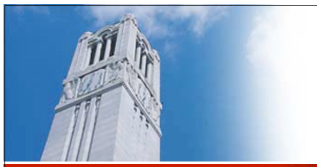


NC STATE UNIVERSITY Computer Science

CSC 774 -- Network Security

Topic 3.1: IKE

IKE Overview

- IKE = ISAKMP + part of OAKLEY + part of SKEME
  - ISAKMP determines
    - How two peers communicate
    - How these messages are constructed
    - How to secure the communication between the two peers
    - No actual key exchange
  - Oakley
    - Key exchange protocol
  - Combining these two requires a Domain of Interpretation (DOI)
    - RFC 2407

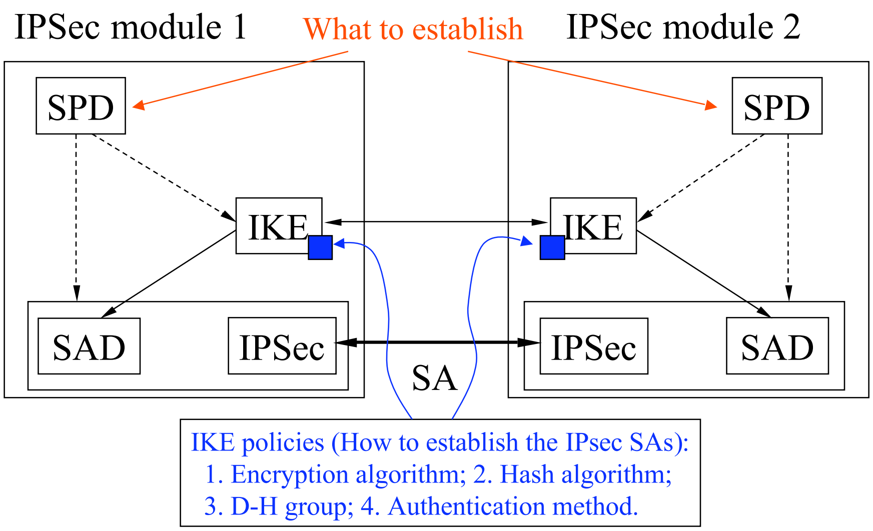
IKE Overview (Cont’d)

- A separate RFC has been published for IKE
  - RFC 2409
- Request-response protocol
  - Initiator
  - Responder
- Two phases
  - Phase 1: Establish an IKE (ISAKMP) SA
    - Essentially the ISAKMP phase 1
    - Bi-directional
  - Phase 2: Use the IKE SA to establish IPsec SAs
    - Key exchange phase
    - Directional

IKE Overview (Cont’d)

- Several Modes
  - Phase 1:
    - Main mode: identity protection
    - Aggressive mode
  - Phase 2:
    - Quick mode
  - Other modes
    - New group mode
      - Establish a new group to use in future negotiations
      - Not in phase 1 or 2;
      - Must only be used after phase 1
    - Informational exchanges
      - ISAKMP notify payload
      - ISAKMP delete payload

IPSEC Architecture Revisited



IKE policies (How to establish the IPsec SAs):
1. Encryption algorithm; 2. Hash algorithm;
3. D-H group; 4. Authentication method.

