# Draft-smyslov-ipsecme-ikev2-downgrade-prevention

IPSECME — IETF 123

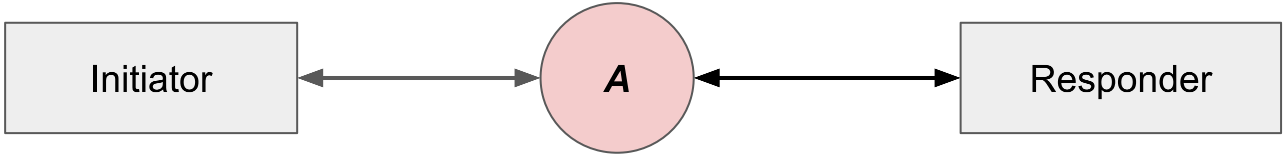
**Valery Smyslov**, Christopher Patton

# Overview

- IKEv2 is vulnerable to a Logjam-like attack in which the attacker learns the key exchanged between an initiator and responder
  - If the initiator and responder prefer a **strong** key exchange (KE) algorithm, but both support some **weak** KE, then an active attacker can ***downgrade*** them to the **weak** KE
  - Attacker must be able to break the **weak** KE online
- Allows a quantum attacker to bypass post-quantum (PQ) KE ⇒ draft-ietf-ipsecme-ikev2-mlkem is vulnerable
- Mitigations
  - Disable classical KE (Diffie-Hellman, DH) by Q-day ⇒ lose connectivity with legacy clients
  - Make IKEv2 downgrade resilient ⇒ draft-smyslov-ipsecme-ikev2-downgrade-prevention

# Preconditions

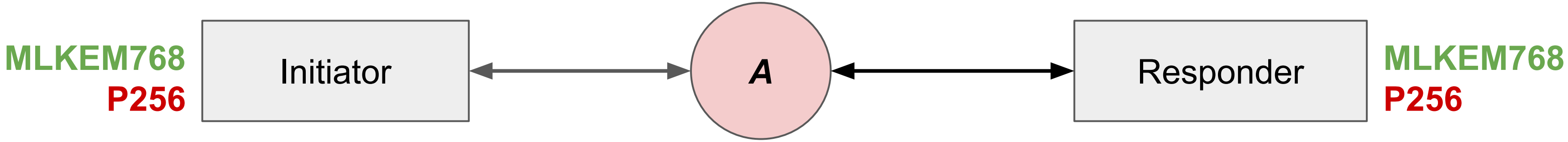
**#1** Attacker is on-path and can modify messages



