Høstprosjekt

Svein Ove Undal

1.0 Innholdsfortegnelse

2.0 Innledning

Bakrunnen til oppgava er simpelt den enkle interessa og erfaringa rundt å sette opp og utvikle eit gateway/kaptiveportal system, som har god oversikt over alle brukerane. Det finst mangen former for gateway/brannmur/kaptiveportals, men det finst ikkje system som gjere det på den måten som er gjort i det prosjektet her.

2.1 Oppgåvebeskrivelse

Oppgåva er ein del av eit større bachelor prosjekt, som forprosjekt vil fokuset ligge på å lage eit simpelt gateway/kaptiveportal system, systemet skall ha kontroll på kvar enkelt brukar. IP, mac-addresse og brukernavn blir knytta opp mot kvarandre. Det vil og vere kontroll av duplikater, eller forandringer av ip / mac-addresser. som forhindrer mangen sikkerhets problem som er kjent med liknande system.

Innlogginga, som på alle andre kaptiveportal system skall gå via web loggin. Der før ein bruker blir autentisert vil alle forespørlar på port 443 eller port 80 bli routa til innloggings sida. Når du logger inn, skall det bli oppretta ein brannmur regel til å tillate den klienten på nett.

Begrenser systemet til ipv4, vil og kunn vere mulig å administrere systemet via terminal.

3.0 Teknologi, Arkitektur

Alt er gjort under, linux debian 6. Av utvikling er det for det meste benytta bash scripting, og litt php. Systemet vil forøvrig fungere på alle nye linux distrubusjoner bassert på debian, gjitt att du har rette pakker.

Det var ikkje eit vanskelig valg, det er ingen operativsystem som er like gode på nettverksadministrasjon som linux. Bash scripting er ikkje nødvendig vis det beste, men til den bruken her av mangen små script fungerer det utmerket. PHP er og i små skala eit svært godt verktøy til kjøring av script på server-sia aktivert over web.

3.1 Linux pakker

iptables: er det som kontrollerar all nettverkstraffikk inn og ut. Iptables er eit sett med regler du definerer sjølv, du bestemmer kriteriene på kva typer nettverks pakker den skall sjå etter. så

bestemmer du kva den skall gjere med den pakken. forkaste, acceptere eller endre destinasjon.

apache2: apache2 er i dag den mest brukte og ikkje minst dokumenterte opensource webserveren, i prosjektet er det den serveren som hoster login-sida.

isc-dhcp-server: dhcp serveren som deler ut ip, dns og gateway info til alle klienter. Her er lease tabellen (oversikt over iper som er delt ut) tatt i bruk til å bestemme att dei som faktisk logger på har ein ip ifra dhcp serveren. Vist ikkje, har dei ein statisk ip-addresse som medører att dei blir nekta nett til dei har aktivert dhcp.

ldap-utils: Ein pakke som gjir blant anna ldapsearch, som blir brukt til å autentisere brukere oppimott oppgjitt open-ldap server.

4.0 Systemets virkemåte

Systemet fungerer som dhcp-server, gateway, kaptive portal og brannmur. Når du kopler deg til nettverket, fysisk eller på eit trådløst nett. får du utdelt nettverks informasjon via dhcp-server, isc-dhcp-server. Den gjir deg ein eigen ip, ipen til gateway og dns server.

Brannmuren er konfigurert sånn att den dropper all traffik utenom, dns, http og https (port 53, 80, 443) all traffikk som går på http og https blir redirekta til systemet sin webserver, som er loginsida index.php (10.0.0.1:80). Som betyr att du alltid, uansett kva addresse du skriver i ein nettleser vill du få opp login sida.