A Clarification About PFS

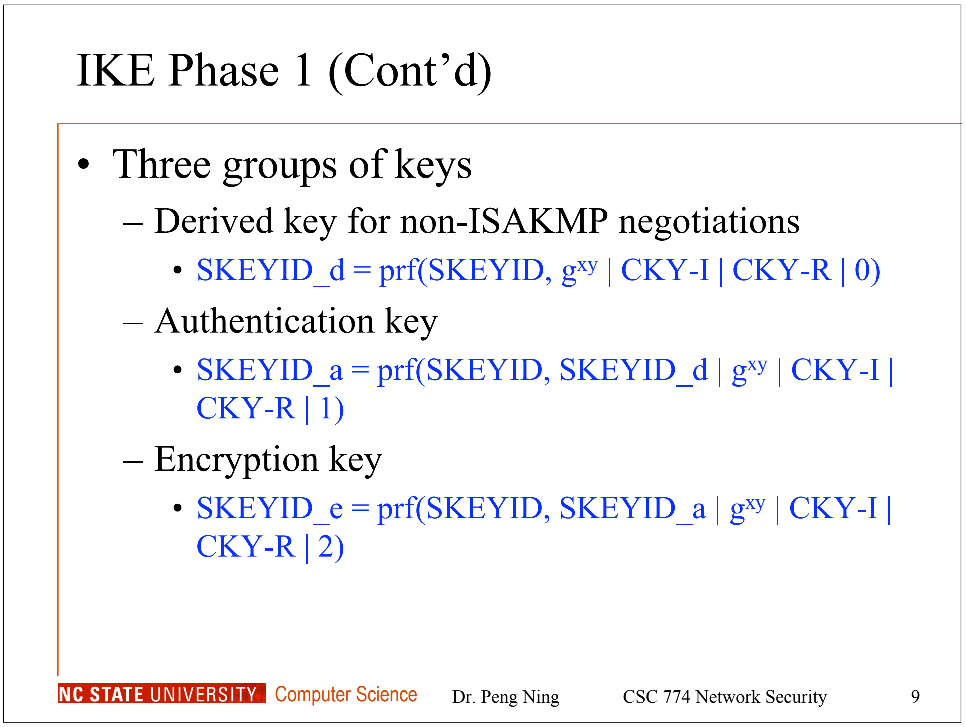
- In RFC 2409:
  - When used in the memo Perfect Forward Secrecy (PFS) refers to the notion that compromise of a single key will permit access to only data protected by a single key.
  - The key used to protect transmission of data MUST NOT be used to derive any additional keys.
  - If the key used to protect transmission of data was derived from some other keying material, that material MUST NOT be used to derive any more keys.
- Is this consistent with what we discussed?

IKE Phase 1

- Four authentication methods
  - – Digital signature
  - – Authentication with public key encryption
  - – The above method revised
  - – Authentication with a pre-shared key

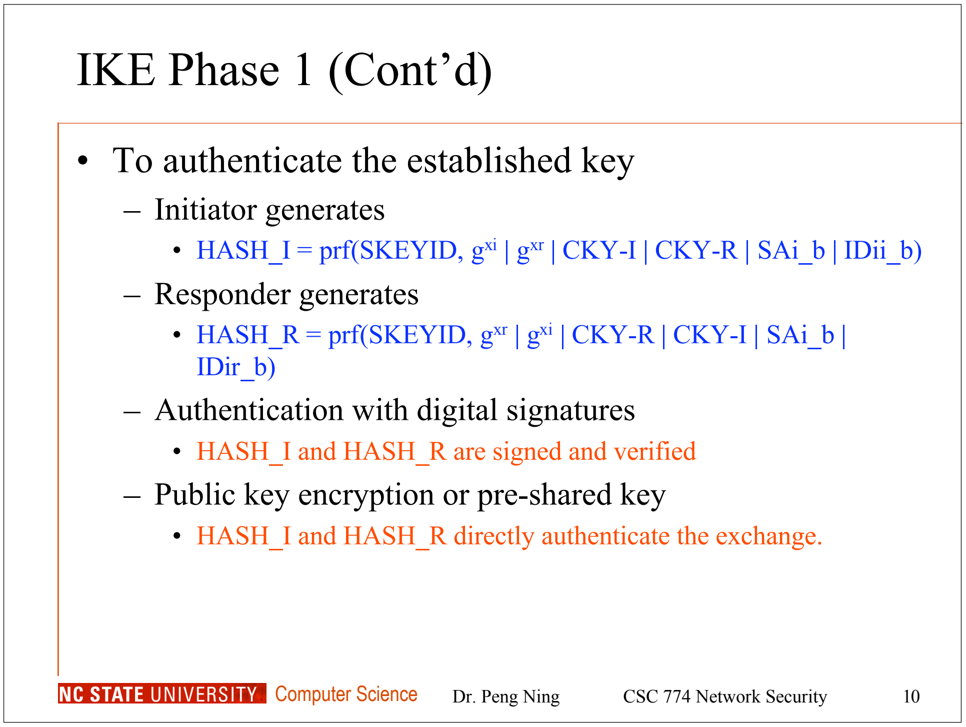
IKE Phase 1 (Cont’d)

