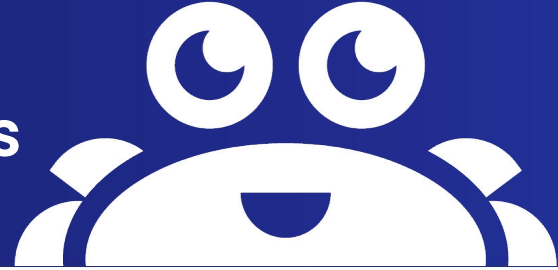


# ~~DH group~~ Key Exchange Method issues



- Initial Exchanges create a Child SA with DH of IKE SA setting.
  - No consistent behaviour if configuration IKE DH != Child DH
- Neither end knows if a rekey will require PFS.
  - and if so, which DH to use.
- Cause: IKE\_AUTH should have contained Child DH proposal.
- Problem: Humans keep using different DH for IKE and ESP.
- Result: IPsec connection works, then fails hour(s) later.....

# DH Issues: Initial Exchanges

- If negotiated IKE DH is not a valid configured Child DH:
  - On Responder:
    - If pfs=no, no issues
    - If pfs=yes, return NO\_PROPOSAL\_CHOSEN?
    - If pfs=yes, assume peer has same Child DH?
  - On Initiator:
    - if pfs=no, no issues
    - if pfs=yes, refuse configuration to load ?
    - if pfs=yes, send Informational Delete after negotiation ?
    - if pfs=yes, assume peer has same Child DH? (problematic)

# DH Issues: REKEY of Initial Child

- Peer's configured child DH group(s) has not been negotiated yet and is unknown
- Rekey with pfs SHOULD use same DH group
- If rekey proposed Child DH is not IKE SA DH:
  - On Responder:
    - Return INVALID\_KEY(dh) ? [wrong, rekey DH group matches KE payload]
    - Return NO\_PROPOSAL\_CHOSEN ? [right, but too confusing]
    - Return another new error code? UNEXPECTED\_DH\_KEY or something?
    - Accept any/none DH, immediately rekey IKE SA to gain pfs with IKE DH

# DH Issues: Additional complications

- Microsoft Windows IKEv2 configured for DH14, rekeys with DH2
  - Probably thinks DH14 is for IKE SA, libreswan has ms-dh-downgrade=yes|no
- draft-ietf-ipsecme-ikev2-sa-ts-payloads-opt does not know PFS / DH settings
- Implementation interop issues with DH/KE transform “DH\_NONE” vs no transform
- Implementation interop issues with initiator rekey using configured parameters instead of established parameters

# Libreswan interoperability issues

transitions are:  
 AUTH -> REKEY -> REKEY  
 CHILD -> REKEY

esp	pfs	ms dh	w/o	AUTH CHILD	mainline REKEY AUTH CHILD	v4.5 REKEY AUTH CHILD	v4.6 REKEY AUTH CHILD	second CHILD	REKEY second CHILD or REKEY
algs	no	no	no	algs	OLD-none	algs-none	OLD-none	algs-none	OLD-none
algs	no	no	yes	algs	algs-none			algs-none	algs-none
algs	no	yes	no	algs	OLD-none	algs-none	OLD-none	algs-none	OLD-none
algs	no	yes	yes	algs	algs-none			algs-none	algs-none
algs	yes	no	no	algs	OLD-IKE	algs-IKE	OLD-IKE	algs-IKE	OLD-OLD
algs	yes	no	yes	algs	algs-IKE			algs-IKE	algs-OLD
algs	yes	yes	no	algs	OLD-IKE, OLD-none	algs-IKE algs-none	OLD-IKE algs-none	algs-IKE, algs-none	OLD-OLD, OLD-none
algs	yes	yes	yes	algs	algs-IKE, algs-none			algs-IKE, algs-none	algs-OLD, algs-none
algs-dh	no	no	no						
algs-dh	no	no	yes						
algs-dh	no	yes	no						
algs-dh	no	yes	yes						
algs-dh	yes	no	no	algs	algs-dh -or- OLD-IKE	algs-dh	OLD-IKE	algs-dh	OLD-OLD #1347
algs-dh	yes	no	yes	algs	algs-dh -or- algs-dh OLD-IKE			algs-dh	algs-dh
algs-dh	yes	yes	no	algs	algs-dh, algs-none -or- OLD-IKE, OLD-none	algs-dh, algs-none	OLD-IKE, OLD-none	algs-dh, algs-none	OLD-OLD, OLD-none
algs-dh	yes	yes	yes	algs	algs-dh, algs-none -or- algs-dh OLD-IKE algs-none			algs-dh, algs-none	algs-dh, algs-none

# Possible solutions

- Disallow IKE DH != Child DH
  - does not fix install base, but will reduce problem over time.
  - Authors of RFC8247 already tried to suggest this to WG at the time :-)
- On responder, if Initial Exchange IKE DH != Child, return NO\_PROPOSAL\_CHOSEN
  - and use the Childless IKE SA to CREATE\_CHILD\_SA Child SA with proper DH
    - Causes race conditions and/or interoperability issues
- In Initial Exchanges, add DH to proposals if pfs=yes (will prob break things)
- On responder, do IKE rekey if Child DH insufficient (doesn't help initiator case)
- In IKE\_AUTH, exchange a new CHILD\_SA\_KEY(dh,..) notify
  - Return INVALID\_KEY / NO\_PROPOSAL\_CHOSEN if new notify mismatched

# Questions for the IPsecME WG

- Q1: Is a new Notify CHILD\_SA\_KEY(dh,...) worth publishing ?
- Q2: Is a new Notify Error code useful ?
- Q2: Is it useful to write up a “DH behaviour updates” doc, updating RFC 7296 ?
- Q3: Did we miss additional issues or other obvious solutions ?