A black background with red text

Description automatically generated

***IT381 Zaštita i bezbednost informacija***

***Domaći zadatak br. 7***

***Ime i prezime: Jovan Pavkovic***

***Broj indeksa: 4442***

**Tekst Zadatka**

Za rešenje zadatka potrebno je:

1. Nessus Zadatak:

Nessus je alat za skaniranje računarskih sistema i upozoravanje u slučaju otkrivanja ranjivosti koje zlonamerni mogu iskoristiti za dobijanje pristupa bilo kojem računaru koji je povezan na mrežu.

Nessus izvršava oko 1200 provera na datom računaru. Testira da li se neki od poznatih napada mogu iskoristiti za dobijanje pristupa tom računaru.

Cilj ovog zadatka je upoznavanje sa Nessus i Metasploit alatima za skaniranje ranjivosti računarskih sistema.

* Preuzeti Nessus
* Instalirati Nessus na Windows ili Linux mašini
* Skanirati svoju mašinu
* Skanirati okolinu.

Odgovoriti:

* Na kojoj je računarskoj arhitekturi baziran Nessus?
* Koja je prednost ovakve arhitekture?
* Koliko se Nessus razlikuj od alata nmap?
* Šta je to Nessus „plugin“?
* U kojem se formatu mogu napisati rezultati Nessus analize.
* Izlistajte i detaljno objasnite koje ste ranjivosti detektovali.
* Zašto se biraju non-DoS skeneri ranjivosti?

**Rešenje zadataka**

1. **Na kojoj je računarskoj arhitekturi baziran Nessus?**

Nessus je softver za skeniranje ranjivosti koji je razvijen od strane Tenable, Inc. Na osnovu informacija dostupnih do mog poslednjeg ažuriranja u januaru 2022. godine, Nessus je komercijalni softver i dostupan je za različite računarske arhitekture, uključujući x86 i x86-64 (32-bitni i 64-bitni sistemi).

Baziran je na klijent-server arhitekturi.

A diagram of a computer system

Description automatically generated

1. **Koja je prednost ovakve arhitekture?**

Klijent-server arhitektura je pristup organizaciji i dizajnu softverskih sistema u kojem se funkcionalnosti sistema distribuiraju između računara ili uređaja, pri čemu jedan entitet (klijent) traži usluge, a drugi entitet (server) pruža usluge. Ovaj pristup donosi nekoliko prednosti, uključujući:

Podela zaduženja (separacija odgovornosti): Klijent-server arhitektura omogućava jasnu podelu zaduženja između klijenta i servera. Klijent je odgovoran za korisnički interfejs i prezentaciju, dok je server odgovoran za obradu podataka i poslovnu logiku. Ova jasna podela olakšava održavanje i razvoj sistema.

**Skalabilnost**: Klijent-server arhitektura omogućava skalabilnost sistema. Serveri se mogu skalirati kako bi podržavali povećani broj klijenata. Ovo olakšava prilagođavanje sistema rastućim zahtevima bez potrebe za promenom klijentske strane.

**Centralizovano upravljanje i kontrola**: Server obično sadrži centralizovane resurse i podatke, što omogućava efikasno upravljanje i kontrolu. Ova centralizacija pomaže u održavanju doslednosti podataka i smanjenju redundancije.

**Lakše održavanje i ažuriranje**: Ako se promene na sistemima vrše, ažuriranja i održavanje mogu se lako primeniti na server strani, a klijenti ne moraju nužno biti zahvaćeni. To olakšava upravljanje promenama i održavanje sistema.

**Efikasnost resursa**: Resursi se mogu efikasnije koristiti u klijent-server arhitekturi, gde serveri mogu biti visoko specijalizovani za obradu podataka, dok klijenti mogu biti laganiji i fokusirani na prezentaciju.

**Bolja bezbednost**: Centralizovana kontrola i upravljanje pružaju bolje mogućnosti za sprovođenje bezbednosnih politika. Pristup podacima i funkcionalnostima može se kontrolisati na serveru.

**Podrška za različite platforme**: Klijent-server model omogućava podršku za različite vrste klijenata (npr. desktop aplikacije, veb pregledači, mobilne aplikacije) koji mogu komunicirati sa serverom koristeći odgovarajuće protokole.

Važno je napomenuti da, iako klijent-server arhitektura ima brojne prednosti, takođe postoji nekoliko izazova, kao što su pitanja skalabilnosti servera, zavisnost od pouzdane mrežne komunikacije i potreba za efikasnim upravljanjem resursima na serveru. Prednost klijent-server arhitekture zavisi od specifičnih zahteva i ciljeva sistema koje želite implementirati.

