Tester treba biti sigurnosni ekspert koji će moći protumačiti rezultate testiranja. U sigurnosnom testiranju se predviđa da će se koristiti alati za skeniranje mreže, alati za brute force napad te ostali alati koji će biti potrebni ovisno o rezultatima sigurnosnog testiranja.

Sigurnosno testiranje započinje metodom „network scanning“, tj. skeniranjem mreže. To je prvi korak u sigurnosnom testiranju nakon kojeg će biti poznato više informacija o mogućim ranjivostima sustava. Za skeniranje mreže može se koristiti bilo koji dostupan alat, no u ovom slučaju će se koristiti nmap koji dolazi s grafičkim sučeljem te se naziva zenmap. Potrebno je pokrenuti skeniranje mreže nakon čega će informacije o mreži biti dostupne.



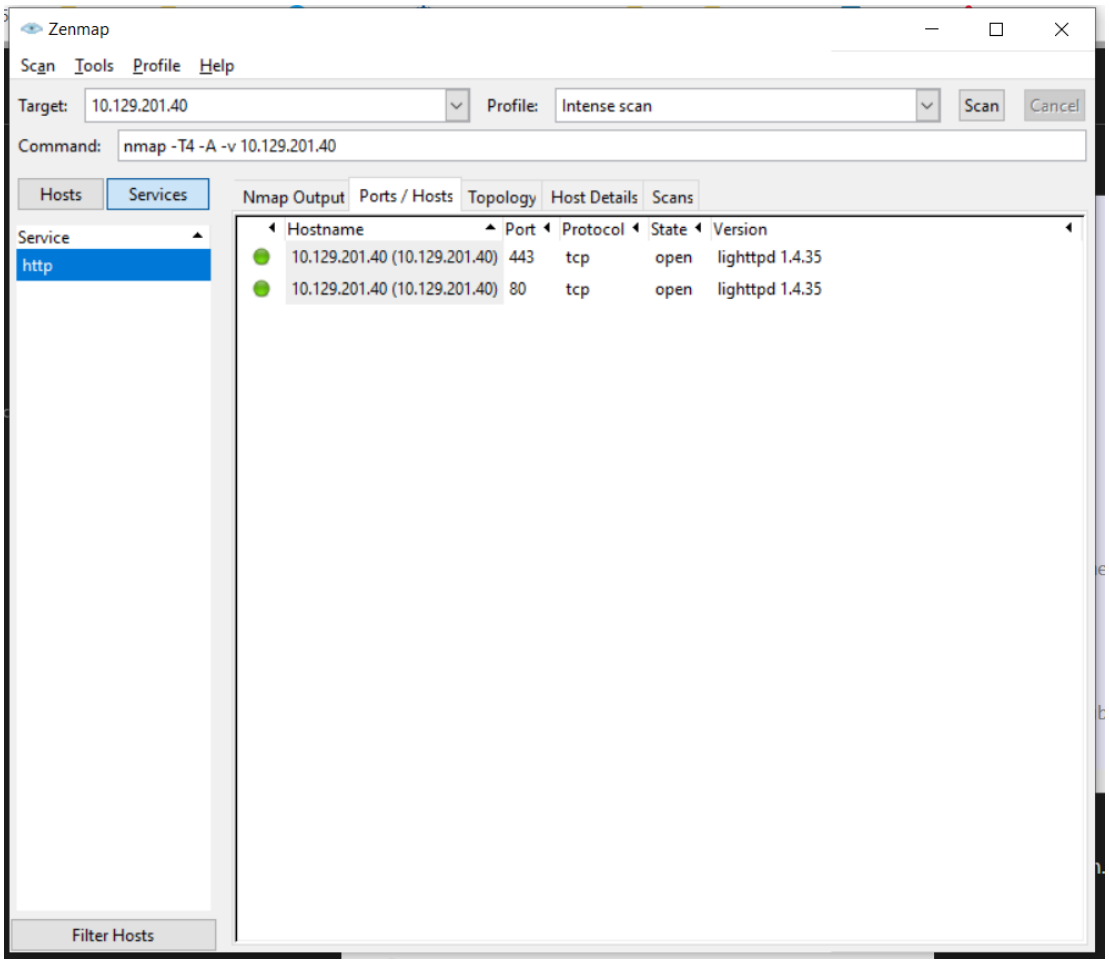

Slika 16: Zenmap analiza i prikaz detalja skeniranja

Zenmap nakon skeniranja pokaže informacije koje je pronašao. Prikazane su informacije o mogućoj vrsti operacijskog sustava: Comau embedded (92%), OpenBSD 4.X (86%), FreeBSD 8.X (85%).

