HRVATSKA AKADEMSKA I ISTRAŽIVAČKA MREŽA
CROATIAN ACADEMIC AND RESEARCH NETWORK

PortSentry programski paket

CCERT-PUBDOC-2002-12-09

CARNet CERT u suradnji s LS&S

Sigurnosni problemi u računalnim programima i operativnim sustavima područje je na kojem CARNet CERT kontinuirano radi.

Rezultat toga rada ovaj je dokument koji je nastao suradnjom CARNet CERT-a i LS&S-a, a za koji se nadamo se da će Vam koristiti u poboljšanju sigurnosti Vašeg sustava.

CARNet CERT, www.cert.hr - nacionalno središte za **sigurnost računalnih mreža** i sustava.

LS&S, www.lss.hr - laboratorij za sustave i signale pri Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Sadržaj

1. Uvod

PortSentry jedan je od alata iz TriSentry skupine programskih paketa (logsentry, hostsentry, portSentry). TriSentry (proizvod tvrtke Psionic Technologies) je skupina besplatnih programskih paketa namijenjenih unaprjeđivanju sigurnosti računalnih sustava te općenitoj prevenciji malicioznih aktivnosti.

Portentry programski paket omogućuje detekciju pregledavanja mrežnih TCP i UDP portova (eng. *port scanning*) u stvarnom vremenu te pravovremenu reakciju nakon što su iste uočene. Posebno razvijene metode detekcije PortSentry programa, osim detekcije standardnih postupaka pregledavanja mrežnih portova (*TCP scan*, *connect scan*), omogućuju i detekciju različitih varijanti istog napada kao što su SYN scan, *FIN scan*, *XMAS scan* , *NULL scan* itd. Svi implementirani načini rada, osim spomenute detekcije, omogućuju i blokiranje napada u stvarnom vremenu te prijavljivanje istih administratoru sustava.

U nastavku dokumenta opisani su osnovni postupci instalacije i konfiguracije portsentry programa te metode putem kojih isti obavlja gore navedene zadaće.

2. Način rada

Kompletna problematika načina rada portsentry programskog paketa vezana je uz mogućnost detekcije pregledavanja TCP/UDP portova te reakcije na njih nakon što uočeni. S obzirom na karakteristike programa, isti se može klasificirati kao program za otkrivanje i sprječavanje malicioznih aktivnosti (eng. *attack detection and prevention tool*).

Različite metode udaljenog pregledavanja portova jedan su od prvih koraka neovlaštenih korisnika prilikom provođenja napada. U fazi otkrivanja sustava (eng. *discovery phase*), neovlašteni korisnici pregledavanjem portova otkrivaju dostupne servise na ciljnom računalu. Lista otvorenih portova napadaču omogućuje preciznije usmjeravanje napada, budući da je poznata lista servisa na koje je moguće usmjeriti svoje maliciozne aktivnosti.

Zadatak portsentry programa je upravo taj da uoči neovlašteno pregledavanje mrežnih portova, da pokuša blokirati daljnje maliciozne aktivnosti te da obavijesti administratora o uočenim problemima.

Portsentry program kontinuirano prati stanje TCP i UDP portova na temelju čega može detektirati sljedeće tipove pregledavanja mrežnih portova:

- ***connect scan*** – ova metoda skeniranja portova provodi se kompletnom uspostavom veze između dva računala za svaki od portova koji se žele ispitati (eng. *three way handshaking*). Ukoliko skenirano računalo prihvati konekciju, skenirani port smatra se otvorenim, a u suprotnom se isti smatra zatvorenim.

Ova metoda skeniranja svakako je najpouzdanija, ali ima i jedan veliki nedostatak, koji je čini neprihvatljivom za neovlaštene korisnike. S obzirom da se ovdje radi o kompletnom postupku uspostave veze između dva računala, svaki pokušaj pregledavanja portova ovim putem biti će zabilježen putem log zapisa. Pažljivi administrator lako će primijetiti sumnjive konekcije prema većem broju portova s iste IP adrese, što će mu omogućiti pravovremeno provođenje preventivnih mjera (ukoliko se pokažu potrebnima).

- ***SYN scan*** – metoda poznata pod imenom "*half-scan*", bazira se na djelomičnoj uspostavi veze s ciljnim računalom. Na sve portove koji se žele skenirati šalje se SYN zahtjev za uspostavom veze nakon čega se prati odgovor ciljnog računala. Nakon primljenog odgovora napadač prekida vezu sa skeniranim računalom čime se postupak uspostave veze u tri koraka prekida nakon drugog koraka. Ovisno o primljenom odgovoru moguće je utvrditi stanje pojedinog porta.