- IKE Phase 1 goal:
  - – Establish a shared secret SKEYID
  - – With signature authentication
    - SKEYID = prf(Ni_b | Nr_b, g^{xy})
  - – With public key encryption
    - SKEYID = prf(hash(Ni_b | Nr_b), CKY-I | CKY-R)
  - – With pre-shared key
    - SKEYID = prf(pre-shared-key, Ni_b | Nr_b)
  - – Notations:
    - prf: keyed pseudo random function prf(key, message)
    - CKY-I/CKY-R: I’s (or R’s) cookie
    - Ni_b/Nr_b: the body of I’s (or R’s) nonce



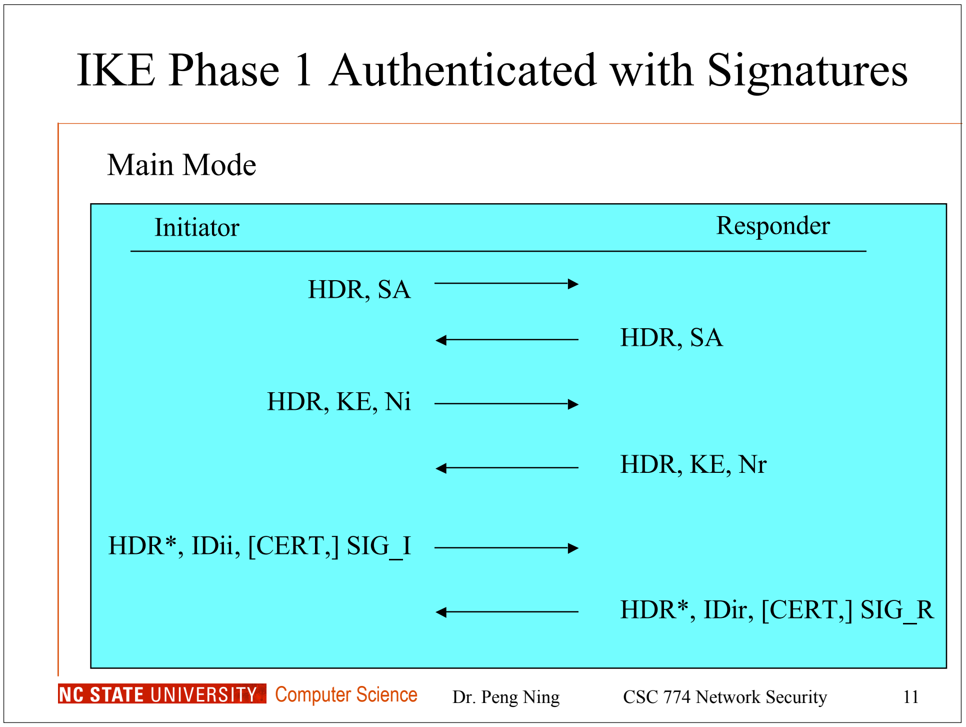
IKE Phase 1 (Cont’d)

- Three groups of keys
  - Derived key for non-ISAKMP negotiations
    - SKEYID_d = prf(SKEYID, g^{xy} | CKY-I | CKY-R | 0)
  - Authentication key
    - SKEYID_a = prf(SKEYID, SKEYID_d | g^{xy} | CKY-I | CKY-R | 1)
  - Encryption key
    - SKEYID_e = prf(SKEYID, SKEYID_a | g^{xy} | CKY-I | CKY-R | 2)