Mrežni skeneri ne daju potpuno točne informacije o vrsti operacijskog sustava već je potrebno detaljnije interpretirati . U ovom slučaju radi se o OpenBSD operacijskom sustavu te je potrebno provjeriti sustav na poznate ranjivosti OpenBSD-a.

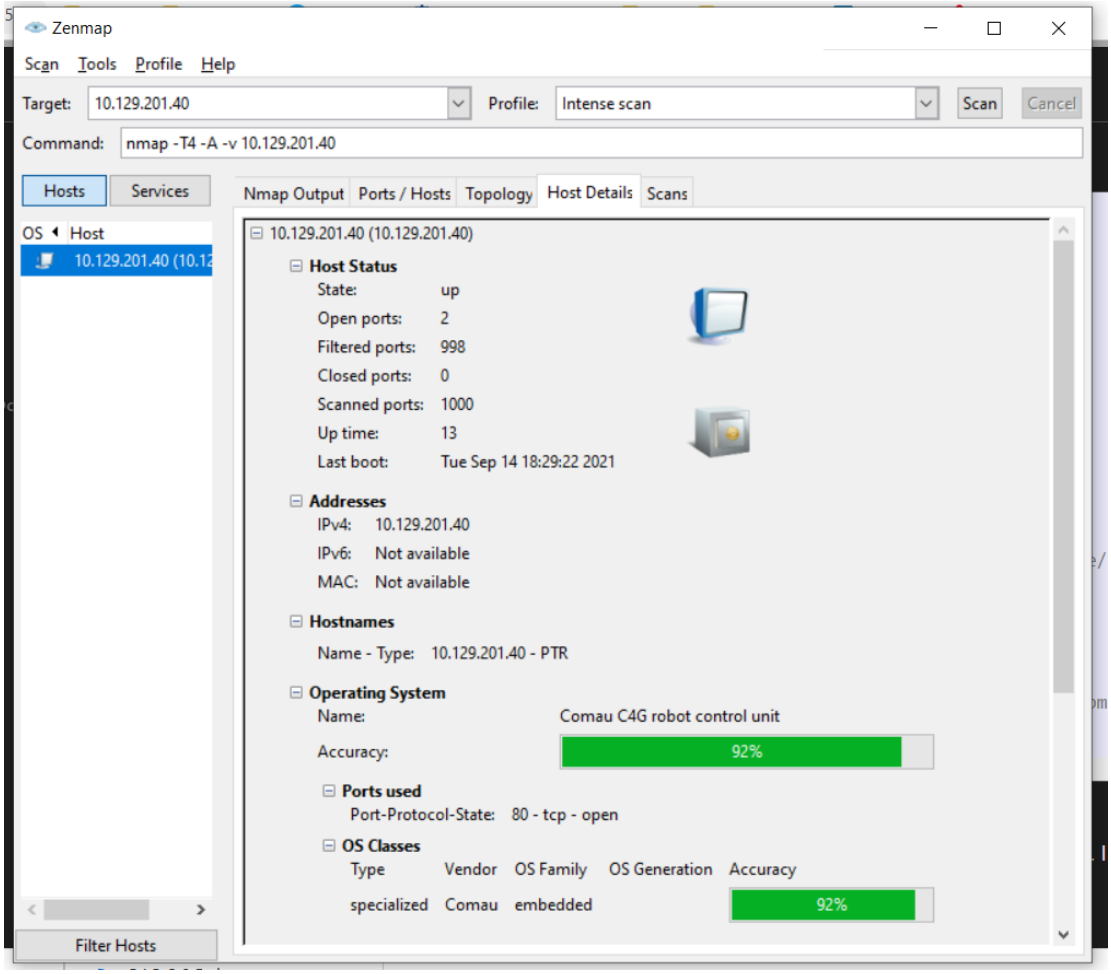
Na stranici https://www.cvedetails.com/vulnerability-list/vendor_id-97/product_id-163/Openbsd-Openbsd.html ili nekoj od sličnih stranica može se naći lista ranjivosti s kratkim opisom i informacijama o ranjivostima.