# Preconditions

**#1** Attacker is on-path and can modify messages

**#2** Both parties prefer PQ, but fallback to non-PQ if PQ not supported by peer

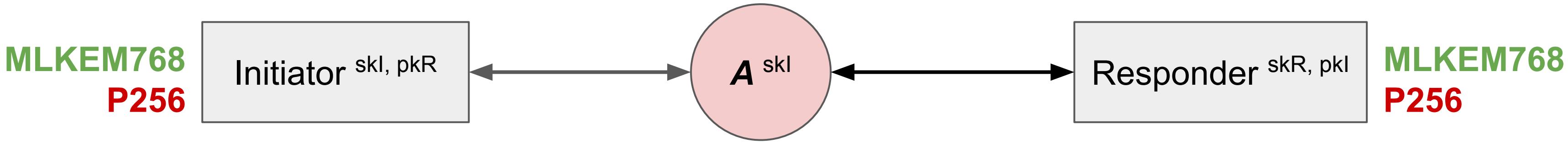


# Preconditions

**#1** Attacker is on-path and can modify messages

**#2** Both parties prefer PQ, but fallback to non-PQ if PQ not supported by peer

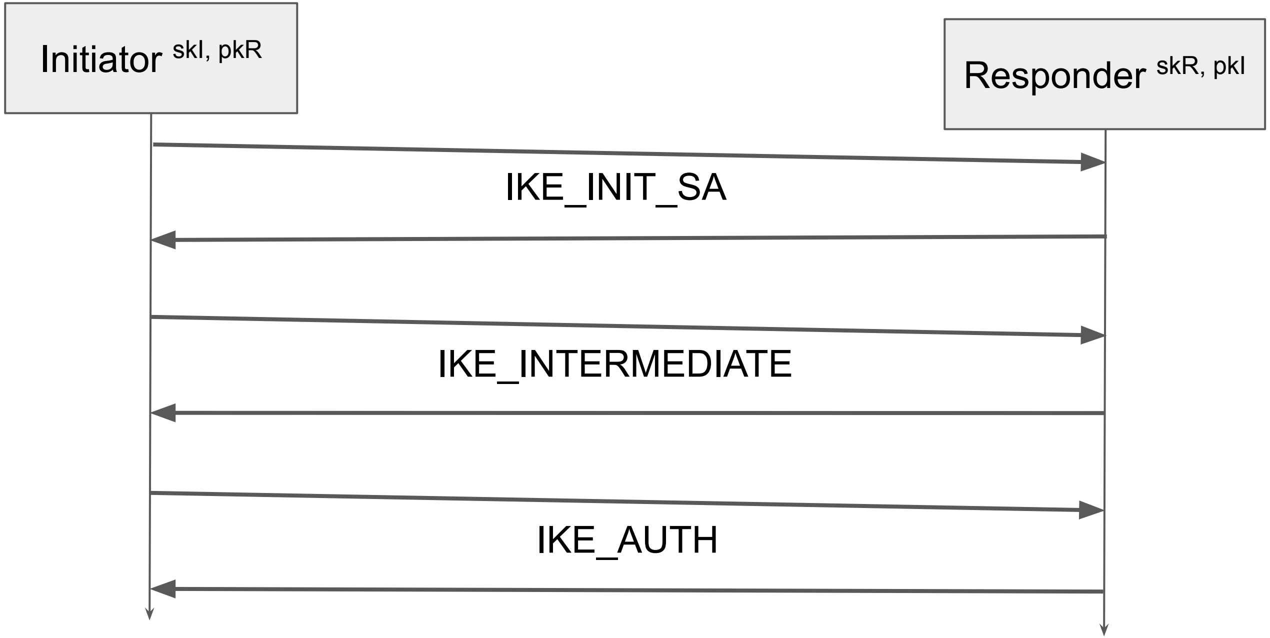
**#3** Attacker has compromised one party's signing key*




* Not strictly necessary; a stronger attack is possible

# Vulnerability

Each party only signs its own IKE_SA_INIT exchange messages.



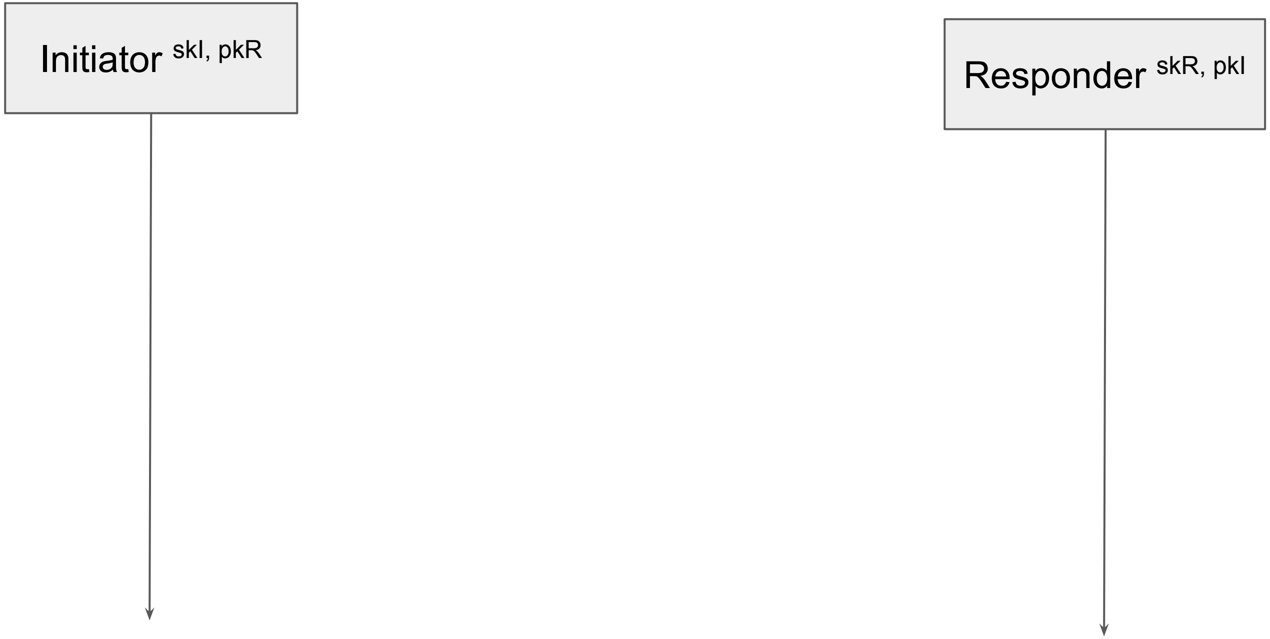
# Vulnerability

Proposals:

**P1**: **P256** + **MLKEM768**

**P0**: **P256** only

Each party only signs its own IKE_SA_INIT exchange messages.



