Sleutelbeheer van IPSec

- Doel:
  - opzetten van SA's
  - bepaling, verdeling en refreshing van sleutels
- kan manueel gebeuren
  - gedaan door sysadmin
  - alleen mogelijk voor kleine en statische omgevingen
- gebeurt beter automatisch
- Protocollen (*what's in a name?*):
  - IKE
  - ISAKMP/Oakley
  - SKEME
  - …

Sleutelbeheer van IPSec (1)

- ISAKMP (Internet Security Association and Key Management Protocol, RFC 2408)
  - raamwerk voor beheer van SA's en sleutels
    - **onderhandelen, wijzigen, vernietigen**
  - bepaalt formaat payload voor uitwisseling van gegevens nodig voor generatie van sleutels en authenticatie
  - bepaalt *niet* sleuteluitwisselingsprotocol
  - gebruikt elementen van Oakley en SKEME
- Oakley (RFC 2412)
  - sleuteluitwisselingsprotocol gebaseerd op Diffie-Hellman
  - generisch (geen opgelegde formaten of informatie-uitwisseling)
  - voegt authenticatie toe van de 2 partijen
  - laat verschillende manieren toe ➔ ≠ modes in IKE/ISAKMP

Sleutelbeheer van IPSec (2)

- SKEME, A Versatile Secure Key Exchange Mechanism for Internet
  - artikel van Hugo Krawczyky (IEEE, november 1995)
  - levert o.a. methode van authenticatie:
    - elke peer encrypteert random getal met publieke sleutel
    - plaintext waarden bepalen mee sleutel
- IKE (Internet Key Exchange: RFC 2409)
  - eigenlijke implementatie voor beheer van SA's en sleutels
  - gebruikt ISAKMP, Oakley en SKEME
  - generisch protocol, ook gebruikt in routing protocollen (RIP, OSPF)
- DOI (Domain of Intepretation, RFC 2407)
  - bepaalt hoe IKE moet gebruikt worden om SA's op te zetten voor IPSEC

Sleutelbeheer van IPSec (3)

- Implementaties
  - Photuris (OpenBSD IPSec)
  - SKIP
  - Pluto (FreeS/WAN - OpenS/WAN)
- IKE werkt over UDP, poort 500

IKE overzicht (1)

Eisen opgelegd door ISAKMP:

- alle uitgewisselde info geheim en geauthentiseerd
- uitsluitend uitwisseling tussen **geauthentiseerde partijen**
- weerstaat aanvallen
  - denial-of-service
    - unieke cookies in boodschappen
    - laten toe snel boodschappen te weigeren
  - man-in-the-middle = boodschappen
    - achterhouden
    - wijzigen
    - terugsturen
    - replay
    - omleiden naar andere ontvangers

IKE overzicht (2)

- voorziet in PFS (perfect forward secrecy)
  dwz. gekraakte sleutel
  - geeft geen bijkomende info over vroegere of nieuwe sleutels
  - kan alleen gebruikt worden bij gegevens geëncrypteerd met die sleutel
  - heeft geen invloed op andere sleutels en SA's
  - nieuwe sleutel is totaal onafhankelijk van vorige
  - opgelet: = optie die kan uitgeschakeld worden

IKE overzicht (3): 3 schema's

Authenticatie: methode	Authenticatie: hoe?	Voordelen	Nadelen
vooraf gedeelde sleutels Pre-shared Key	berekenen van (H)MAC over uitgewisselde informatie	eenvoudig	• gedeelde sleutel moet vooraf verdeeld zijn (token) • alleen IP-adres als ID
DSS of RSA handtekeningen	handtekenen van berekende hash over uitgewisselde informatie	• andere ID's dan IP-adres • partnercertificering niet noodzakelijk voor onderhandelingen	• vooraf uitgewisselde certificaten vereist
publieke sleutel encryptie (RSA)	hash van nonce geëncrypteerd met private sleutel	• betere beveiliging bovenop DH uitwisseling • ID beveiligd	• publieke sleutels of certificaten vooraf beschikbaar • intensieve berekeningen