Potrebno je provjeriti dali se neka od navedenih ranjivosti može iskoristiti u sljedećim koracima. Ukoliko ne može, potrebna je daljnja analiza rezultata.



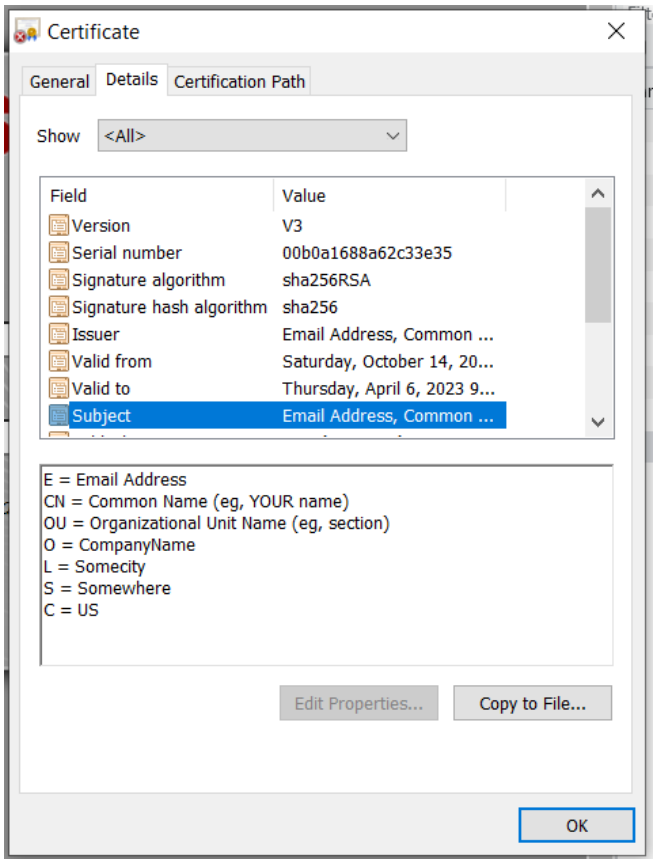

Slika 17: Prikaz Zenmap skeniranih poslužitelja

Slijedeća stvar koja se može primijetiti je da su otvorena dva porta 443 i 80, što znači da je aktivna web aplikacija, port 80 je obično korišten od strane web aplikacija za HTTP protokol dok je port 443 korišten za HTTPS protokol.




Slika 18: Zenmap rezultati i analiza poslužitelja

U sekciji korištenih portova vidi se da je port 80 korišten stoga je slijedeći korak inspekcija web aplikacije




Slika 19: Detalji o SSL certifikatu aplikacije

Daljnjom inspekcijom otkriveno je da SSL certifikat nije valjan, te je potrebno što prije instalirati valjani SSL certifikat s obzirom da je trenutno sustav podložan brojnim ranjivostima. Instaliranje i održavanje SSL certifikata je nadalje potrebno uvrstiti u sigurnosnu policu, a eventualne konfiguracije je potrebno urediti.

Slika 20: Grafičko sučelje pfsense administratoskog sučelja

Otvaranjem aplikacije i analizom otkrilo se da se radi o administratorskom sučelju za „pfSense“ firewall i ruter. Slijedeći korak je pokušati probiti korisničko ime i lozinku te ući u administratorsko sučelje.

```
/index.php (Status: 200)
/help.php (Status: 200)
/themes (Status: 301)
/stats.php (Status: 200)
/css (Status: 301)
/edit.php (Status: 200)
/includes (Status: 301)
/license.php (Status: 200)
/system.php (Status: 200)
/status.php (Status: 200)
/javascript (Status: 301)
/changelog.txt (Status: 200)
/classes (Status: 301)
/exec.php (Status: 200)
/widgets (Status: 301)
/graph.php (Status: 200)
/tree (Status: 301)
/wizard.php (Status: 200)
/shortcuts (Status: 301)
/pkg.php (Status: 200)
/installer (Status: 301)
/wizards (Status: 301)
/xmlrpc.php (Status: 200)
/reboot.php (Status: 200)
/interfaces.php (Status: 200)
/csrf (Status: 301)
/system-users.txt (Status: 200)
/filebrowser (Status: 301)
```