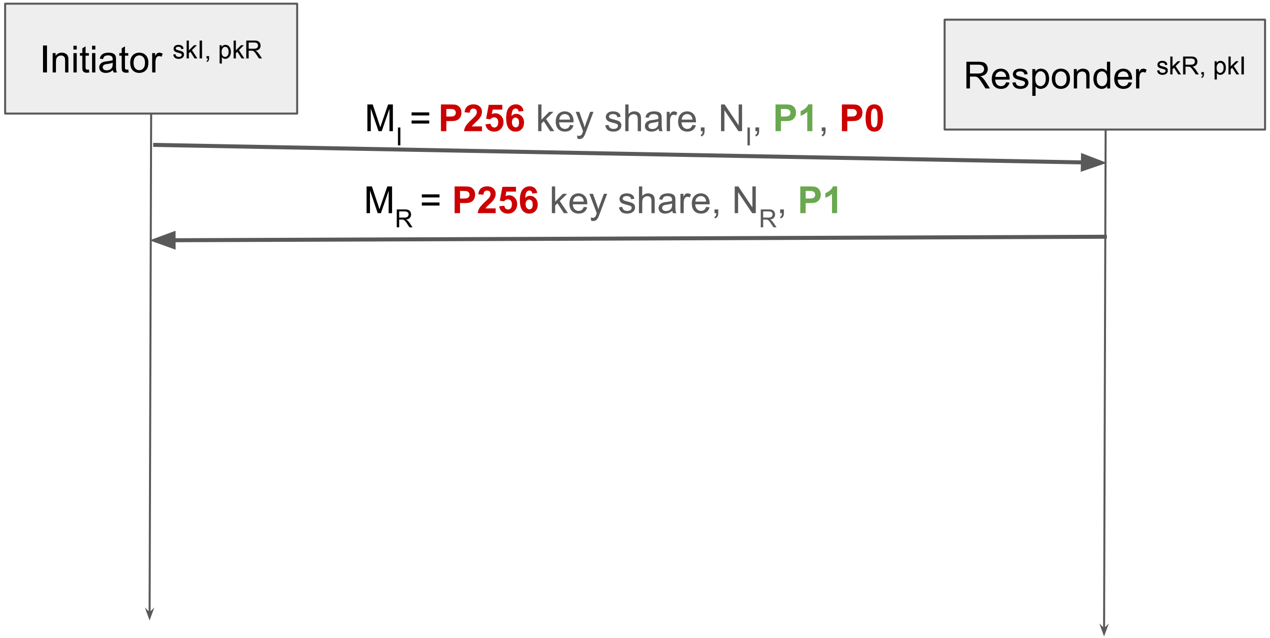
# Vulnerability

Proposals:

**P1**: **P256** + **MLKEM768**

**P0**: **P256** only

Each party only signs its own IKE_SA_INIT exchange messages.



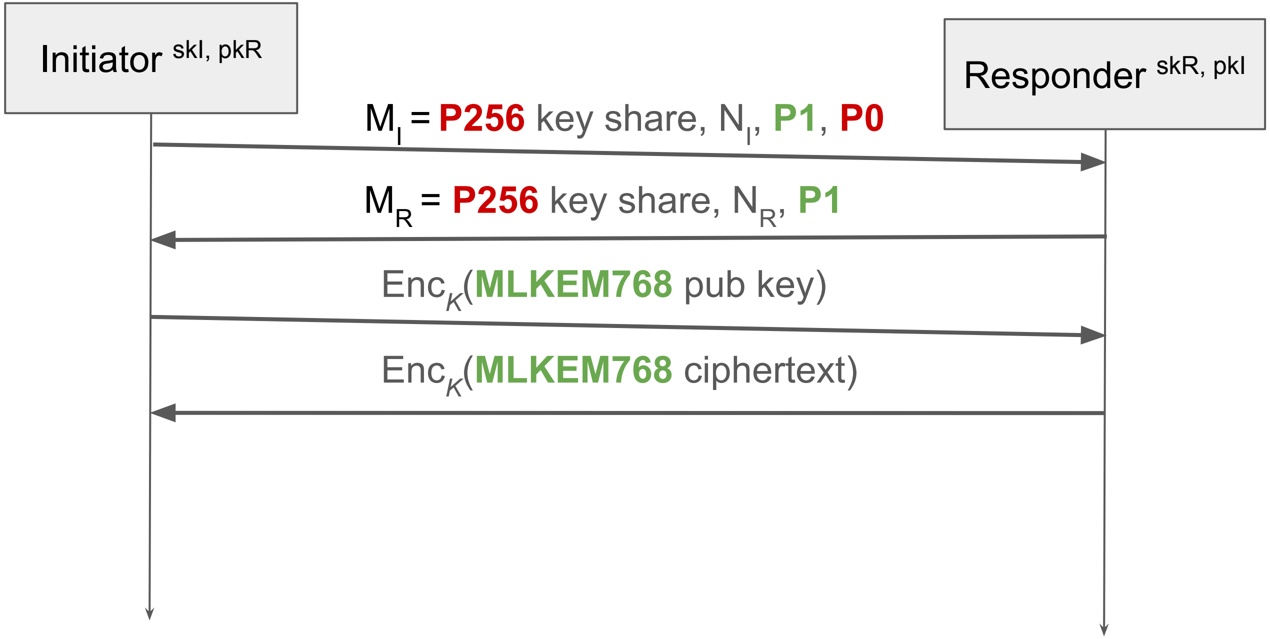
# Vulnerability

Each party only signs its own IKE_SA_INIT exchange messages.