IPSec 67

IKE overzicht (2)

Twee fasen:

- **fase 1:** opzetten van ISAKMP-SA
  - verwezenlijken van een *master secret* waarvan alle andere sleutels voor de SA's zullen worden afgeleid
  - gebruikt om de ISAKMP-boodschappen te beschermen uit fase 2
    - bepalen van cryptografische protocollen en parameters
    - die zorgen voor authenticatie en encryptie
  - gebeurt 1x per dag of week
- **fase 2:** opzetten van IPSec-SA's
  - stelt SA's op, beveiligd door realisaties van fase 1
  - minder complexe uitwisselingen
  - refresh van sleutels bv. om 2 of 3 minuten

IPSec 68

IKE overzicht (3)

- voorziet in gebruik van *Permanent Identifiers*
- oplossing als IP-adres van host vooraf niet gekend is
- permanente identificatie op basis van
  - naam *of*
  - e-mailadres
- gebruik van certificaten met link tussen publieke sleutel en permanente identificatie
- fase1-boodschappen bevatten dergelijke certificaten
- er kan link gelegd worden met tijdelijk IP-adres (boodschap maakt deel uit van IP-datagram)

IKE fase 1: 3 modes

- 3 modes: main, aggressive en base
- Main mode
  - 6 UDP-datagrams worden uitgewisseld
    - **A doet reeks voorstellen aan B, B kiest een voorstel**
    - **sleutelinformatie wordt uitgewisseld**
    - **partijen worden geauthenticeerd**
  - eerste uitwisseling gebeurt op basis van IP-adressen
- Aggressive mode
  - uitwisseling van slechts 3 pakketten
  - eerste bericht bevat reeds DH publieke waarden
  - identiteit niet alleen op basis van IP-adres
  - identiteit is uitgewisseld vóór het bepalen van een gedeelde sleutel
  - kan gebruikt indien initiator A policy kent van B (inbellen nr bedrijf)

IKE fase 1: 3 modes (2)

- Aggressive mode
  - alleen maar voordelen? neen
  - kwetsbaar voor denial of service aanval
    - sturen van veel voorstellen met *spoofed* IP-adressen
    - vergt veel berekeningen voor de ontvanger
  - niet meer aangeraden door de IPSec-werkgroep
- Base mode: 4 datagrammen
  - doel
    - voordelen van aggressive mode
    - nadelen ervan wegwerken
  - er wordt gewacht op 2de datagram van initiator, bewijst zijn bestaan
  - niet alle problemen zijn zo opgelost
    - identiteiten niet beschermd
    - sleutelinformatie slechts uitgewisseld na controle van identiteiten