Slika 21: Slika Gobuster detalja brute force napada

Pomoću Gobuster alata izvršen je brute force napad na aplikaciju kako bi se dobio prikaz svih skrivenih sadržaja. Iz priloženog se vidi da su dostupne changelog.txt i system-users.txt

Name | Headers | Preview | Response | Initiator | Timing | Cookies

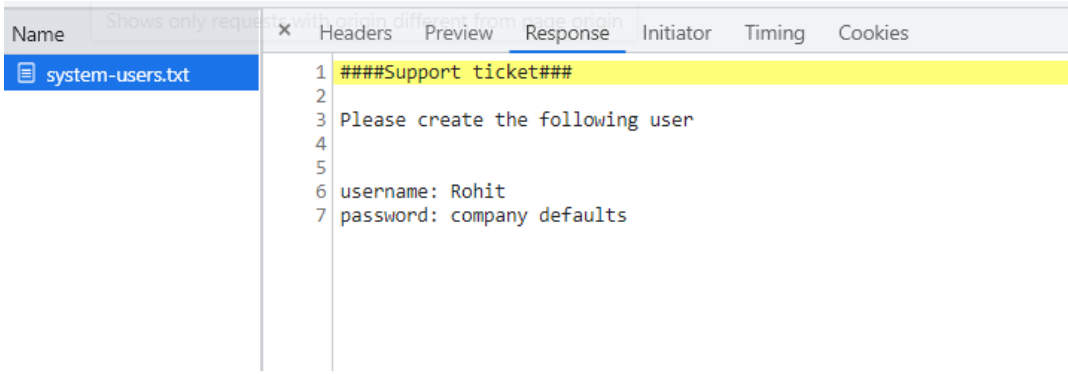
changelog.txt

```
1  # Security Changelog
2
3  ### Issue
4  There was a failure in updating the firewall. Manual patching is therefore required
5
6  ### Mitigated
7  2 of 3 vulnerabilities have been patched.
8
9  ### Timeline
10 The remaining patches will be installed during the next maintenance window
```

Slika 22: Prikaz changelog sadržaja datoteke

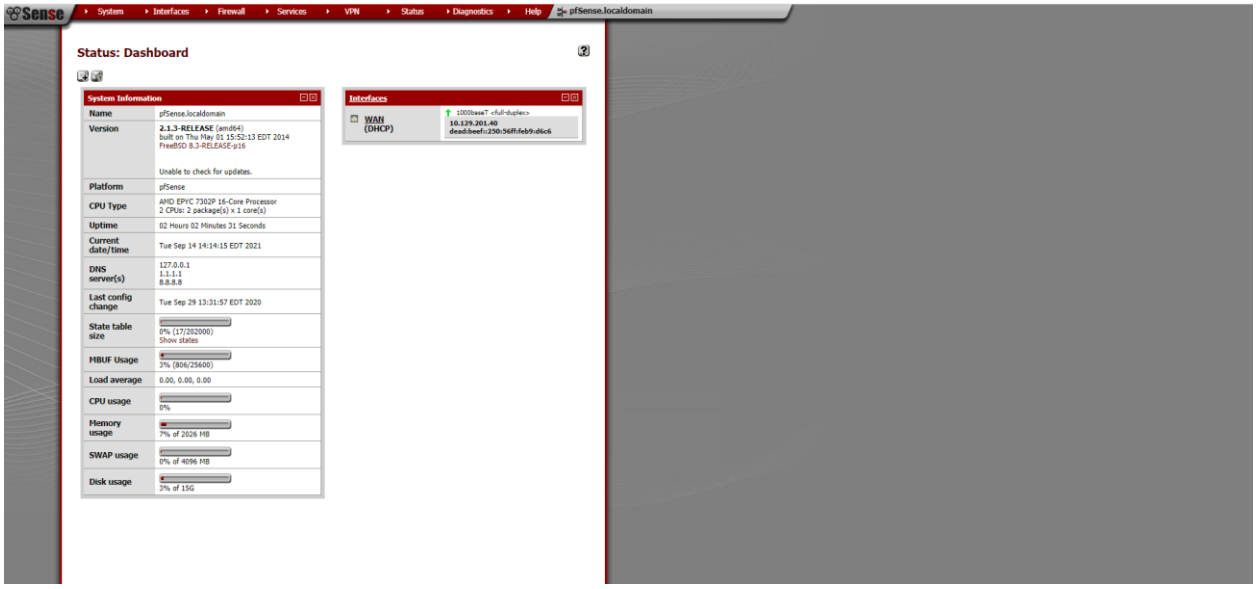
Iz priloženog se vide detalji iz changelog-a što predstavlja sigurnosnu ranjivost te bi se trebalo spriječiti pristupanje changelog.txt-u bez validne autorizacije.