IKE Phase 1 (Cont’d)

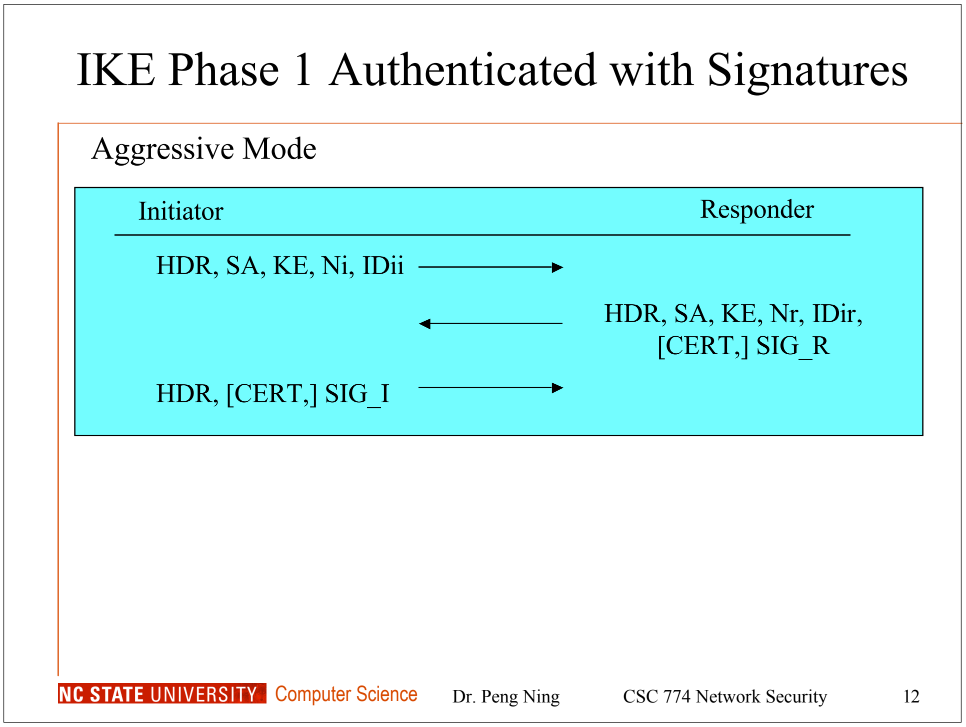
- To authenticate the established key
  - Initiator generates
    - HASH_I = prf(SKEYID, g^{xi} | g^{xr} | CKY-I | CKY-R | SAi_b | IDii_b)
  - Responder generates
    - HASH_R = prf(SKEYID, g^{xr} | g^{xi} | CKY-R | CKY-I | SAi_b | IDir_b)
  - Authentication with digital signatures
    - HASH_I and HASH_R are signed and verified
  - Public key encryption or pre-shared key
    - HASH_I and HASH_R directly authenticate the exchange.



IKE Phase 1 Authenticated with Signatures

Main Mode

Initiator		Responder
HDR, SA	→	
	←	HDR, SA
HDR, KE, Ni	→	
	←	HDR, KE, Nr
HDR*, IDii, [CERT,] SIG_I	→	
	←	HDR*, IDir, [CERT,] SIG_R