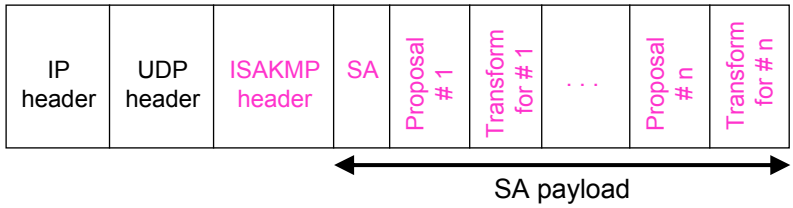
IKE fase 1

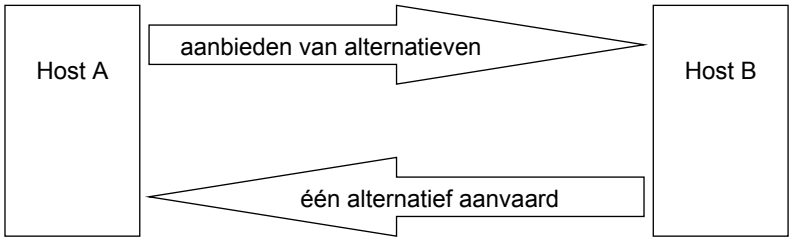
- koude start, geen vroegere sleuteluitwisseling tussen host A en host B
- gebruikt Oakley Main mode (ISAKMP Identity Protect exchange)
- 6 boodschappen:
  - 1&2
    - bepalen eigenschappen van SA's voor ISAKMP
    - onbeveiligd, niet geauthenticeerd
  - 3&4
    - uitwisseling nonces en opstellen master key via Diffie-Hellman
    - onbeveiligd, niet geauthenticeerd
  - 5&6
    - authenticatie van identiteit van beide partijen
    - beveiligd met materiaal bekomen uit 1-4

IKE fase 1 / boodschap 1

- A stuurt een reeks keuzemogelijkheden naar B
  - Vb:
    {3DES, MD5}, {BLF, SHA-1, DH-G1}, {CAST, MD5, DH-G1},
    {3DES, SHA-1,DH-G2}
- B kiest vb. {3DES, SHA-1,DH-G2}
- elk voorstel bevat algritmen voor
  - conventionele encryptie, berekening van hash, sleuteluitwisseling
  - laatste niet indien pre-shared-key is gebruikt
- DH-Gx
  - verwijst naar de groepen van DH-algoritme
  - elke groep heeft ander aantal bits voor modulus-priemgetal
- informatie zit in *transform payload* van de boodschap

IKE fase 1 / boodschap 1





IKE fase 1 (2)

- **Boodschap 1**
  - A zendt naar B cleartext als payload van UDP-pakket
  - source en destination IP adressen zijn die van A en B
  - destination port 500 in UDP-header
  - ISAKMP boodschap bevat 1 of meer voorstellen door B in overweging te nemen
    - ISAKMP-header
      (bevat o.a. 2 cookies als id van SA, zie volgende slide)
    - Security Association veld:
      - identificeert implementatiedomein
      - bedoeling is IPSec-uitwisseling ➔ dus DOI is IP
    - Proposal payload (zie verder)
    - Transform payload (zie verder)

OAKLEY cookie

- pseudorandom getal
  - gegenereerd door elke partij, geëchoed in het antwoord
  - door beide partijen te verifiëren
  - 64 bits
- eisen voor een cookie
  - afhankelijk van de twee partijen (vermijdt gebruik van willekeurige IP-adressen en poorten)
  - generatie en verificatie gebeurt snel (DOS-aanvallen)
  - uitwisselende entiteiten gebruiken ook geheime informatie
- voorbeeld van methode
  - hash (MD5) over
    - IP-adressen van bron en bestemming
    - UDP-poorten van bron en bestemming
    - lokaal gegenereerde geheime waarde (nonce)

ISAKMP-boodschap

- ISAKMP-header
  bevat o.a.
  - Initiator cookie (64 bit pseudorandom getal)
  - Responder cookie (door A op 0 gezet)
  - Exchange type
    - 5 types met verschillend aantal uitgewisselde boodschappen
    - hier = 0, main mode
  - Next payload
    - geeft soort payload aan direct volgend op header
  - Message ID (hier = 0)
  - Flags
  - Length of message
- elke payload bevat een generische header met
  - next header (type van volgende payload)
  - payload lengte

