Group Domain of Interpretation (GDOI)

<draft-ietf-msec-gdoi-02.txt>

Mark Baugher (Cisco)
Thomas Hardjono (Verisign)
Hugh Harney (SPARTA)
Brian Weis (Cisco)

Group DOI

Changes from draft -01

Implementation Status

Future Plans

Changes to draft -01

- Section 2.4.2
  - Text is added stating that the GDOI prototol SHOULD NOT use port 500.
  - Is that strong enough?

Changes to draft -01

- Section 3.2
  - Resurrected the optional KE payloads. They may be used to further protect the keys sent in the KD payload.
  - Re-defined the SEQ payload to be optional in the registration message, and added text saying it is only required when the group policy includes policy rekey message policy.

Changes to draft -01

- Sections 3.3 & 3.4
  - Added text clarifying the initiator and responder operations for the registration message.
  - Helps in understanding the system flow.

Changes to draft -01

- Section 5.1
  - Included the ID payload by copy rather than by reference.
- Sections 5.3 & 5.4
  - Define IPSEC and ISAKMP assigned numbers by referencing the IANA registries rather than the RFCs

Changes to draft -01

- Section 5.4.1
  - Tweaked the fields of the ESP TEK payload based on implementation experience.
  - Corrected a cut-n-paste error and added some explanatory text

Changes to draft -01

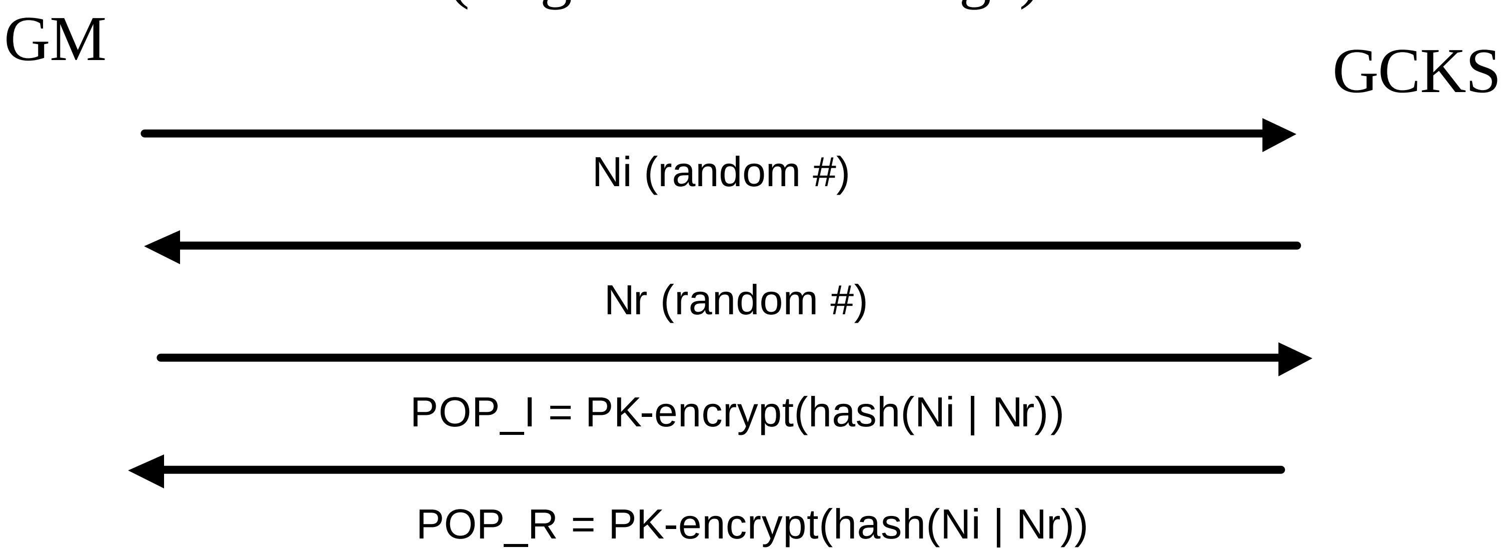
- Section 8.0
  - Added an IANA Considerations section.
    - New DOI number needed
    - New payloads need assigned numbers
    - New GDOI registry needs to be formed

Changes to draft -01

- Remedied attacks on the rekey message discovered by Catherine Meadows
  - Man-in-the-middle
    - Assumes IKE Phase 1 keys protecting the GDOI exchange can be broken in real time.
  - Dishonest group member

Use of nonces in GDOI

(Registration Message)



GM

GCKS

Ni (random #)