Osnovna prednost *SYN scan* metode pregledavanja portova u odnosu na ranije opisanu *connect scan* metodu je ta što nedovršeni postupci uspostave veze neće rezultirati generiranjem log zapisa na stani ciljnog računala (osim ako nije instaliran specijalizirani alat kao što je portsentry). Na ovaj način napadaču je omogućeno neprimjetno skeniranje portova udaljenog računala, što ovu metodu čini prihvatljivijom za neovlaštene korisnike koji žele pritajiti svoje aktivnosti.

- ***FIN scan –*** metoda skeniranja portova bazira se na slanju TCP FIN paketa na svaki od portova za koji se želi ustanoviti njegovo stanje. Ovisno o primljenom odgovoru moguće je utvrditi da

li je port otvoren ili zatvoren. Slično kao i kod *SYN scan* metode i ovdje će pokušaji pregledavanja portova proći nezabilježeno.

- ***NULL scan –*** metoda bazira se na slanju paketa koji nemaju postavljenu niti jednu TCP zastavicu. Ovakvi paketi inače su neregularni, ali mogu poslužiti za ustanovljavanje stanja pojedinih portova. Praćenjem odgovora i ovdje je moguće utvrditi stanje analiziranih portova. Metoda je također neprimjetna.
- ***XMAS scan*** – metoda suprotna *NULL scan* postupku skeniranja portova. Na svaki od portova šalju se paket sa postavljenim FIN, URG i PUSH zastavicama, što su inače također neregularni TCP paketi. Praćenjem odgovora moguće je i ovdje ustanoviti stanje pojedinih portova. Metoda je neprimjetna.
- ***Full XMAS scan*** – Slično kao i upravo opisana *XMAS scan* metoda, samo što su u ovom slučaju postavljene sve TCP zastavice (SYN,ACK,RST, FIN, URG i PUSH).
- ***UDP scan –*** metoda skeniranja provodi se slanjem posebno osmišljenih UDP paketa te praćenjem odgovarajućih odgovora. Usporedbom odaslanih i primljenih paketa moguće je utvrditi stanje skeniranog porta. *UDP scan* metoda nešto je manje pouzdana od ranije navedenih metoda.

Ukoliko `portsentry` detektira pregledavanje mrežnih portova koje ne spada u niti jednu od gore navedenih skupina, biti će prijavljeno nepoznato skeniranje portova. Kod većine programa ovog tipa detekcija nepoznatih metoda skeniranja biti će neprijavljena, što takve programe čini posebno osjetljivima na napade novijeg datuma.

Osnovni preduvjet za uspješnu detekciju skeniranja mrežnih portova vezan je za određivanje pripadnosti pojedinih paketa. Za svaki primljeni paket potrebno je utvrditi da li isti pripada legitimnom prometu, ili je dio malicioznog prometa generiranog u svrhu utvrđivanja stanja mrežnih portova.

Ovo je prilično težak zadatak, pogotovo ako se uzme u obzir da neke od metoda koriste posve legitimne postupke za analizu stanja portova (*conect scan*, *SYN scan*, *UDP scan*). Metode kao što su *FIN*, *XMAS*, *NULL scan* i druge, nešto je lakše detektirati s obzirom da se koriste nelegitimni TCP paketi. S druge strane, takve metode prolaze neprimijećeno, osim ukoliko nisu instalirani specijalizirani alati kao što je `portsentry`.

`Portsentry` programski paket pokušava identificirati maliciozne pakete u odnosu na legitimni promet na dva načina:

- ne analizira se mrežni promet upućen na portove koji su u stanju `LISTEN` (oni portovi na kojima su trenutno aktivni mrežni poslužitelji). Ovo vrijedi ukoliko je program pokrenut u naprednom *stealth* modu (Poglavlje 6.3).
- administrator sustava putem konfiguracijske datoteke definira niz portova, čije se stanje prati u svrhu detekcije malicioznog pregledavanja mrežnih portova

`Portsentry` ima dodatno ugrađenu "inteligenciju" kojom se pokušavaju uzeti u obzir konekcije prema dinamički otvorenim portovima, koji su posljedica načina rada nekih servisa (npr. FTP). Kod FTP servisa (aktivni mod rada), nakon inicijalne uspostave veze, poslužitelj otvara konekciju prema klijentu na dinamički otvoren "visoki" TCP port (> 1024).

Kako bi se u što većoj mjeri smanjio utjecaj ovakvih legitimnih konekcija na pouzdanost rada programa, `portsentry` interno prati stanje ovakvih servisa te dinamički prilagođava listu portova koju treba pratiti. Ukoliko se primijete konekcije prema dinamički otvorenim portovima (za koje je poznato da su dio netom inicirane sesije) iste se neće uzimati u obzir. Nakon što je veza prekinuta, normalno se nastavlja praćenje svih portova koji su definirani konfiguracijom programa.