Proposals:

**P1**: **P256** + **MLKEM768**

**P0**: **P256** only



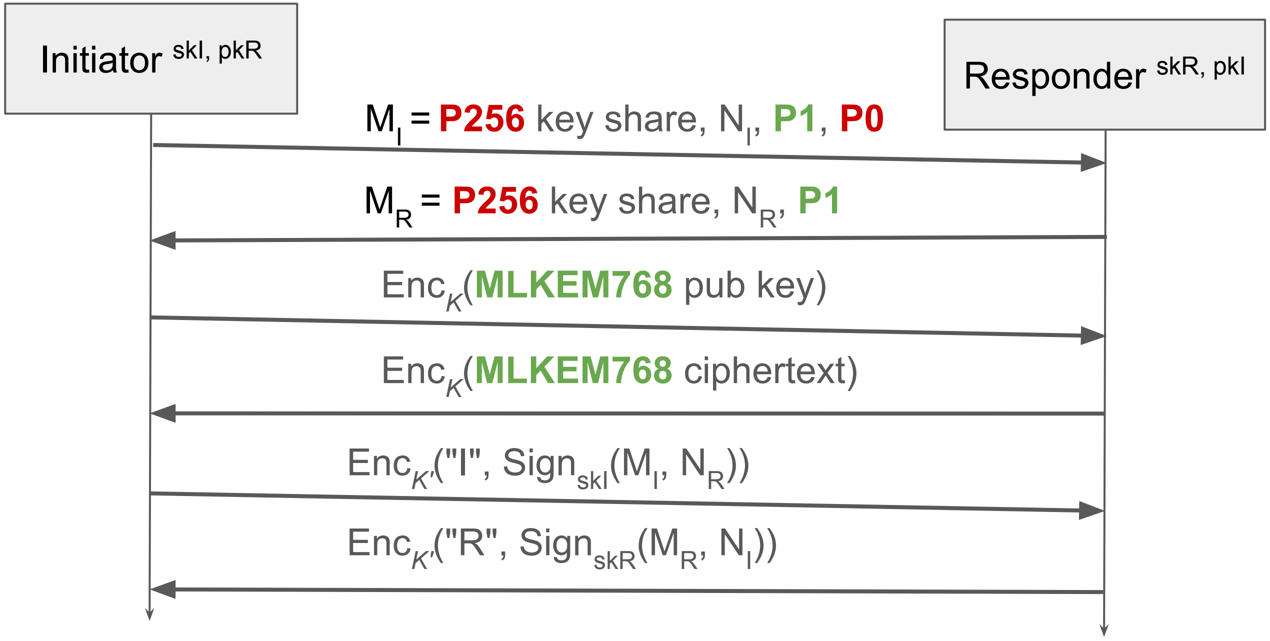
# Vulnerability

Proposals:

**P1**: **P256** + **MLKEM768**

**P0**: **P256** only

Each party only signs its own IKE_SA_INIT exchange messages.