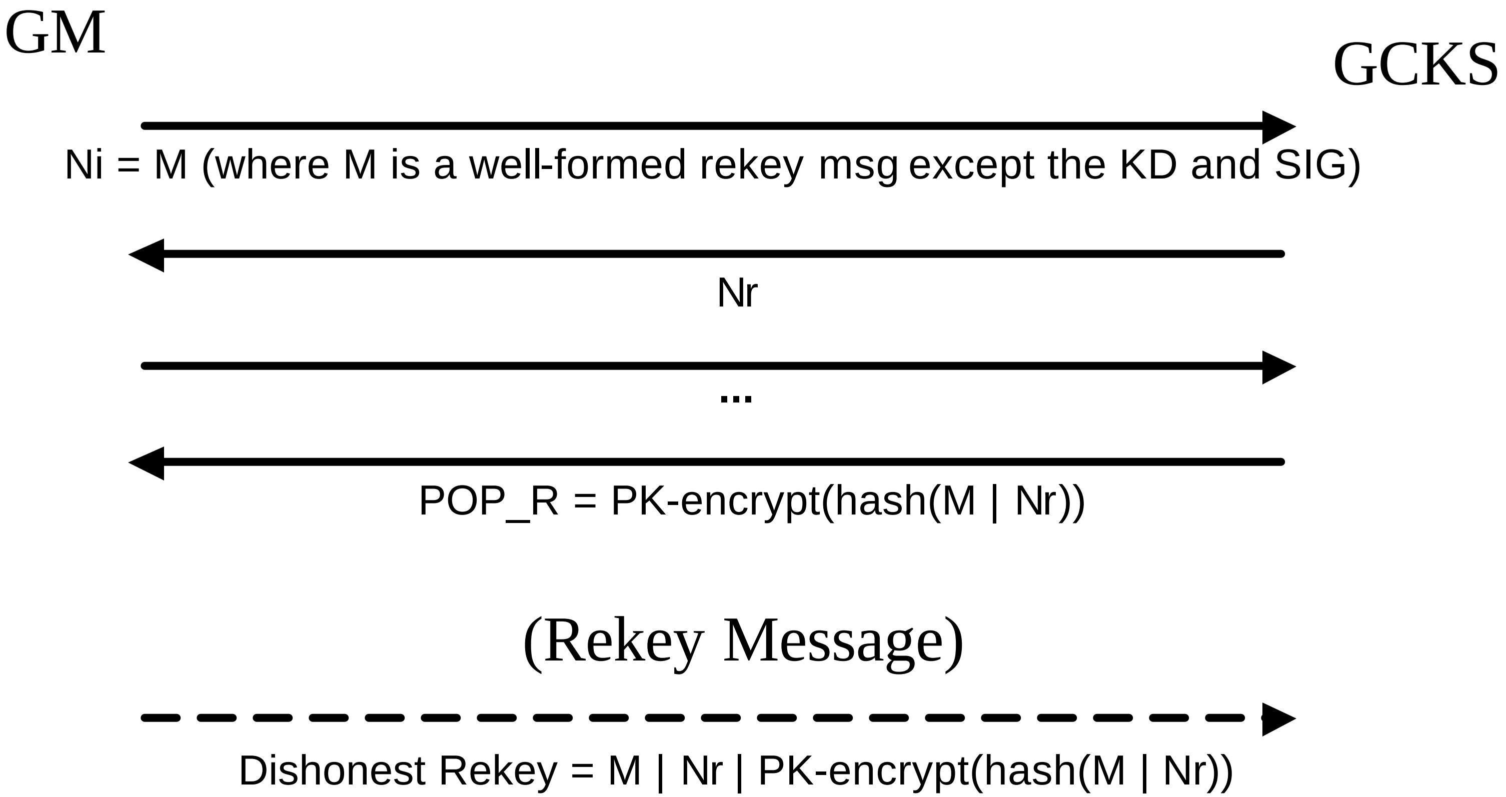
Nr (random #)

POP_I = PK-encrypt(hash(Ni | Nr))

POP_R = PK-encrypt(hash(Ni | Nr))

- Liveliness indication
- Proof of Possession encrypted content

Dishonest Group Member Attack



GM

GCKS

Ni = M (where M is a well-formed rekey msg except the KD and SIG)

Nr

...

POP_R = PK-encrypt(hash(M | Nr))

(Rekey Message)

Dishonest Rekey = M | Nr | PK-encrypt(hash(M | Nr))

This only happens if …

- Ni is the size of the rekey message
- GCKS uses same keypair for
  - encrypting the POP payload, and
  - signing a rekey message
- The POP hash algorithm is the same as the SIG hash algorithm

It was mitigated by ...