ISAKMP soorten payload

Type	Parameters	Description
Security Association (SA)	Domain of Interpretation, Situation	Used to negotiate security attributes and indicate the DOI and Situation under which negotiation is taking place.
Proposal (P)	Proposal #, Protocol-ID, SPI Size, # of Transforms, SPI	Used during SA negotiation; indicates protocol to be used and number of transforms.
Transform (T)	Transform #, Transform-ID, SA Attributes	Used during SA negotiation; indicates transform and related SA attributes.
Key Exchange (KE)	Key Exchange Data	Supports a variety of key exchange techniques.
Identification (ID)	ID Type, ID Data	Used to exchange identification information.
Certificate (CERT)	Cert Encoding, Certificate Data	Used to transport certificates and other certificate-related information.
Certificate Request (CR)	# Cert Types, Certificate Types, # Cert Auths, Certificate Authorities	Used to request certificates; indicates the types of certificates requested and the acceptable certificate authorities.
Hash (HASH)	Hash Data	Contains data generated by a hash function.
Signature (SIG)	Signature Data	Contains data generated by a digital signa wture function.
Nonce (NONCE)	Nonce Data	Contains a nonce.
Notification (N)	DOI, Protocol-ID, SPI Size, Notify Message Type, SPI, Notification Data	Used to transmit notification data, such as an error condition.
Delete (D)	DOI, Protocol-ID, SPI Size, # of SPIs, SPI (one or more)	Indicates an SA that is no longer valid.

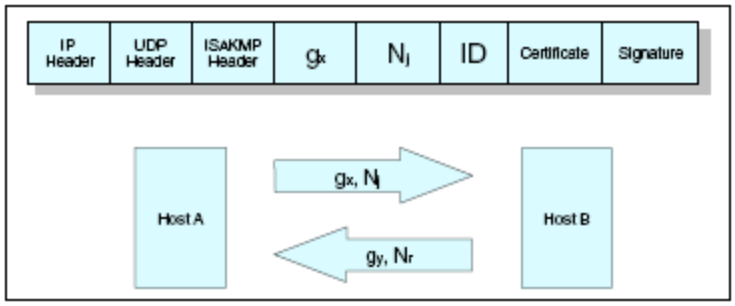
ISAKMP-boodschap (2)

- Proposal payload
  - bevat info over te onderhandelen SA
  - type protocol = PROTO_ISAKMP (hier, kan ook ESP of AH zijn)
  - SPI van afzender = 0 (kan nog niet bepaald worden)
  - aantal transforms (elke transform in 1 transform-payload)
- Transform payload
  - definieert algemeen de transformatie voor de beveiliging
  - vbn: 3DES voor ESP, SHA1 voor AH
  - hier KEY_OAKLEY
  - relevante attributen
    - methoden voor authenticatie
    - functie om randomgetallen te bepalen
    - encryptiemethode

IKE fase 1: (2)

- Boodschap 2
  - antwoord van B op boodschap 1: B kiest 1 alternatief
  - responder cookie wordt ingevuld
  - source en destination IP adressen zijn die van B en A
  - cookie-A en cookie-B dienen als SPI voor de ISAKMP-SA
  - rest is zelfde: 1 proposal met 1 transform payload
- Opmerkingen:
  - SPI van ISAKMP-SA bepaald
  - identiteiten van partijen die beweren A en B te zijn, is nog niet geverifieerd
  - nu begint uitwisseling om sleutels te bepalen

IKE fase 1: (3)



- **Boodschap 3** van A naar B in ISAKMP-payload
  - Diffie-Hellman publieke waarde a^x (x is privaat voor A en geheim) (in Key Exchange payload)
  - nonce N_i (initiator) (in Nonce payload)
  - extra payloads indien authenticatie met certificaat en handtekening (ID, Certificate, Signature) (N_i en ID geëncrypteerd)

IKE fase 1 (4)

- Boodschap 4
  - van B naar A
  - Diffie-Hellman publieke waarde a^y (y is privaat voor B en geheim) (in Key Exchange payload)
  - nonce N_r (responder) (in Nonce payload)
- A & B genereren sleutels
- A en B kennen elk:
  - N_i en N_r
  - x resp. y (private DH-waarde)
  - a^x en a^y (publieke DH-waarde)
  - kunnen a^{xy} berekenen
  - initiator en responder cookies

zie ook volgende dia

IKE fase 1 (5)