Ovakav pristup može se usporediti sa *stateful inspection* načinom rada vatrozid sustava, gdje se praćenjem stanja pojedinih konekcija dinamički otvaraju portovi na vatrozidu koji su nužni za funkcioniranje pojedinih servisa.

Na ovaj način umanjuje se broj lažnih upozorenja (eng. *false positives*) koja se često javljaju kao posljedica nemogućnosti praćenja stanja pojedinih mrežnih sesija.

Osim mogućnosti detekcije TCP i UDP *scan* metoda tako što se prati promet upućen prema pojedinim portovima, `portsentry` prati i neke druge parametre koji mogu ukazivati na neovlaštene aktivnosti. `Portsentry` će tako prijaviti ukoliko se detektira IP paket sa neregularnom veličinom zaglavlja ili sumnjivo postavljenim IP opcijama i slično.

3. Implementacija

Kao što je već ranije opisano, `portsentry` prati stanje pojedinih portova u svrhu detekcije potencijalno malicioznih paketa koji upućuju na neovlašteno pregledavanje mrežnih portova. Da bi se omogućila što pouzdanija detekcija napada skeniranjem portova, `portsentry` interno održava listu svih IP adresa sa kojih je inicirana veza prema računalu na kojem je program instaliran.
Stanje iniciranih konekcija bilježi se u obliku polja IP adresa, na temelju kojeg se kasnije u stvarnom vremenu određuje da li je neko računalo već ranije iniciralo sesiju prema istom računalu. Spomenuto polje realizirano je u obliku dvodimenzionalne tablice u kojoj se bilježe ranije spomenute IP adrese sa pridruženim brojačem (eng. *counter*).
Za svaku konekciju koja je inicirana s IP adrese koja već postoji u tablici, brojač se povećava za jedan čime se interno bilježi broj konekcija s iste IP adrese. Nakon što se pređe definirani prag za koji se smatra da je mjerodavan pokazatelj neovlaštenog pregledavanja portova, `portsentry` program reagira te obavještava administratora sustava.
Reakcija `portsentry` programa na primijećeno pregledavanje portova ovisiti će o konfiguraciji programa, o čemu će više riječi biti u nastavku dokumenta.

4. Blokiranje detektiranih napada

Postoji nekoliko načina na koje `portsentry` može reagirati na detektirane napade. To su:

- kreiranje posebnih ruta koje će blokirati pakete sa malicioznih IP adresa;
- mogućnost povezivanja sa vatrozid sustavima baziranim na `ipfwadm`, `ipchains`, `iptables`, `ipfw`, `ipfilter` programskim paketima. U ovom slučaju vatrozid je taj koji blokira pakete sa maliciozne IP adrese, nakon što je obaviješten od strane `portsentry` programa;
- dodavanjem maliciozne adrese u `/etc/host.deny` datoteku.

Svaka od navedenih metoda posjeduje svoje specifičnosti, koje će ukratko biti opisane u nastavku dokumenta.

4.1. Blokiranje usmjeravanjem prometa

U ovom slučaju, nakon što su primijećene maliciozne aktivnosti s odgovarajuće IP adrese, `portsentry` u *routing* tablicu ciljnog sustava ubacuje poseban zapis kojim se odbacuju svi sljedeći paketi sa te IP adrese (eng. *bit-bucket*). Preusmjeravanjem prometa na ovaj način sav promet generiran s maliciozne IP adrese biti će nepovratno izgubljen, a ciljno računalo se za tu IP adresu ponaša kao nepostojeće.
Iako je ova metoda vrlo praktična za blokiranje daljnjih aktivnosti malicioznih korisnika, ista posjeduje jedan prilično veliki nedostatak.
Naime, na ovaj način napadaču se otvara mogućnost provođenja napada uskraćivanjem računalnih resursa (eng. *denial of service*), tako da lažiranjem IP paketa (eng. *spoofing*) blokira promet prema legitimnim računalima. Ukoliko napadač pažljivo osmisli lažirane IP pakete, postoji mogućnost da se ovakvo računalo potpuno "odreže" od ostatka mreže.
Dodatni problem vezan je za gomilanje *route* zapisa, nakon detekcije većeg broja malicioznih aktivnosti. Kreiranjem zasebnih *routa* za svaku pojedinačnu IP adresu može dovesti do povećanog broja zapisa, koji mogu utjecati na performanse sustava. Ovo danas više nije toliko ozbiljan problem, s obzirom na procesorsku snagu današnjih poslužitelja, ali još uvijek može biti utjecajan faktor kod slabijih računalnih arhitektura.

4.2. Povezivanje s vatrozid sustavima

Sljedeća mogućnost blokiranja malicioznog prometa povezana je s nekim Linux vatrozid programima. Trenutno `portsentry` podržava interakciju s sljedećim Linux vatrozid komponentama: `ipfwadm`, `ipchains`, `iptables`, `ipfw`, `ipfilter`. Nakon što se detektira maliciozni promet s određene IP adrese, `portsentry` korištenjem nekog od navedenih programa blokira promet sa tih adresa.