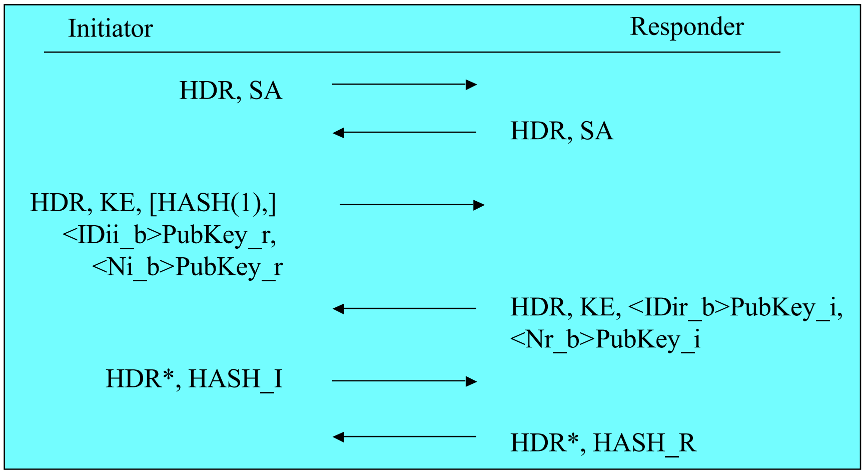
IKE Phase 1 Authenticated with Signatures

Aggressive Mode

Initiator		Responder
HDR, SA, KE, Ni, IDii	→	
	←	HDR, SA, KE, Nr, IDir, [CERT,] SIG_R
HDR, [CERT,] SIG_I	→	

IKE Phase 1 Authenticated with Public Key Encryption

Main Mode



Initiator | Responder

HDR, SA →

← HDR, SA

HDR, KE, [HASH(1),] <IDii_b>PubKey_r, <Ni_b>PubKey_r →

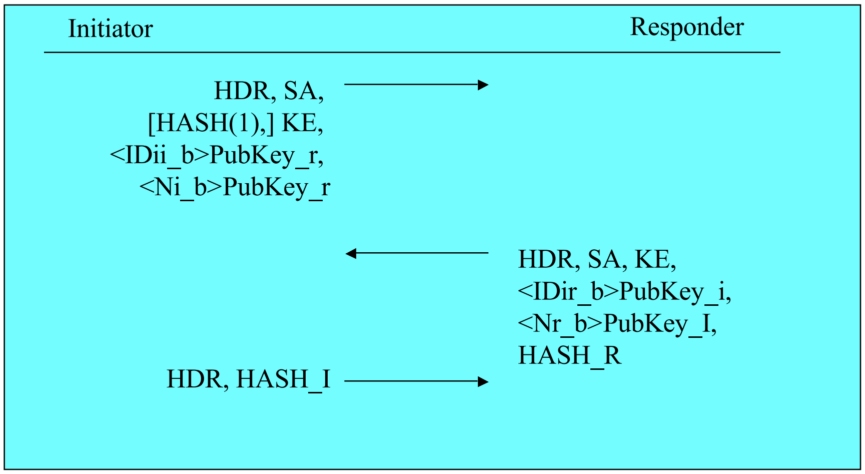
← HDR, KE, <IDir_b>PubKey_i, <Nr_b>PubKey_i