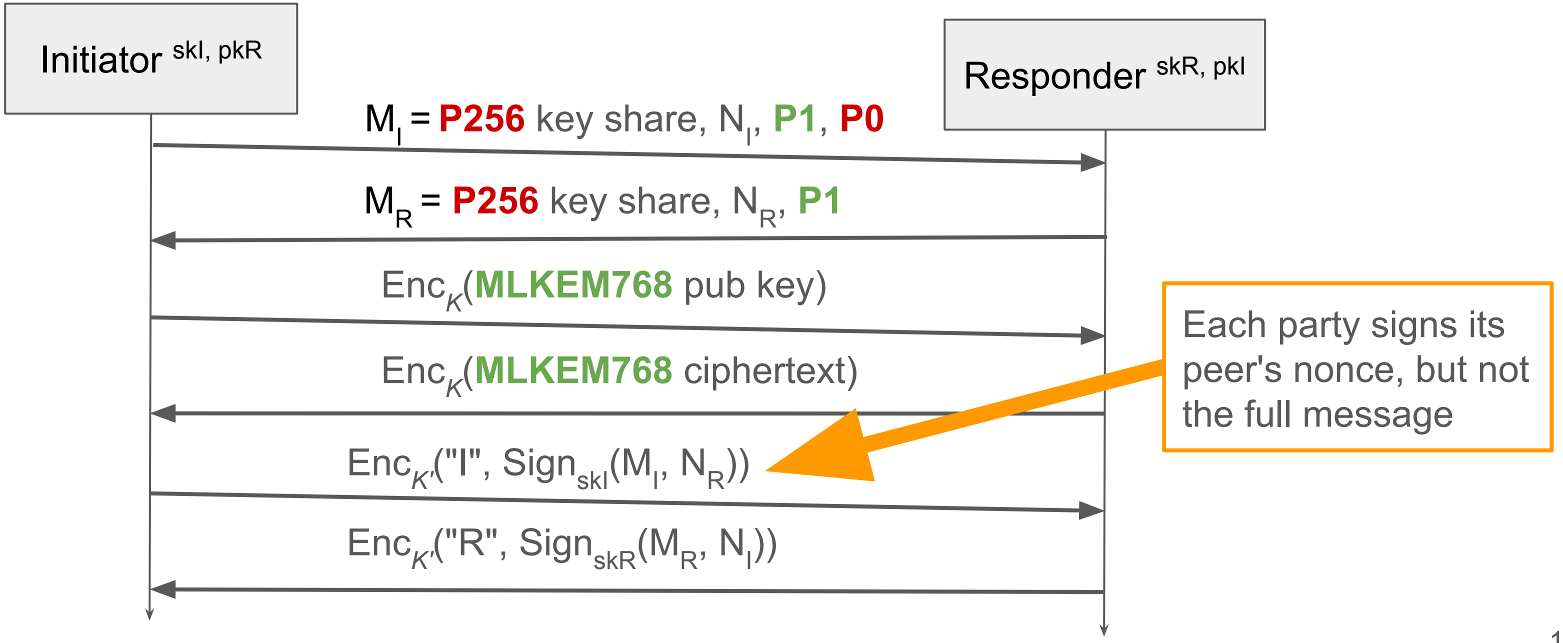
# Vulnerability

Each party only signs its own IKE_SA_INIT exchange messages.

Proposals:

**P1**: **P256** + **MLKEM768**

**P0**: **P256** only



# Attack

Vuln.

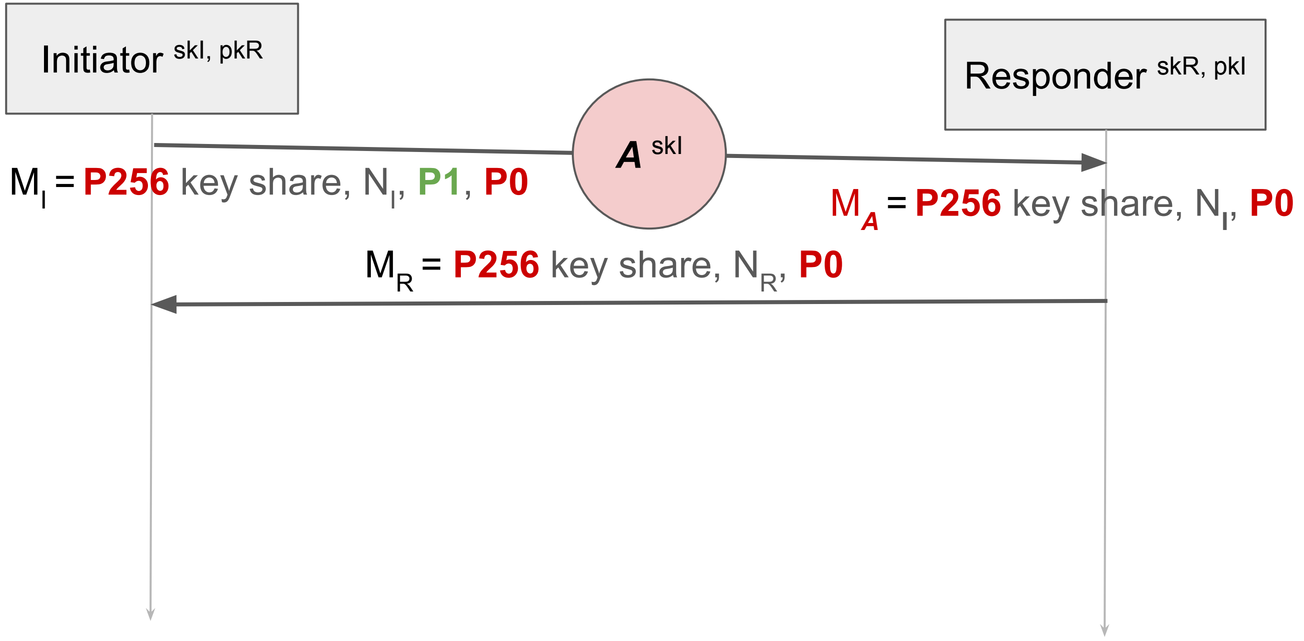
Each party only signs its own IKE_SA_INIT exchange messages.