Promet se blokira dodavanjem novog pravila u konfiguraciju vatrozida kojim se regulira daljnji promet sa malicioznih IP adresa.
Ova metoda sadrži isti nedostatak kao i ranije opisano blokiranje dodavanjem IP ruta. Neovlašteni korisnik može pažljivo osmišljenim lažiranjem paketa onemogućiti komunikaciju s legitimnim računalima, što može izazvati ozbiljne probleme.

4.3. Korištenjem `/etc/host.deny` datoteke

Ova metoda bazira se na dodavanju maliciozne IP adrese u `/etc/hosts.deny` datoteku, koju inače koristi TCP Wrapper programski paket. S obzirom na karakteristike `/etc/hosts.deny` datoteke, ova metoda omogućuje samo djelomičnu zaštitu.
Na ovaj način moguće je kontrolirati pristup samo onim servisima koji se štite putem TCP Wrapper programskog paketa. To su tipično oni servisi koji se pokreću putem `inetd` super poslužitelja. Iako ova metoda omogućuje samo djelomičnu zaštitu, ista je najmanje osjetljiva na ranije opisane DoS napade.
Osjetljivost prve dvije konfiguracije (poglavlja 4.1 i 4.2) na napade uskraćivanjem računalnih resursa na prvi pogled čini se prilično ozbiljnom prijetnjom.
No, provedena testiranja ipak su pokazala da se neovlašteni korisnici rijetko kada koriste *spoofing* metodama prilikom pregledavanja portova udaljenog računala. U gotovo 99,99% slučajeva pregledavanje portova provodi se standardnim metodama i alatima bez korištenja lažiranja adresa. Upravo se iz tog razloga korisnicima preporučuju automatske metode blokiranja adresa, budući da iste pokazuju najbolja svojstva sa stanovišta sigurnosti.

5. Instalacija programa

Instalacija `portsentry` programskog paketa prilično je jednostavan postupak. Program je dostupan u `tar.gz` arhivi, a moguće ga je dobaviti sa sljedeće URL adrese: http://www.psionic.com.
Prije samog prevođenja programa, potrebno je nekoliko varijabli unutar `portsentry_config.h` datoteke prilagoditi sustavu. To su:

Ime varijable	Značenje	Inicijalna vrijednost
`CONFIG_FILE`	Lokacija konfiguracijske datoteke `portsentry` programa.	`/usr/local/psionic/portsentry2/portsentry.conf`
`WRAPERR_HOSTS_DENY`	Lokacija TCP Wrapperr `hosts.deny` datoteke.	`/etc/hosts.deny`
`SYSLOG_FACILITY`	*Facility* parametar pod kojim `portsentry` šalje log zapise `syslogd` poslužitelju.	`LOG_DAEMON`
`SYSLOG_LEVEL`	Prioritet pod kojim se šalju log zapisi `syslogd` poslužitelju.	`LOG_NOTICE`

Tablica 1- Varijable portsentry_config.h datoteke

U većini slučajeva potrebno je modificirati `CONFIG_FILE` varijablu, a preporučuje se i modifikacija `SYSLOG_FACILITY` varijable kojoj je potrebno pridijeliti jednu od `LOCAL` vrijednosti (`LOCAL0` – `LOCAL7`), čime se olakšava izdvajanje `portsentry` log zapisa od ostalih logova sustava. Ostale varijable nije potrebno mijenjati.
Nakon što je obavljeno podešavanje gore navedenih varijabli, moguće je krenuti s prevođenjem programa. Prevođenje programa obavlja se zadavanjem `make` naredbe, kojoj je kao argument potrebno proslijediti ime operacijskog sustava za koji se program provodi.
`Make` naredbi trenutno je moguće proslijediti sljedeće vrijednosti: `Linux`, `BSD`, `OpenBSD`, `FreeBSD`, `NetBSD` te `generic`. To znači da se na Linux operacijskom sustavu prevođenje provodi zadavanjem sljedeće naredbe:

```
# make linux
```

Nakon toga potrebno je instalirati program zadavanjem naredbe:

```
# make install
```

Ovi postupci rezultirati će instalacijom programa u `/usr/local/psionic` direktorij. Mjesto instalacije moguće je modificirati promjenom `INSTALLDIR` varijable unutar `Makefile` datoteke. Ukoliko je instalacija obavljena bez greške, može se nastaviti s konfiguracijom programa.