Iz priloženog se vidi kako sigurnosne konfiguracije nisu dobro posložene te je potrebno provjeriti i konfigurirati pristupe do kritičnih datoteka te uvrstiti u sigurnosnu policu promjene koje je potrebno provesti kako bi kritične datoteke bile pravilno zaštićene.




Slika 23: Prikaz system-users sadržaja datoteke

Iz system-users.txt se vidi da je jedan od korisničkih računa vidljiv te da mu je lozinka standardna lozinka za pfsense firewall administratorsko sučelje. Stoga se pretpostavlja da je jedan od korisničkih računa rohit/pfsense




Slika 24: Prikaz glavne stranice administratoskog sučelja

U sljedećem koraku nakon uspješne autorizacije, vidi se pristup administratorskom sučelju te korisnik može zaustaviti firewall ili promijeniti sigurnosne postavke.

Važno je napomenuti da je sigurnosno testiranje na stvarnom informacijskom sustavu puno detaljnije i preciznije, najčešće se provode sve metode testiranja, te se uzimaju u obzir analize iz faze procjene sigurnosne ranjivosti.

# 6. Zaključak

Sigurnosno testiranje je dio procesa menadžmenta sigurnosnog rizika koji počinje s procjenom sigurnosnog rizika. Bez valjane procjene sigurnosnog rizika, sigurnosno testiranje neće biti smisleno i kompletno te neće biti usmjereno na konkretne sigurnosne rizike organizacije. U procjeni sigurnosnog rizika definiraju se svi bitni parametri kako bi se sigurnosno testiranje moglo provesti. Važno je analizirati postojeće sigurnosne police i pravila kako bi se odredio cilj i smjer sigurnosnog testiranja. Česte konfuzije oko pojmova koji su ne rijetko sinonimi jedan za drugog ponekada otežava proces sigurnosnog testiranja jer se u fazi procjene sigurnosnog rizika nije jasno odredilo što točno označava pojam sigurnosnog testiranja za tu organizaciju te koja su njihova očekivanja od procjene rizika i sigurnosnog testiranja. Jednom kada se jasno analiziraju ciljevi, i očekivanja smjer sigurnosnog testiranja je puno jasniji.

U procjeni sigurnosnog rizika vrlo je važno i definirati sve važne parametre sustava kao što su digitalna oprema, vrsta i povezanost sustava i detalji o mreži i korištenim aplikacijama. Nakon definiranja vrsta testiranja vrlo je važno dobiti dopuštenja za izvođenje planiranih vrsta testiranja. Važno je definirati vrste testova koji će se izvoditi, popis alata koji će se koristiti, ip adresu s koje će doći „napadi“ te očekivani utjecaj na sustav (sporost ili rušenje sustava.

Faza sigurnosnog testiranja se nakon procjene sigurnosnog rizika ponavlja često, te ovisno o rezultatima dolazi do uvođenja novih sigurnosnih polica, popravaka ili patch verzija aplikacija te uređivanja sigurnosne konfiguracije. Svaka od metoda sigurnosnog testiranja se ponavlja ovisno o kompleksnosti i zahtjevima sustava te korelaciji s prijašnjim rezultatima metode sigurnosnog testiranja.