A & B genereren elk hetzelfde stel sleutels

- gebeurt a.d.h.v. onderhandelde pseudorandomfctie (prf), vb HMAC
- delen door komma gescheiden ≡ concatenatie
- SKEYID afhankelijk van toepassing
  - authenticatie met digitale handtekening: prf(Ni, Nr, g^{xy})
  - authenticatie publieke sleutels: prf(hash(Ni,Nr),cookieA, cookieB)
  - authenticatie met pre-shared key: prf(presharedkey, Ni, Nr)
- SKEYID_x afhankelijk van toepassing
  - gebruikt in fase2 om sleutels voor "echte" SA's te bepalen
    SKEYID_d = prf(SKEYID,g^{xy},cookieA, cookieB,0)
  - authenticatie van ISAKMP boodschappen
    SKEYID_a = prf(SKEYID, SKEYID_d,g^{xy},cookieA,cookieB,1)
  - encrypteren van ISAKMP boodschappen
    SKEYID_e = prf(SKEYID, SKEYID_a, g^{xy}, cookieA, cookieB,2)

IPSec 83

IKE fase 1 (6)

- A en B hebben identieke sleutels afgeleid
- dienen voor authenticatie en encryptie
- beveiligen de verdere ISAKMP-uitwisselingen, ook in fase 2 (=opzetten van SA)
- tot nu toe zijn identiteiten van beide partijen nog niet geauthenticeerd ten overstaan van mekaar
- gebeurt in boodschap 5:
  - A → B ➔ B authenticeert A

IPSec 84

IKE fase 1: boodschap 5

Uitwisseling ter authentisering

- **Boodschap 5** van A naar B
  - bevat *Identity*, *Certificate* en *Signature payload*
  - identiteit:
    - **id van certificaat, niet noodzakelijk in relatie tot IP-adres**
  - certificaat:
    - **als aanwezig → B verifieert handtekening CA**
    - **als niet aanwezig → B vraagt dit aan CA, LDAP, secure-DNS, …**
  - handtekening: (SAp is volledige SA ***payload*** uit boodschap1)
    niet berekend op payload van bericht, maar op de gemeenschappelijke gegevens
    HASH_I = prf(SKEYID, a^x, a^y, cookieA, cookieB, SAp , ID_A)
    eventueel getekend door A; te valideren door B

IPSec 85

IKE fase 1: boodschap 6

- **Boodschap 6** van B naar A
  - analoog als boodschap 5, andere volgorde parameters
- Opmerkingen:
  - steeds 6 boodschappen, maar inhoud afhankelijk van methode van authenticering
  - ISAKMP-payload in boodschappen 5 en 6 gecodeerd met
    - **algoritmen uit 1 & 2**
    - **sleutels uit 3 & 4**
    dus geen gebruik van ESP of AH in dit stadium, maar beveiliging op applicatieniveau (ISAKMP payload)
  - Message-id (in ISAKMP-hoofding)
    = 0 in fase1
    enige manier om te weten of het gaat om fase1 of fase2
  - ISAKMP boodschap via UDP → geen garantie op ontvangst

IPSec 86

IKE fase 2 (Oakley Quick Mode)

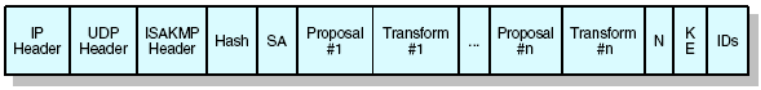
- host **A start onderhandelingsproces** om SA's te bepalen
- **ISAKMP-payload** (niet header) wordt **geëncrypteerd** met onderhandelde methoden uit fase1
- authenticatie d.m.v.
  - hash-functies op materiaal uitgewisseld in fase1 (SKEYID) en fase2
  - fase2 authenticatie is gebaseerd op certificaten op een onrechtstreekse manier :
    gebruikt SKEYID_a uit fase1 dat gecertificeerd is dmv certificaten

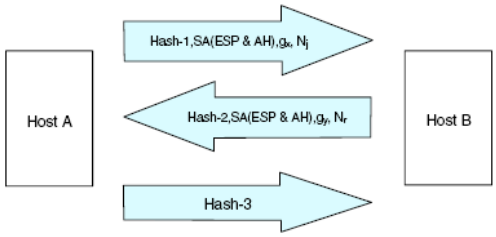
IKE fase 2 (2)