6. Konfiguracija programa

Za konfiguraciju `portsentry` programa zadužene su dvije datoteke. To su:

- portsentry`.conf` i
- portsentry`.ignore`

Datoteka `portsentry.ignore` jednostavno sadrži listu IP adresa koje `portsentry` program ignorira u postupku detekcije skeniranja portova. Sve konekcije koje dolaze s adresa navedenih u `postsentry.ignore` datoteci tretirati će se kao "sigurne" konekcije te se kao takve neće uzimati u obzir. Namjena ove datoteke je sprječavanje eventualnog blokiranja adresa onih računala s kojim se intenzivno komunicira. Unutar ove datoteke trebala bi se minimalno nalaziti `localhost` adresa 127.0.0.1, zatim IP adrese pridjeljene ostalim lokalnim sučeljima te IP adrese lokalnih računala sa kojima se svakodnevno legitimno komunicira.

Slijedi uređivanje `portsentry.conf` datoteke, unutar koje je moguće podesiti sve ostale parametre programa.

Najvažniji segment uređivanja `portsentry.conf` datoteke vezan je za listu TCP i UDP portova koje će program nadzirati, budući da će definiranje iste najviše utjecati na način rada programa. Osim spomenute liste portova ovdje se definiraju i lokacije nekih ostalih datoteka koje program koristi za svoj rad (`portsentry.ignore`, `portsentry.history`, `portsentry.blocked`).

Treba napomenuti da inicijalna konfiguracija `portsentry` programa ne omogućuje prevođenje IP adresa u FQDN imena računala. Za omogućavanje iste potrebno je varijabli `RESOLVE_HOST` pridijeliti vrijednost 1. Osim ukoliko nije nužno neophodno ne preporučuje se korištenje ove mogućnosti, budući da ista povlači učestalo iniciranje velikog broja DNS upita, što se može negativno odraziti na performanse sustava.

`Portsentry` programski paket koristi tri osnovne metode detekcije neovlaštenog pregledavanja portova. To su:

- klasična metoda (eng. *classical scan detection*)
- *stealth* metoda
- napredna *stealth* metoda (eng. *advanced stealth scan detection*)

Za svaku od podržanih metoda potrebno je podesiti odgovarajuće varijable `portsentry.conf` datoteke, kojima će se definirati način rada za podržane metode.

6.1. Klasična metoda detekcije

Konfiguracija klasične metode detekcije skeniranja portova provodi se uređivanjem sljedećih varijabli unutar `portsentry.conf` konfiguracijske datoteke:

- `TCP_PORTS` – lista TCP portova koja će se pratiti u svrhu detekcije neovlaštenog pregledavanja portova
- `UDP_PORTS` – lista UDP portova koja će se pratiti u svrhu detekcije neovlaštenog pregledavanja portova

Spomenute varijable kao vrijednosti prihvaćaju niz zarezom odvojenih portova koje će program "slušati" (eng. *listen*) nakon pokretanja. Za razliku od ostalih načina detekcije, u ovom se načinu rada `portsentry` program veže (eng. *bind*) na sve zadane portove, gdje se postavlja u `LISTEN` stanje.

Ponašanje programa u ovom slučaju može se usporediti s klasičnim mrežnim poslužiteljima, gdje svaki poslužitelj nakon pokretanja "zauzima" određeni mrežni port. Nakon što je port jednom "zauzet", niti jedan drugi program se više ne može vezati na isti port (dok se isti ne oslobodi).

U nastavku je dan ispis `netstat` komande izvršene nakon što je program pokrenut u klasičnom modu rada (`aware` konfiguracija - Tablica 1).

```
# netstat -anp
tcp  0.0.0.0:54320   0.0.0.0:*          LISTEN      6156/portsentry
tcp  0.0.0.0:49724   0.0.0.0:*          LISTEN      6156/portsentry
tcp  0.0.0.0:40421   0.0.0.0:*          LISTEN      6156/portsentry
```

```
tcp  0.0.0.0:32774   0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:32773   0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:32772   0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:32771   0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:31337   0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:27665   0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:20034   0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:12346   0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:12345   0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:6667    0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:5742    0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:2000    0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:1524    0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:1080    0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:635     0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:540     0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:143     0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:119     0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:15      0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:11      0.0.0.0:*           LISTEN      6156/portsentry
tcp  0.0.0.0:1       0.0.0.0:*           LISTEN      6156/portsentry
tcp  161.53.64.27:22 161.53.64.180:1390  ESTABLISHED 6145/sshd
tcp  0.0.0.0:22      0.0.0.0:*           LISTEN      12368/sshd
tcp  0.0.0.0:6000    0.0.0.0:*           LISTEN      238/X
tcp  0.0.0.0:21      0.0.0.0:*           LISTEN      222/proftpd
tcp  0.0.0.0:79      0.0.0.0:*           LISTEN      201/inetd
tcp  0.0.0.0:776     0.0.0.0:*           LISTEN      173/rpc.statd
tcp  0.0.0.0:111     0.0.0.0:*           LISTEN      86/portmap
udp  0.0.0.0:54321   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:32774   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:32773   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:32772   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:32771   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:32770   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:31335   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:34555   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:37444   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:31337   0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:2049    0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:700     0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:641     0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:640     0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:635     0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:513     0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:162     0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:161     0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:69      0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:9       0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:7       0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:1       0.0.0.0:*                       6188/portsentry
udp  0.0.0.0:514     0.0.0.0:*                       12549/syslogd
udp  0.0.0.0:177     0.0.0.0:*                       235/gdm
udp  0.0.0.0:517     0.0.0.0:*                       201/inetd
udp  0.0.0.0:774     0.0.0.0:*                       173/rpc.statd
udp  0.0.0.0:111     0.0.0.0:*                       86/portmap
```

```
raw  0.0.0.0:1       0.0.0.0:*                   -
raw  0.0.0.0:6       0.0.0.0:*                   -
```