Važno je istaknuti da čak i sustavi s dobrim rezultatima na sigurnosnim testovima nisu sto posto zaštićeni, te da nikad neće biti savršeno sigurni, no bitno je inkrementalno poboljšavati razinu sigurnosti sustava te kontinuirano raditi na sigurnosnim procjenama i sigurnosnom testiranju kako opseg sustava rasta. Važan aspekt na kojem menadžment sigurnosnog rizika mora kontinuirano raditi je edukacija zaposlenika o sigurnosnim policama i pravilima te usvajanje novih i sigurnosnih polica kako sustav napreduje i raste.

# Popis literature


Landoll (2006). *The security risk assessment handbook. A complete guide for performing security risk assessments.* USA: CRC Press.

Gregg. *CompTIA Advanced Security Practitioner study guide. Second Edition*. Canada: Sybex

Wack, Tracy, Souppaya (2003). *Computer security Guideline on Network Security Testing, Recommendations of the National Institute of Standards and Technology.* Gaithersburg: Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology

Felderer, Johns, Buchler, Breu (2016). *Security Testing: A Survey.* doi: 10.1016/bs.adcom.2015.11.003.

OWASP (2014). *Testing Guide version 4.0*

Scarfone, Souppaya, Cody, Orebaugh (2008). *Computer security: Technical Guide to Information Security Testing and Assessment.* USA: National Institute of Standards and Technology

Barcelo, Herzog (2010). *The Open Source Security Testing Methodology Manual.* Institute for security and open methodologies.

Stewart, Chapple, Gibson (2015). *Certified Information Systems Security Professional.* USA: Sybex

Rudy J., Rodwald P. (2020). *Job Scheduling with Machine Speeds for Password Cracking Using Hashtopolis.* doi: 10.1007/978-3-030-48256-5_51.

Ahmad, W. (2020). *Introduction to penetration testing 5-15. doi: 10.13140/RG.2.2.25396.27525.*

Mishra, Shailendra. (2021). *Efficacy of Unconventional Penetration Testing Practices. Intelligent Automation and Soft Computing. 31. 223-239. doi: 10.32604/iasc.2022.019485.*

Singh, Vrijendra. (2014). *Analysis of Current Wi-Fi Security Practices via War Driving and Proposed Solution. International Journal of Advanced Computational Engineering and Networking.* 2. 45-49.

Naseer, M.& Rusdi, J. & Shanono, N. & Salam, S. & Zulkiflee, M. & Abu, N. (2021). *Malware Detection: Issues and Challenges. Journal of Physics: Conference Series*. 1807. 012011. doi: 10.1088/1742-6596/1807/1/012011.

Heena, & Mehtre, B.. (2021). Advances In Malware Detection- An Overview.

Liao, S.& Zhou, C. & Zhao, Y. & Zhang, Z. & Zhang, Chengwei & Gao,Y. & Zhong, G. (2020). *A Comprehensive Detection Approach of Nmap: Principles, Rules and Experiments.* 64-71. doi: 10.1109/CyberC49757.2020.00020.

Ma, D & Tsudik, G. (2008). *A New Approach to Secure Logging*. 5094. 48-63. doi: 10.1007/978-3-540-70567-3_4.

Jhala, N. (2014). *Network Scanning & Vulnerability Assessment with Report Generation.*

Oktay, T. & Sayar, A. (2017). *Analyzing Big Security Logs in Cluster with Apache Spark.* 131-138. doi: 10.1007/978-3-319-47898-2_14.