Preconditions

**#1** Attacker is on-path and can modify messages

**#2** Both parties prefer PQ, but fallback to non-PQ if PQ not supported by peer

**#3** Attacker has compromised one party's signing key



# Attack

**Vuln.**

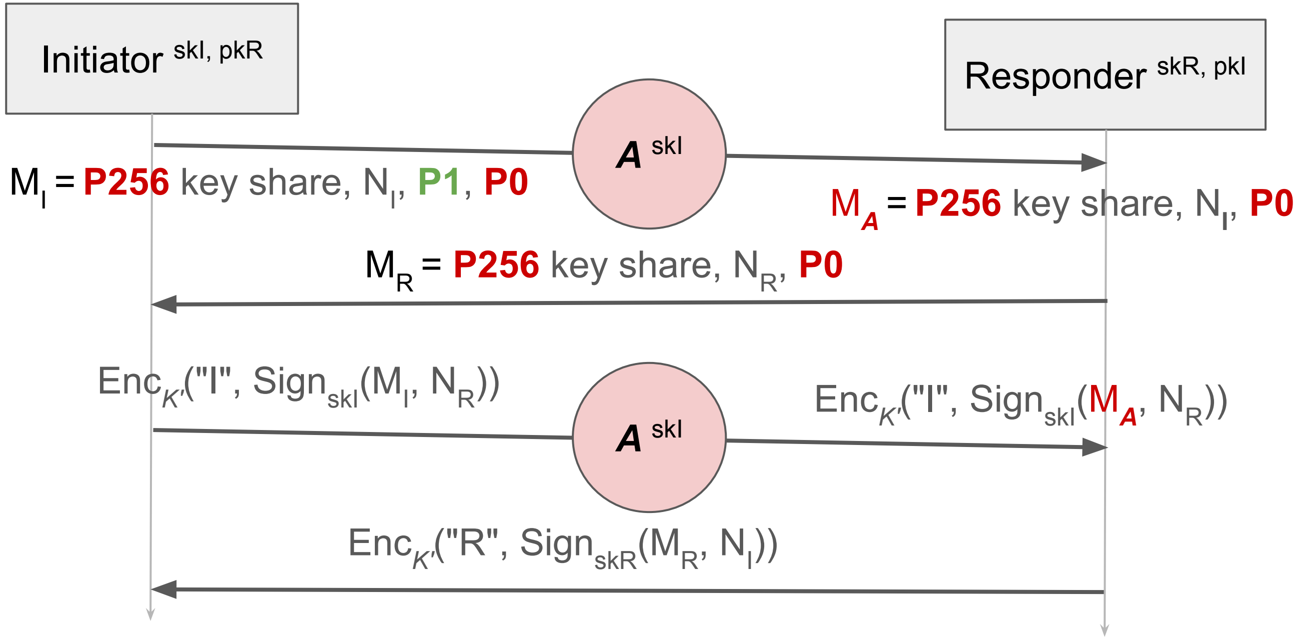
Each party only signs its own IKE_SA_INIT exchange messages.

**Preconditions**

**#1** Attacker is on-path and can modify messages

**#2** Both parties prefer PQ, but fallback to non-PQ if PQ not supported by peer

**#3** Attacker has compromised one party's signing key



# Attack

Vuln.