HDR*, HASH_I →

← HDR*, HASH_R

IKE Phase 1 Authenticated with Public Key Encryption

Aggressive Mode



Initiator | Responder

HDR, SA, [HASH(1),] KE, <IDii_b>PubKey_r, <Ni_b>PubKey_r →

← HDR, SA, KE, <IDir_b>PubKey_i, <Nr_b>PubKey_I, HASH_R

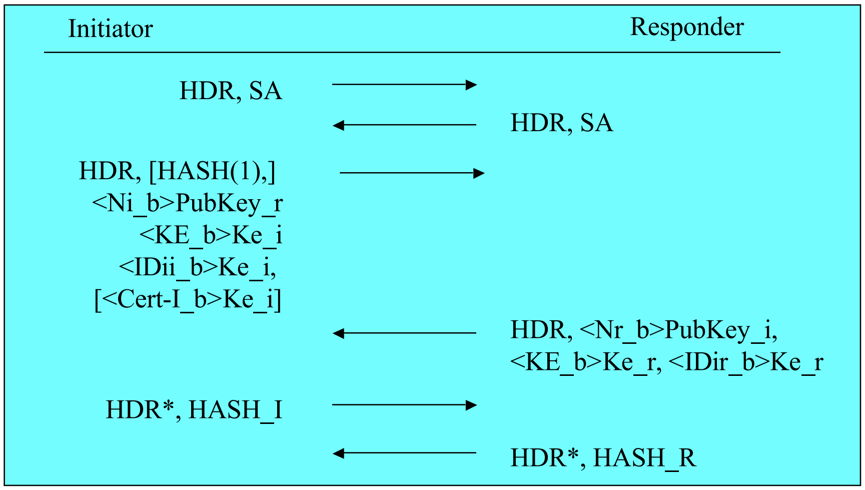
HDR, HASH_I →

Observations

- Authenticated using public key encryption
  - No non-repudiation
    - No evidence that shows the negotiation has taken place.
  - More difficult to break
    - An attacker has to break both DH and public key encryption
  - Identity protection is provided in aggressive mode.
  - Four public key operations
    - Two public key encryptions
    - Two public key decryptions

IKE Phase 1 Authenticated with A Revised Mode of Public Key Encryption

Main Mode



Initiator

Responder

HDR, SA →

← HDR, SA

HDR, [HASH(1),]
<Ni_b>PubKey_r
<KE_b>Ke_i
<IDii_b>Ke_i,
[<Cert-I_b>Ke_i] →

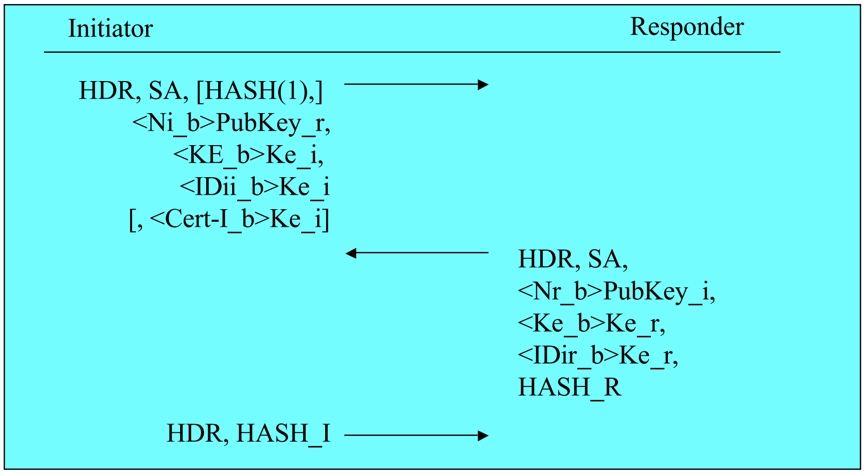
← HDR, <Nr_b>PubKey_i,
<KE_b>Ke_r, <IDir_b>Ke_r

HDR*, HASH_I →

← HDR*, HASH_R

IKE Phase 1 Authenticated with A Revised Mode of Public Key Encryption

Aggressive Mode



Initiator

Responder

HDR, SA, [HASH(1),]
<Ni_b>PubKey_r,
<KE_b>Ke_i,
<IDii_b>Ke_i
[, <Cert-I_b>Ke_i]