- gebruik van **Oakley Quick mode** in twee modes
  - *zonder* Key Exchange attribuut
    om sleutels te refreshen *zonder* PFS (Perfect Forward Secrecy)
  - *met* Key Exchange attribuut
    om sleutels te refreshen *met* PFS (extra Diffie-Hellman publieke sleuteluitwisseling in boodschappen 1&2)
  - PFS is dus optioneel (raar maar waar)
- **3 boodschappen**

IKE fase 2

- **3 boodschappen**





IPSec 89

IKE fase 2 (3)

- **Boodschap 1** van A naar B
  - authenticeert A en selecteert nonce
  - stelt meerdere SA's voor aan B
  - wisselt Diffie-Hellmann sleutelinfo uit
  - geeft aan of A op eigen houtje handelt of als proxy-onderhandelaar voor andere entiteit
  - ISAKMP boodschap bevat 1 of meer voorstellen
    - ISAKMP-header
    - Hash payload
    - Security Association (hier implementatiedomein, dus IP)
    - Proposal + Transform payloads (meerdere)
    - Nonce
    - Key Exchange payload (indien Perfect Forward Secrecy)
    - ID's (eindpunten van deze SA)

IPSec 90

IKE fase 2 (4)

- **ISAKMP-header** bevat o.a.
  - Initiator en responder cookies uit fase 1
  - Exchange type (hier Quick mode)
  - Message ID gekozen door A (hier ≠ 0)
  - Flags (encryptievlag aan, payload geëncrypteerd)
- **Hash**: HASH_1 = prf(SKEYID_a, M-ID, SA, Nqmi , KE, IDqmi , IDqmr)
- **SA** is de volledige payload van de boodschap, incl. alle voorstellen
- **KE** is nieuwe gekozen publieke Diffie-Hellman waarde
- Nonce payload nieuw gekozen: Nqmi
- Proposal+Transform paren
  - genummerd
  - SPI random gekozen door A
  - Transform bevat protocol (ESP of AH) en algoritme

IKE fase 2 (5)

- **Boodschap 2** van B naar A
  - wordt verstuurd nadat boodschap 1 succesvol werd geauthenticeerd dmv hash_1
  - message-id is zelfde als gekozen door A
  - volgende waarden worden door B aangepast:
    - Hash
      HASH_2 = prf(SKEYID_a, Nqmi , M-ID, SA, Nqmr , KE, IDqmi , IDqmr)
    - Security Association
      - één gekozen uit aanbod
      - SPI gekozen door B
    - Nonce
    - KE: Diffie-Hellman waarde van B

IKE fase 2 (6)

Genereren van de sleutels door A en B:

- in de veronderstelling van PFS
- voor gegevens gestuurd door A en ontvangen door B:
  $KEYMAT_{AB}$ = prf(SKEYID_d, gqm^{xy} , protocol, SPI_B , Nqmi , Nqmr)
- voor gegevens gestuurd door B en ontvangen door A:
  $KEYMAT_{BA}$ = prf(SKEYID_d, gqm^{xy} , protocol, SPI_A , Nqmi , Nqmr)
- het kan nodig zijn dat er meerdere sleutels worden gegenereerd voor vb. volgende situaties:
  - genereren van de integrity check value (ICV) van verzonden gegevens
  - valideren van ICV van ontvangen gegevens
  - encrypteren van verzonden gegevens
  - decrypteren van ontvangen gegevens

IKE fase 2 (7)

- **Boodschap 3** van A naar B
  - bevat alleen ISAKMP header en hash payload
  - HASH_3 = prf(SKEYID_a, 0, M-ID, Nqmi , Nqmr)
  - ontvangen hash wordt door B geverifieerd

Eindelijk is nu gegevensuitwisseling mogelijk !!