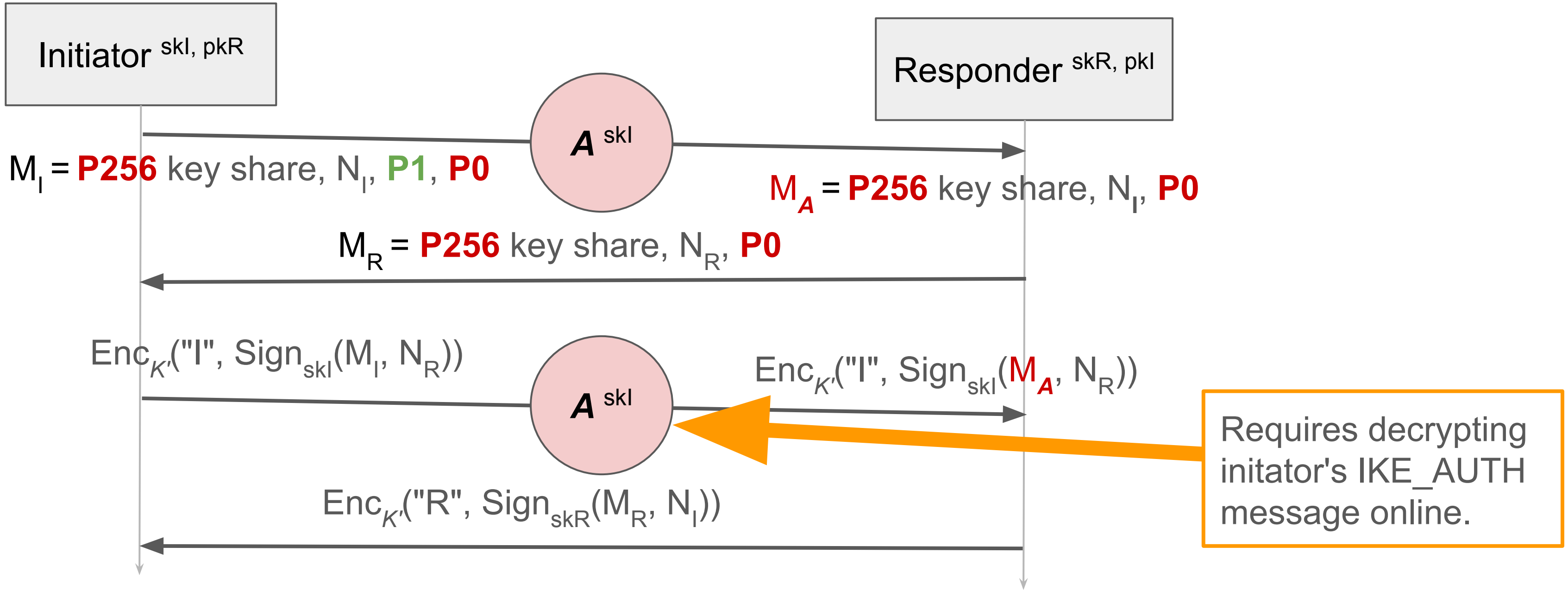
Each party only signs its own IKE_SA_INIT exchange messages.

Preconditions

**#1** Attacker is on-path and can modify messages

**#2** Both parties prefer PQ, but fallback to non-PQ if PQ not supported by peer

**#3** Attacker has compromised one party's signing key



# Attack

Vuln.

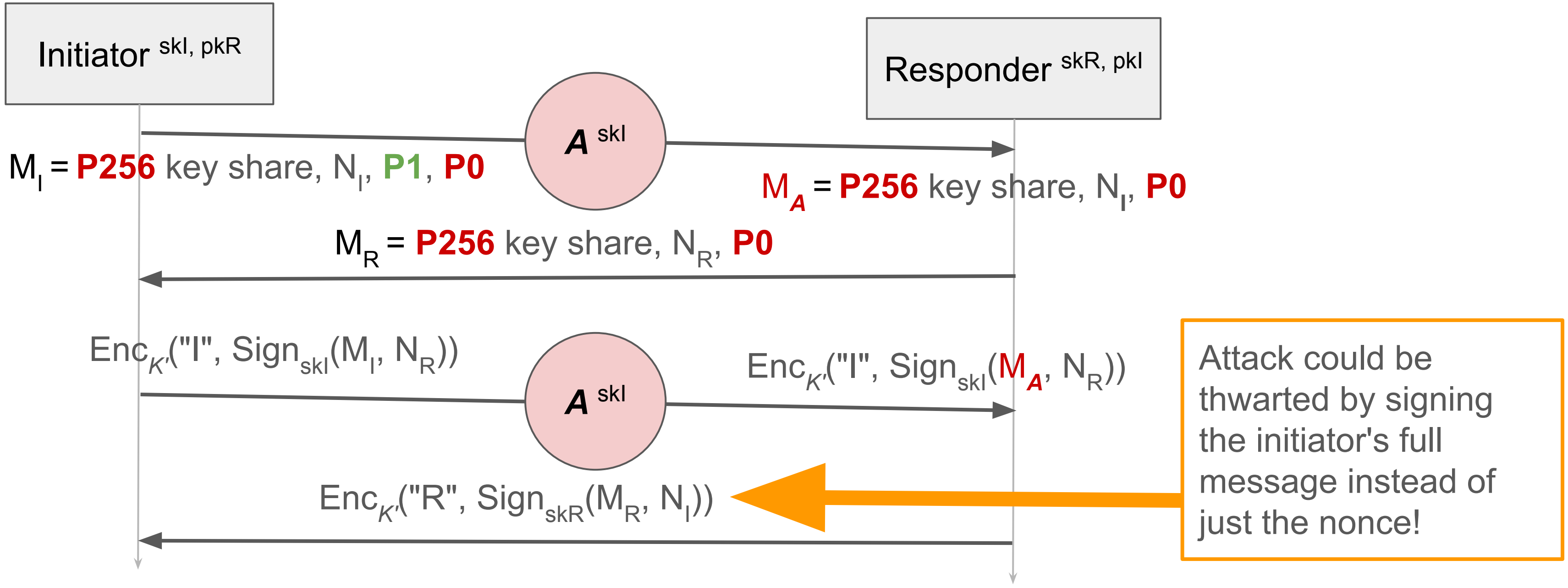
Each party only signs its own IKE_SA_INIT exchange messages.