1. **Koliko se Nessus razlikuj od alata nmap?**

Nessus i Nmap su dva različita alata, iako oba imaju ulogu u oblasti sigurnosti i skeniranja mreže. Evo nekoliko ključnih razlika između Nessusa i Nmapa:

**Funkcionalnost**:

**Nessus**: Nessus je alat za skeniranje ranjivosti koji analizira ciljani sistem u potrazi za poznatim sigurnosnim ranjivostima. Pruža informacije o tome gde postoje potencijalne slabosti koje bi mogle biti iskorišćene od strane napadača.

**Nmap**: Nmap je alat za skeniranje mreže koji se koristi za otkrivanje uređaja na mreži, identifikaciju otvorenih portova na tim uređajima, te određivanje vrsta i verzija servisa koji rade na tim portovima. Dok Nmap može pružiti informacije o otvorenim portovima, on se ne fokusira specifično na identifikaciju ranjivosti.

**Fokus**:

**Nessus**: Nessus se fokusira na identifikaciju sigurnosnih ranjivosti na ciljanim sistemima. Koristi bazu podataka sa poznatim ranjivostima kako bi analizirao rezultate skeniranja.

**Nmap**: Nmap se fokusira na identifikaciju otvorenih portova i vrsta servisa na tim portovima. To ga čini korisnim za mapiranje mreže i identifikaciju potencijalnih ciljeva, ali ne pruža detaljne informacije o specifičnim ranjivostima.

**Licenca**:

**Nessus**: Nessus ima komercijalnu verziju koja se plaća, ali takođe ima i besplatnu "Home" verziju koja ima određena ograničenja u pogledu broja IP adresa koje možete skenirati.

**Nmap**: Nmap je besplatan i open-source alat i može se koristiti bez naknade.

**Korišćenje**:

**Nessus**: Nessus se često koristi u organizacijama kako bi redovno skenirao mreže i sisteme u potrazi za potencijalnim bezbednosnim pretnjama.

**Nmap**: Nmap se često koristi za mapiranje mreže, pronalaženje otvorenih portova, te osnovno skeniranje sistema radi identifikacije servisa koji rade na tim sistemima.

U zaključku, Nessus i Nmap imaju različite svrhe i fokusiraju se na različite aspekte bezbednosti. Nessus je usmeren na identifikaciju i analizu ranjivosti, dok je Nmap više orijentisan ka identifikaciji mrežnih uređaja i servisa. Organizacije često koriste ove alate zajedno kao deo sveobuhvatnog pristupa sigurnosti mreže.

1. **Šta je to Nessus „plugin“?**

U kontekstu Nessus-a, "plugin" se odnosi na skup programa ili skripti koje Nessus koristi za identifikaciju i evaluaciju potencijalnih ranjivosti na ciljanim sistemima tokom procesa skeniranja. Ovi plugini su ključni za funkcionalnost Nessus-a jer omogućavaju alatu da prepozna specifične karakteristike, servise, ili slabosti u ciljanim sistemima.

Svaki plugin u Nessus-u je odgovoran za prepoznavanje određene ranjivosti, specifičnog servisa, ili drugih karakteristika sistema. Plugini se redovno ažuriraju kako bi se pratila najnovija otkrića ranjivosti i promene u tehnologijama.

1. **U kojem se formatu mogu napisati rezultati Nessus analize.**

Nessus analiza rezultata može se sačuvati u nekoliko različitih formata, a izbor zavisi od potreba korisnika i namene analize. Evo nekoliko uobičajenih formata za čuvanje rezultata Nessus analize:

**Nessus XML Format (.nessus)**: Ovo je osnovni format koji koristi Nessus za čuvanje rezultata analize. .nessus datoteke su u XML formatu i sadrže detaljne informacije o skeniranju, uključujući identifikaciju ranjivosti, informacije o hostovima, i druge relevantne podatke. Ova datoteka je često korisna za dalju analizu i obradu podataka.

**HTML Format (.html)**: Nessus može generisati izveštaje u HTML formatu koji je prilagođen za pregled u veb pregledaču. Ovo je korisno ako želite deliti rezultate analize sa nekim ko može pregledati izveštaj preko veb pregledača.

**CSV Format (.csv)**: Rezultati analize mogu se izvesti u CSV (Comma-Separated Values) formatu, što omogućava lako uvoz i analizu podataka u alatima za rad sa tabelama, kao što su Microsoft Excel ili Google Sheets.

**Nessus DB (.db)**: Ovo je interni format baze podataka koji se koristi za skladištenje rezultata skeniranja i konfiguraciju Nessus servera.

**PDF Format (.pdf)**: Nessus takođe može generisati izveštaje u PDF formatu koji je pogodan za štampanje ili deljenje u elektronskom formatu.

1. **Zašto se biraju non-DoS skeneri ranjivosti?**

Non-DoS skeneri skeniraju server tako da ne izazivaju prekid rada ili nemogućnost ostalih klijenata da mu pristupe. Zbog toga mogu duže vreme skenirati server, a da pri tome ostanu neprimećeni.