Under innloggінga blir det tatt eindel tester. Du kan ikkje ha statisk ip-addresse, dhcp-serveren lager ei oversikt over alle leases som er delt ut, er din klient ikkje i lista vil du bli nekta å logge inn. Den vil og teste om din ip, mac og brukernavn ligger i systemet, isåfall blir du nekta å logge inn. Etter det blir brukernavnet og passordet sjekka, stemmer det vil det bli oppretta ein brannmuregel som tillater din pc sin ip til å kopple til nettet. Det blir og oppretta ei fil med ditt brukernavn i clients mappa, som innehalder din ip, og mac-addresse.

Det er og lagt inn eit liten ekstra ting som gjere det lettare å feilsøke. Det er 3 forskjellige bilder som blir lasta når du skall lese login sida. Vist du ikkje er innlogga vil du få opp ein ipfire-tux, er du innlogga skall du få opp ein world-tux, er du banna får du opp ein shocked-tux. Hensikten er å raskt finne ut av om det er noko gale med clienten, eller serveren.

Når klienten er logga inn er det ein test som blir kjørt på alle klientane, er det ikkje mulig å pinge klienten blir den tvungen til å logge seg inn igjen. Den tester og om det finst duplikater av mac, klienten vil då og få fjærna dens nett tilgang, som tvinger klienten til å logge seg på igjen. Vist alt

går bra vil det bli loggført antall tilkoplinger på nettet klienten har, og kor masse data klienten har lasta opp og ned.

4.1 Scripts

4.1.1 brannmur.sh

Scriptet setter nødvendige brannmuregler, det fjærner alle tidligare regler, tillater å forwarde ipv4. den setter og oppgjit ip til eth1, og restarter dhcp serveren.
brannmur regler:

- tillater 1 til mangen NAT
- redirekter all traffikk på port 80 og 443 til oppgjit ip
- tillater forwarding av traffik på port 443 og port 80
- tillater forwarding av trafikk på port 53
- dropper all anna forwarding traffik

Scriptet må bli kjørt etter restart av server.

4.1.2 accept.sh

Er scriptet som blir kjørt kvar gong nokon logger seg på, scriptet tar 3 variabler. ip, bruker og passord. Scriptet sjekker, om du har ein dhcp-lease, om det er andre klienter med det brukernavnet eller samme ip-addresse. så om du har rett bruker / passord. Om alt stemmer lager scriptet ein brannmur regel for din ip, og kjører scriptet lease.sh. klienten skall no ha full tilgang til internett.

4.1.3 drop.sh

Script som blir kjørt får å slette tilgangen av den gjitte bruker. drop.sh tar 1 eller 2 variabler. ip og ein eventuell grunn. først finner scriptet brukernavnet til tilhøyrandes ip, så fjærner scriptet rekrusivt alle regler i iptables som innehelder ip-addressa. Om scriptet får 2 variabler, vil den i tilleg til å droppe tilgangen til brukaren, og legge ei fil i error mappa med teksten i den andre variabelen.

4.1.4 ping.sh

Script som er meint å kjøre ein gang i blant, som for eksempel kvart 2 minnut. Scriptet sjekker om alle klienter går å pinge, eller om det finst duplikater av mac-addresser. vist det er tilfelle vil klienten bli tvungen til å logge seg inn på nytt igjen. Scriptet driver og logg føring. skriver til ei fil antall connections og kor masse data klienten har sendt og motatt.

4.1.5 findUser.sh

simpelt script som returnerer brukernavnet til den ipen, ved å gå igjøna alle filer i clients til den finner ein ip som stemmer overens med klienten som kopler til.

4.1.6 lease.sh

Scriptet tar to variabler, brukernavn og ip. søker igjøna lease fila for å finne mac-addressa til gjitt ip. scriptet lagrer så ei fil med ip og mac-addrese i client mappa. filnavnet blir brukernavnet.

4.1.7 Index.php

php fila som kontrollerer alt. Ved hjelp av ip-addressa til klienten sjekker scriptet om du allerede har nett, om du er banna. eller om du ikkje endå er logga inn. Det blir demonstrert med 3 forkjellige bilder som blir lasta. Poenget her er å gjere det lettare å feilsøke eventuelle problemer med klienten.