Iz danog ispisa može se jasno vidjeti broj portova na kojima je pokrenut `portsentry` program.
Inicijalni sadržaj `portsentry.conf` datoteke nakon instalacije programa, korisniku ostavlja tri liste portova koje se mogu odabrati kao predložak za rad u klasičnom načinu rada.
Liste se međusobno razlikuju prema broju portova koji se nadziru, čime se može regulirati sigurnosni nivo konfiguracije programa. Ovisno o potrebama sustava na kojemu je program instaliran moguće je odabrati jednu od triju konfiguracija navedenih u sljedećoj tablici.

Ime konfiguracije	Broj portova (TCP/UDP)	Komentar
`comprehensive`	54/32	"Najstroža" konfiguracija koja uključuje sve važnije portove u području od `tcpmux` TCP/1 porta do TCP/54321 porta kojeg koriste neke trojan aplikacije kao što su `SchoolBus` ili `BackOrifice`.
`aware`	30/18	Nešto blaža konfiguracija koja pokriva slično područje adresa kao `comprehensive` konfiguracija ali s manjim brojem uključenih portova.
`bare-bones`	24/14	Najblaža konfiguracija koja nadzire najmanji broj TCP i UDP portova.

Tablica 2 - Inicijalne konfiguracije klasičnog načina rada portsentry programa

Ovisno o namjeni računala na kojem se program koristi, potrebno je odabrati jednu od raspoloživih konfiguracija te je eventualno dodatno prilagoditi specifičnim potrebama sustava.
Vrijednost `TCP_PORTS` i `UDP_PORTS` varijabli nevažeća je ukoliko se program pokrene u naprednom modu (poglavlje 6.3)

6.2. *Stealth* metoda detekcije

`Portsentry` program pokrenut u *stealth* modu pratiti će istu listu portova definiranu `TCP_PORTS` i `UDP_PORTS` varijablama, samo što se u ovom slučaju program ne veže na niti jedan mrežni port. Pokretanje `netstat` naredbe nakon što je program pokrenut u *stealth* modu potvrđuje ovu tvrdnju:

```
# netstat -anp
 Proto   Local adress      Foreign adress        State         PID/Program Name
 tcp     161.53.64.27:22   161.53.64.180:1390    ESTABLISHED   6145/sshd
 tcp     0.0.0.0:22        0.0.0.0:*             LISTEN        12368/sshd
 tcp     0.0.0.0:6000      0.0.0.0:*             LISTEN        238/X
 tcp     0.0.0.0:21        0.0.0.0:*             LISTEN        222/proftpd
 tcp     0.0.0.0:79        0.0.0.0:*             LISTEN        201/inetd
 tcp     0.0.0.0:776       0.0.0.0:*             LISTEN        173/rpc.statd
 tcp     0.0.0.0:111       0.0.0.0:*             LISTEN        86/portmap
 udp     0.0.0.0:514       0.0.0.0:*
 udp     0.0.0.0:177       0.0.0.0:*
 udp     0.0.0.0:517       0.0.0.0:*
 udp     0.0.0.0:774       0.0.0.0:*
 udp     0.0.0.0:111       0.0.0.0:*
 raw     0.0.0.0:1         0.0.0.0:*
 raw     0.0.0.0:6         0.0.0.0:*
```

Može se primijetiti kako u ovom slučaju `portsentry` program ne "sluša" niti na jednom od portova, već na nivou operacijskog sustava analizira sve pakete koji stignu na jedan od kontroliranih portova.

6.3. Napredna *stealth* metoda detekcije

Napredni *stealth* način rada (eng. *advanced stealth mode*) vrlo je sličan upravo opisanoj *stealth* metodi (6.2). Osnovne razlike između ove dvije metode vezane su za način na koji program određuje listu portova koju će pratiti uz set naprednih mogućnosti. Pritom se prvenstveno misli na ranije opisano dinamičko praćenje mrežnih konekcija te automatsko prilagođavanje programa na temelju uočenih promjena (Poglavlje 2).