HDR, SA,
<Nr_b>PubKey_i,
<Ke_b>Ke_r,
<IDir_b>Ke_r,
HASH_R

HDR, HASH_I

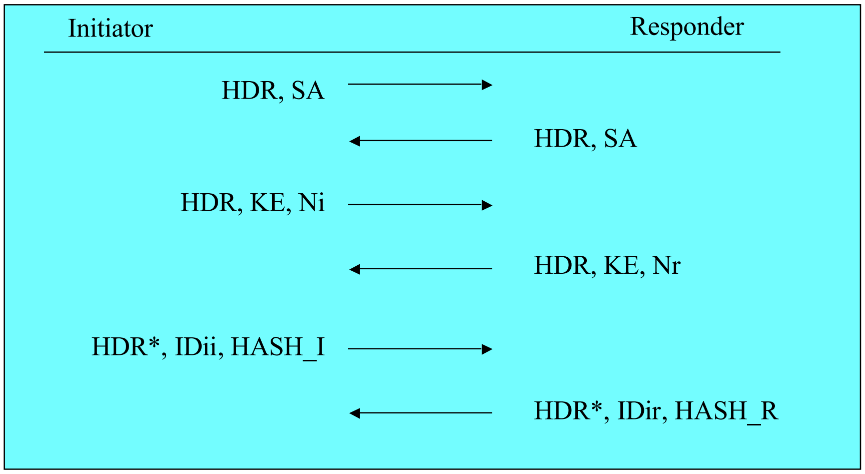
Further Details

Ne_i=prf(Ni_b, CKY-I)
Ne_r=prf(Nr_b, CKY-R)

- Ke_i and Ke_r are taken from Ne_i and Ne_r, respectively.

IKE Phase 1 Authenticated with Pre-Shared Key

Main Mode



Initiator

Responder

HDR, SA →

← HDR, SA

HDR, KE, Ni →

← HDR, KE, Nr

HDR*, IDii, HASH_I →

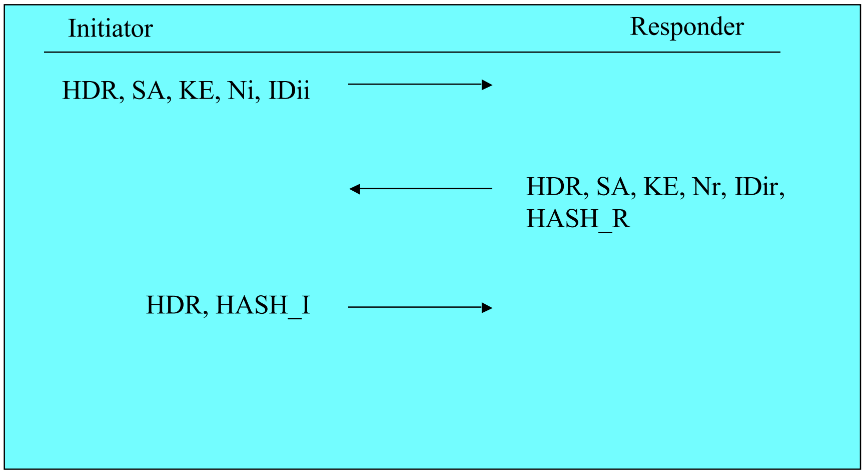
← HDR*, IDir, HASH_R

IKE Phase 1 Authenticated with Pre-Shared Key (Cont'd)

- What provide the authentication?
- Why does it work?

IKE Phase 1 Authenticated with Pre-Shared Key

Aggressive Mode



Initiator

Responder

HDR, SA, KE, Ni, IDii →

← HDR, SA, KE, Nr, IDir, HASH_R

HDR, HASH_I →

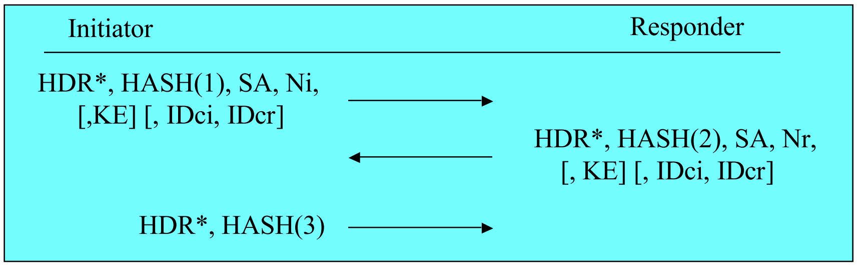
IKE Phase 2 -- Quick Mode

- Not a complete exchange itself
  - Must be bound to a phase 1 exchange
- Used to derive keying materials for IPsec SAs
- Information exchanged with quick mode must be protected by the ISAKMP SA
- Essentially a SA negotiation and an exchange of nonces
  - Generate fresh key material
  - Prevent replay attack