Når knappen på sida blir aktivert vil den prøve å autentisere deg, med å kjøre accept.sh. om alt går bra får du nett. Du vil sjå att bildet på sida forandrer seg til ein “world-tux” og du blir snart redirekta til tihlde.org si side. og har no muligheten til å bruke nettet fritt.

om du allerede var logga inn og trykker på knappen vil den logge deg ut, du ser då att bilde på sida forandrer seg til ein ipfire-tux.

5.0 Konkusjon

Prosjektet har våre ei forprosjekt til ein større bachelor oppgave, hensikten var å sjå om det er mulig å lage eit sånt system på ein effektiv måte. Som det heilt klart er, systemet fungerer svært bra i praksis og på grunn av att det er modulbassert er det lett å vidare legge til funksjonalitet.

Det er selfølgelig litt funksjonalitet eg gjerne skulle ha fått til, der tida ikkje strekte til. Manglar som ein måte å forhindre dns-tunnelering, og automatiske brannmuregler som fordeler båndbredden på brukerane om det blir mangel på kapasitet.

Eg er nøgd med sluttresultatet, det fungerer smertefritt og har stort potensiale i vidare utvikling.

6.0 Installasjon

Ein nyare versjon av debian/ubuntu er nødvendig for å få det her til å fungere, det er og

nødvendig å ha følgandes pakker installert.
ldap-utils, apache2, sudo, dhcp3-server, git

```
sudo apt-get install ldap-utils apache2 dhcp3-server git
```

6.1 dhcp3-server:

endre fila /etc/default/isc-dhcp-server så nederstelinja ser sånn ut

```
INTERFACES=”eth1”
```

(viktig att du har 2 nettverkskort, og att eth0 er til koppla nett. og eth1 er nettverket som du skall gji nett.)

endre fila /etc/dhcp3/dhcpd.conf til følgandes

```
ddns-update-style none;

option domain-name "dittdomene.org";
option domain-name-servers 8.8.8.8, 8.8.4.4;

default-lease-time 600;
max-lease-time 7200;
authoritative;

log-facility local7;


subnet 10.0.0.0 netmask 255.255.255.0 {
range 10.0.0.10 10.0.0.20;
option broadcast-address 10.0.0.255;
option subnet-mask 255.255.255.0;
option routers 10.0.0.1;
}
```

legge til ip til eth1

```
sudo ifconfig eth1 10.0.0.1 netmask 255.255.255.0
```

Restarte dhcp serveren

```
sudo /etc/init.d/isc-dhcp-server restart
```

6.2 Apache2, sudoers

apache-2 trenger i utgangspunktet ingen endring. Men du er nødt til å tillate www-data til å bruke sudo på enkelte script.

Så legg til følgandes nederst i fila /etc/sudoers.d

(Anbefaler å aktivere ssl modulen på apache2, så det er mulig å ha innlogginga via https. Det innebærer å produsere sjølvsignerte sertifikat, for å halde installasjonen simpelt blir det ikkje tatt med her)

```
www-data ALL = (ALL) NOPASSWD: /var/www/accept.sh, /var/www/drop.sh,
/var/www/index.php, /var/www/tv.php, /var/www/findUser.sh
```

6.3 scripta

clone eit prosjektet med git, og legge alt i rett mappe

```
cd /var/www
sudo git clone https://github.com/sveinou/project.git
cd project && sudo mv * /var/www && cd .. && sudo rm -r project
```

6.4 Crontab

legge til ping script som skall kjørast kvart 2 min.

```
sudo crontab -e
```

legg til nederst i fila.

```
*/2 * * * * /var/www/ping.sh
```

6.5 Brannmur / testing

Kjør brannmur fila.

```
sudo ./brannmur.sh
```

Då skall alt fungere! får å teste det trengs det ein switch tilkopla eth1 til server maskina, klienter på den vil då få nett via systemet.

Bruker forøvrig tihlde sin ldap-server, så brukeren testsau med passord 0188 skall fungere.