Nakon što se `portsentry` program pokrene u naprednom *stealth* modu, isti će pratiti sve portove u području definiranom `ADVANCED_PORTS_TCP` i `ADVANCED_PORTS_UDP` varijablama, osim onih koji su trenutno u `LISTEN` stanju. To znači da program iz zadane liste portova isključuje one na kojima su trenutno pokrenuti neki od mrežnih poslužitelja, čime se iz analize želi isključiti legitimni mrežni promet.

Za konfiguraciju naprednog *stealth* moda rada bitne su sljedeće varijable:

- `ADVANCED_PORTS_TCP` – Gornja granica područja TCP portova koji se žele nadzirati. Inicijalna vrijednost ove varijable je 1024, što znači da će se kontrolirati svi TCP portovi u području od 1-1024, osim onih na kojima su trenutno pokrenuti neki od mrežnih poslužitelja.
- `ADVANCED_PORTS_UDP` – isto kao i ranije opisana varijabla, samo za UDP portove.
- `ADVANCED_EXCLUDE_TCP` – lista portova koji će se isključiti iz analize pored onih koji su već "zauzeti". Ovdje se tipično navodi NETBIOS TCP/139 port, a moguće je definirati i druge ukoliko se ukaže potreba.
- `ADVANCED_PORTS_UDP` – isto, samo za UDP portove. Primjer ovdje navedenih portova su 137 i 138 (NETBIOS), 520 (RIP), 67 (bootp) i sl.

U ovome modu rada `TCP_PORTS` i `UDP_PORTS` varijable su nevažeće.

6.4. Blokiranje napada

U ovom poglavlju opisana je konfiguracija programa s obzirom na mogućnosti reakcije na uočene maliciozne aktivnosti. Podržane metode već su ranije opisane (poglavlje 3), a ovdje će biti ukratko opisani konfiguracijski parametri.

Postoje dvije osnovne varijable kojima se opisuje kako će program reagirati na uočene napade. To su `BLOCK_TCP i BLOCK_UDP` varijable koje mogu poprimiti tri vrijednosti 0, 1 i 2, čime se definira ponašanje programa. U sljedećoj tablici dano je značenje svake od njih.

Vrijednost varijable	Značenje
0	Generira se log zapis, ali se ne blokira daljnje provođenje napada. Poželjna opcija ukoliko se samo žele nadzirati neovlaštena pregledavanja portova, bez posebne reakcije nakon što su ista uočena.
1	Generira se log zapis, ali se dodatno na jedan od podržanih načina (poglavlje 3) blokira daljnje provođenje napada.
2	Generira se log zapis, nakon čega se pokreće naredba ili skripta od strane administratora. Korisna opcija ukoliko se žele posebno definirati postupci nakon što se uoče maliciozne aktivnosti.

Tablica 3: Podešenja varijabli BLOCK_TCP i BLOCK_UDP

Ukoliko se varijablama `BLOCK_TCP i BLOCK_UDP` pridjeli vrijednost 1, potrebno je definirati način blokiranja daljnjeg provođenja napada. Kao što je već ranije rečeno, podržane su tri osnovne metode:

- blokiranjem rute prema napadaču
- povezivanje s vatrozidom
- korištenjem `/etc/hosts.deny` datoteke

Prednosti i nedostatci pojedinih metoda već su ranije opisani (poglavlje 3) tako da će ovdje biti opisan samo način podešavanja istih.

Željena metoda odabire se definiranjem `KILL_ROUTE` varijable. Ovisno o vrijednosti ove varijable, poduzimati će se različite akcije nakon detekcije malicioznih aktivnosti. Primjer definiranja `KILL_ROUTE` varijable za različite metode detekcije:

1. Blokiranje rute

   ```
   KILL_ROUTE="/sbin/route add -host $TARGET$ gw 333.444.555.666"
   ```

   Napomena: Definiranje nepostojeće rute nešto je drugačije za različite operacijske sustave (Solaris, HP-UX, ...) tako da će ova naredba varirati od sustava do sustava. Gore navedena naredba vrijedi za Linux operacijske sustave.
2. Povezivanje s vatrozidom

```
KILL_ROUTE="/usr/local/bin/iptables -I INPUT -s $TARGET$ -j
DROP"
```

Osim `iptables` naredbe, moguće je korištenje sljedećih naredbi: `ipfwadm`, `ipchains`, `ipfw` i `ipfilter`.

`KILL_HOST_DENY` dodatno omogućuje ubacivanje IP adrese u `/etc/hosts.deny` datoteku nakon što je sa iste uočen pokušaj pregledavanja portova. Preporučuje se korištenje ove mogućnosti u kombinaciji s jednom od dvije gore navedene opcije. budući da ista pruža dodatni nivo zaštite u slučaju pada sustava. Naime, nakon ponovnog pokretanja sustava (eng. *reboot*), zadane `ipchains` i `route` naredbe postaju nevažeće, čime se uklanja ranije postavljena zaštita. Nasuprot tome sadržaj `/etc/hosts.deny` ostati će nepromijenjen čime će se omogućiti bar određeni nivo zaštite.