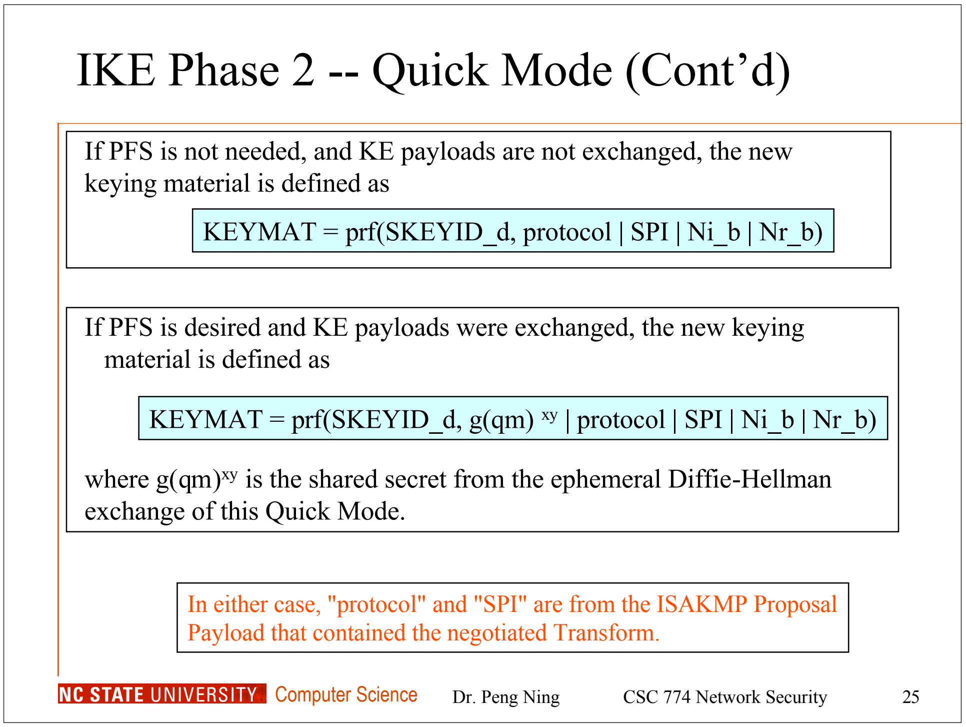
IKE Phase 2 -- Quick Mode (Cont'd)

- Basic Quick Mode
  - Refresh the keying material derived from phase 1
- Quick Mode with optional KE payload
  - Transport additional exponentiation
  - Provide PFS

IKE Phase 2 -- Quick Mode (Cont'd)


Initiator
Responder
HDR*, HASH(1), SA, Ni, [,KE] [, IDci, IDcr]
HDR*, HASH(2), SA, Nr, [, KE] [, IDci, IDcr]
HDR*, HASH(3)

HASH(1) = prf(SKEYID_a, M-ID | SA | Ni [| KE] [| IDci | IDcr)
HASH(2) = prf(SKEYID_a, M-ID | Ni_b | SA | Nr [| KE] [| IDci | IDcr)
HASH(3) = prf(SKEYID_a, 0 | M-ID | Ni_b | Nr_b)



IKE Phase 2 -- Quick Mode (Cont'd)

If PFS is not needed, and KE payloads are not exchanged, the new keying material is defined as

KEYMAT = prf(SKEYID_d, protocol | SPI | Ni_b | Nr_b)

If PFS is desired and KE payloads were exchanged, the new keying material is defined as

KEYMAT = prf(SKEYID_d, $g(qm)^{xy}$ | protocol | SPI | Ni_b | Nr_b)

where $g(qm)^{xy}$ is the shared secret from the ephemeral Diffie-Hellman exchange of this Quick Mode.

In either case, "protocol" and "SPI" are from the ISAKMP Proposal Payload that contained the negotiated Transform.