Preconditions

**#1** Attacker is on-path and can modify messages

**#2** Both parties prefer PQ, but fallback to non-PQ if PQ not supported by peer

**#3** Attacker has compromised one party's signing key



# Stronger variant

**Vuln.**

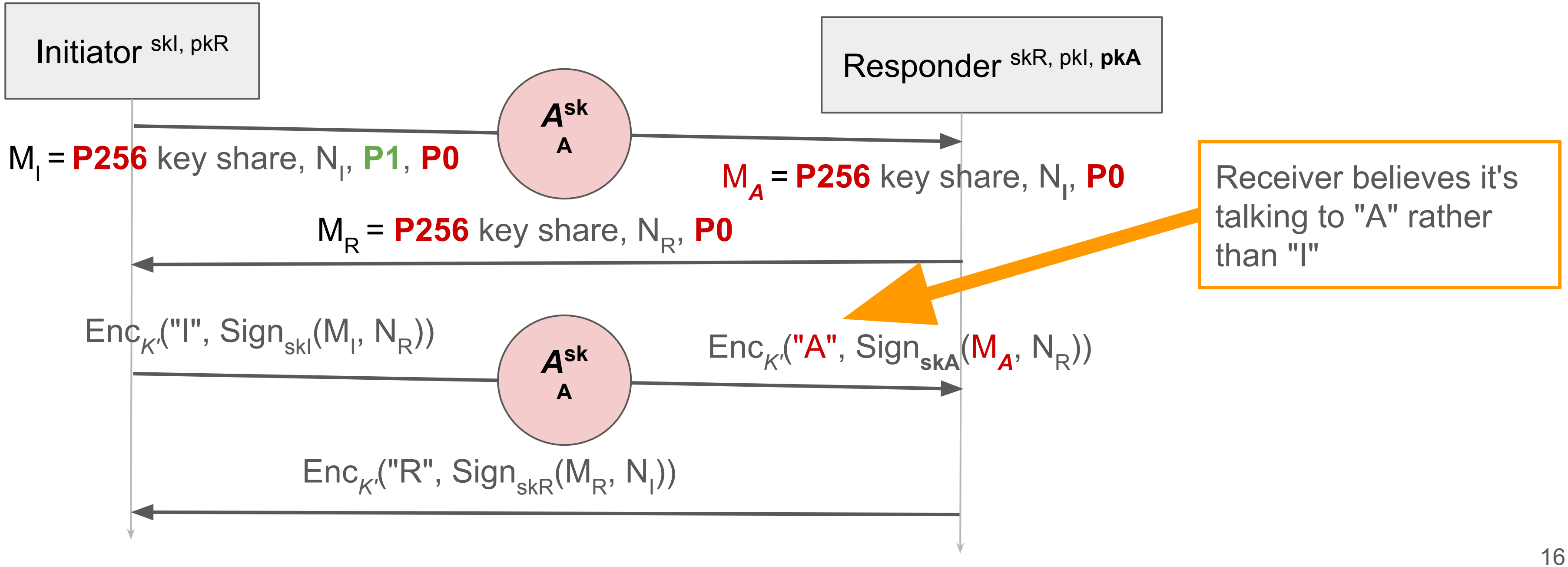
Each party only signs its own IKE_SA_INIT exchange messages.

**Preconditions**

**#1** Attacker is on-path and can modify messages

**#2** Both parties prefer PQ, but fallback to non-PQ if PQ not supported by peer

**#3 Attacker has access to *any* trusted peer's signing key (not necessarily the victim's)**



# Mitigations

- Option #1: Disable non-PQ KE by Q-day
  - Sacrifice connectivity with legacy peers
    - Question: Could we disable non-PQ only for peers who are known out-of-band to support PQ? Peer is not identified until its IKE_AUTH message.
- Option #2: Add downgrade prevention to IKEv2

# draft-smyslov-ipsecme-ikev2-downgrade-prevention

New extension for IKEv2:

- Overrides authentication logic: each party signs ***the entire IKE_AUTH exchange*** rather only only its outbound messages (similar to TLS 1.3).
- Support for extension is ***always*** notified in IKE_INIT_SA (peers that don't support the extension will ignore it).
- If you support the extension ***and*** your peer notifies support, then you always use the new authentication logic.

How does this prevent downgrades?

- The parties confirm they've had the same conversation.
- If attacker rewrites IKE_INIT_SA so that it doesn't notify support, then authentication will fail if both support the extension.

# Best option for mitigation the downgrade attack?

1. Disable classical-only KE by Q-day?
2. Adopt draft-smyslov-ipsecme-ikev2-downgrade-prevention? If so, are we ready for an adoption call?
3. Something else?