- Recommending that the GCKS:
  - SHOULD NOT use the same key for encrypting the POP as signing the rekey.
- Bounding the nonce to be between 8 and 128 bytes.
  - A rekey message is calculated to be larger than 128 bytes

Group DOI

Changes from draft -01

Implementation Status

Future Plans

Implementation Status: Interoperability

- Two implementations:
  - Nortel implementation based on FreeSWAN (Linux)
  - Cisco implementation based on isakmpd (Linux and OpenBSD)
- Successful interop of registration protocol this week!

Implementation Status: Creation of IPSec SAs

Using the isakmpd implementation:

- The client received multicast SAs from an isakmpd-based GDOI key server and loaded them into the OpenBSD kernel.
- IPSec SAs were created (OpenBSD)
- Multicast packets matching the SAs were encrypted by one host and decrypted by another.

Implementation Status: Creation of SRTP SAs

Again using the isakmpd implementation:

- The client received multicast SAs from an isakmpd-based GDOI key server on behalf of an SRTP application.
- The application created SRTP SAs (Linux and OpenBSD)
- Multicast packets matching the SAs were correctly processed by the SRTP code.

Group DOI

Changes from draft -01

Implementation Status

Future Plans

Working Group Last Call

We think we're ready:

- This is the 5th version of the draft.
- Catherine Meadows performed a formal security analysis and all issues have been resolved to her satisfaction.
- We have two implementations which interoperate
- Testing has shown that GDOI can accurately create both IPSec and SRTP SAs.

Next work items

- Clarify the text with MUST, SHOULD, MAY keywords per RFC 2119.
- Re-format for easier conversion from Word to text.
- Check the document against the AD nits list, etc.

Reference implementation

- The isakmpd based implementation will be released as a reference implementation early next year.

BACKUP SLIDES

Message 1: Request

```
Initiator (Member)                Responder (GCKS)
------------------                ----------------
HDR*, HASH(1), Ni, ID     -->

* Protected by IKE Phase 1 SA Hashes, encryption occurs after HDR
```

```
  HASH(1) = prf(SKEYID_a, M-ID | Ni | ID)
```

- HASH provides message authentication
- NONCE is used for replay protection
- ID indicates the desired group to join

Message 2: Policy Push

```
Initiator (Member)                     Responder (GCKS)
--------------------                   ------------------
                              <--      HDR*, HASH(2), Nr, SA
```

```
  HASH(2) = prf(SKEYID_a, M-ID | Ni_b | Nr | SA)
```

- SA contains specific policy for the Category-2 and Category-3 SAs. E.g., which crypto algorithms to use.

Message 3: Ack

```
Initiator (Member)                Responder (GCKS)
------------------                ----------------
HDR*, HASH(3) [, KE_I]    -->

         [,CERT] [,POP_I]
```

```
  HASH(3) = prf(SKEYID_a, M-ID | Ni_b | Nr_b [ | KE_I ] [| POP_I])
```

- KE_I obtains perfect forward secrecy (if desired)
- CERT send a public key used for authorization (if needed for POP_I)
- POP_I provides evidence that the client has possession of a private or secret key

Message 4: Key Download

```
Initiator (Member)                    Responder (GCKS)
--------------------                  -----------------
                               <--     HDR*, HASH(4), [KE_R,] SEQ, KD

                                                    [,CERT] [,POP_R]
```

```
  HASH(4) = prf(SKEYID_a, M-ID | Ni_b | Nr_b [ | KE_R ] | SEQ | KD [| POP_R])
```

- SEQ provides the sequence number which will be used for the next rekey message.
- KD provides the keys for the policy delivered in the SA payload

Registration Protocol

```
Initiator (Member)                Responder (GCKS)
------------------                ----------------

HDR*, HASH(1), Ni, ID     -->
                          <--     HDR*, HASH(2), Nr, SA
HDR*, HASH(3) [, KE_I]    -->
         [,CERT] [,POP_I]
                          <--     HDR*, HASH(4), [KE_R,] SEQ, KD
                                                   [,CERT] [,POP_R]
* Protected by IKE Phase 1 SA Hashes, encryption occurs after HDR
```

Rekey Message

```
Member                                   GCKS or Delegate
-------                                  ------------------
            <----   HDR*, SEQ, SA, KD, [CERT,] SIG
```

* Protected by (current) KEK after HDR

** SIG is over entire message including HDR, excluding SIG

- The “cookie pair” in the ISAKMP HDR acts as a SPI which identifies the group.
- SEQ contains a counter used for replay protection
- SIG contains a digital signature of the packet for authentication