Vaiyan (2010). *https://www.computerworld.com/article/2518328/heartland-breach-expenses-pegged-at--140m----so-far.html*

TrendMicro (2016). *https://www.trendmicro.com/vinfo/us/security/news/cyber-attacks/2012-linkedin-breach-117-million-emails-and-passwords-stolen-not-6-5m*

Weise. https://eu.usatoday.com/story/tech/2016/05/31/360-million-myspace-accounts-breached/85183200/

BBC (2013) https://www.bbc.com/news/technology-24740873

O'Donnel (2013). https://threatpost.com/2014-marriott-data-breach-exposed-500m-guests-impacted/139507/

Reuters CNBC (2014). https://www.cnbc.com/2014/05/22/hackers-raid-ebay-in-historic-breach-access-145-mln-records.html

BBC (2015). https://www.bbc.com/news/technology-34311203

Goud (2018). https://www.cybersecurity-insiders.com/over-150-million-users-of-myfitnesspal-affected-by-cyber-attack/

Dutta (2020) https://codeburst.io/inside-canvas-security-breach-that-affected-139-million-user-accounts-78467e315681

# Popis slika

Slika 1. Hotel Mariott (Izvor: https://media.threatpost.com/wp-content/uploads/sites/103/2018/11/30083518/bmimc-exterior-0001-hor-feat-e1543584929332.jpg ) ..... 3
Slika 2. Ebay logo (Izvor: https://www.trefis.com/stock/xlk/articles/240829/ebay-suffers-hack-attack/2014-05-28) ..... 3
Slika 3: Prikaz nmap ispisa nakon skeniranja (Izvor: https://nmap.org/zenmap/images/zenmap-no-648x700.png ) .....13
Slika 4: Prikaz detalja o poslužitelju (Izvor: https://nmap.org/zenmap/images/zenmap-hd-648x700.png ) .....14
Slika 5: Prikaz Wireshark sučelja (Izvor: https://www.wireshark.org/docs/wsug_html/wsug_graphics/ws-main.png).....15
Slika 6: Nessus sučelje i rezultati skeniranja (Izvor: https://www.tenable.com/sites/all/themes/tenablefourteen/img/nessus/nessus-vulnerability-grouping-and-snoozing_large.png) .....17
Slika 7: Nessus prikaz sadržaja (Izvor: https://media-s3-us-east-1.ceros.com/tenable/images/2018/12/04/9431d4e1871db487575704540178a97d/reporting-screen.png?imageOpt=1&fit=bounds&width=1249 ) .....17
Slika 8: John the Ripper komandne naredbe (Izvor: https://miro.medium.com/max/777/1*JpTERXiDauT2VtuJv-BthA.png) .....19
Slika 9: Hashsuit mogućnosti algoritama (Izvor: https://hashsuite.openwall.net/tt_benchmark.png ) .....20
Slika 10: Rezultati nakon korištenja Hashshuite (Izvor: https://hashsuite.openwall.net/hash_suite_win10_v37.png ).....20
Slika 11: Grafičko sučelje Tripwire alata (Izvor: https://www.tripwire.com/-/media/tripwiredotcom/images/product-screenshots/ip360_1.png?h=668&iar=0&w=1000&rev=857d7c56f02c47a2bf729c9953dc52b2&hash=968376276679E0D0C056C3789550694E) .....23
Slika 12: Grafičko sučelje GravityZone alata .....25
Slika 13: Grafičko sučelje Bitdefender alata (Izvor: https://i.pcmag.com/imagery/reviews/04rK8yQu2cIXBMdJVdlQRnz-29..1595951943.png ) .25
Slika 14: ToneLoc opcije za skeniranje (Izvor: https://flylib.com/books/3/85/1/html/2/images/fig18_05.jpg) .....26
Slika 15: Rezultati Tone Loc skeniranja (Izvor: https://flylib.com/books/3/85/1/html/2/images/fig18_06.jpg) .....27
Slika 16: Zenmap analiza i prikaz detalja skeniranja.....31
Slika 17: Prikaz Zenmap skeniranih poslužitelja .....32
Slika 18: Zenmap rezultati i analiza poslužitelja.....33
Slika 19: Detalji o SSL certifikatu aplikacije .....34
Slika 20: Grafičko sučelje pfsense administratoskog sučelja .....35
Slika 21: Slika Gobuster detalja brute force napada .....36
Slika 22: Prikaz changelog sadržaja datoteke .....36
Slika 23: Prikaz system-users sadržaja datoteke.....37
Slika 24: Prikaz glavne stranice administratoskog sučelja .....37