Slijedeća varijabla koju je potrebno spomenuti, bitna je ukoliko se varijablama `BLOCK_TCP` i `BLOCK_UDP` pridijeli vrijednost 3. U tom slučaju nakon detekcije napada `portsentry` će pokrenuti naredbu ili skriptu ljuske koju je definirao administrator sustava. Na ovaj način administratoru se omogućuje provođenje posebno definiranih akcija u slučaju detekcije napada (npr. `reverse finger` servis ili nešto slično).

Potrebno je na kraju spomenuti još jednu varijablu, kojom se može utjecati na način detekcije pregledavanja portova. Radi se o varijabli `SCAN_TRIGGER` kojom se definira maksimalni broj konekcija prema kontroliranim portovima, prije nego što se detektirane aktivnosti proglase potencijalno malicioznima. Inicijalna vrijednost ove varijable je 1, pri čemu će `portsentry` svako spajanje na kontrolirani port prijaviti kao pokušaj pregledavanja portova.

U svrhu smanjivanja lažnih upozorenja preporučuje se prilagođavanje ove varijable, ovisno o namjeni računala na kojem je program instaliran.

7. Pokretanje programa

Nakon što je program ispravno konfiguriran, potrebno ga je pokrenuti kako bi mogao obavljati zadaće za koje je namijenjen. Postoji šest različitih načina u kojima je moguće pokrenuti `portsentry` program. To su:

- `portsentry -tcp` – klasični (eng. *basic*) način rada za TCP protokol.
- `portsentry -udp` – klasični (eng. *basic*) način rada za UDP protokol.
- `portsentry -stcp` – *stealth* (eng. *stealth*) način rada za TCP protokol.
- `portsentry -sudp` – *stealth* (eng. *stealth*) način rada za UDP protokol.
- `portsentry -atcp` – napredni *stealth* (eng. *advanced stealth*) način rada za TCP protokol.
- `portsentry -audp` – napredni *stealth* (eng. *advanced stealth*) način rada za UDP protokol.

Važno je spomenuti da za svaki protokol (TCP i UDP) program može biti pokrenut u samo jednom modu.

8. Testiranje programa

`Portsentry` program testiran je u svim podržanim načinima rada (`-tcp`, `-udp`, `-stcp`, `-sudp`, `-atcp`, `-audp`). Simulacija napada pregledavanjem mrežnih portova provedena je `nmap` programskim paketom, trenutno najpopularnijim automatiziranim alatom za pregledavanje mrežnih portova. Za svaki od načina detekcije, pokrenute su sve metode pregledavanja portova trenutno podržane od strane `nmap` programskog paketa. Rezultati testiranja priloženi su u sljedećoj tablici:

metoda pregledavanja portova	klasični način rada (-tcp, -udp)	Stealth način rada (-stcp, -sudp)	napredni stealth način rada (-atcp, -audp)
connect scan	DA	DA	DA
SYN scan	NE	DA	DA
FIN scan	NE	DA	DA
Null scan	NE	DA	DA
XMAS scan	NE	DA	DA

metoda pregledavanja portova	klasični način rada (-tcp, -udp)	Stealth način rada (-stcp, -sudp)	napredni stealth način rada (-atcp, -audp)
ACK scan	DA*	DA	DA*
UDP scan	DA	DA*	DA
window scan	DA*	DA*	DA*
IP protocol scan	NE	NE	NE

Tablica 4 - Rezultati testiranja programa

Napomena: Točke testiranja označene zvjezdicom (*) govore da je program uspješno detektirao pregledavanje portova, ali da ga je pogrešno protumačio. Tu se tipično radi o ACK i *window scan* metodama pregledavanja portova koje je program protumačio kao XMAS scan metodu.

9. Zaključak

Provedena testiranja pokazala su da `portsentry` program predstavlja izvrsno rješenje u pogledu detekcije neovlaštenog pregledavanja mrežnih TCP i UDP portova. Budući da je pregledavanje mrežnih portova jedan od prvih koraka neovlaštenih korisnika, `portsentry` program administratorima pomaže u pravovremenom otkrivanju i prevenciji malicioznih aktivnosti. Rezultati testiranja programa dani su u prethodnom poglavlju (Poglavlje 8).

Program je uspješno i u vrlo kratkom vremenu reagirao na većinu metoda pregledavanja mrežnih portova. Metode blokiranja prometa nakon detektiranog napada, također su se pokazale vrlo efikasnima, zajedno sa mogućnošću izvršavanja proizvoljno definiranih naredbi, odnosno skripti ljuske.