Protocolos de enrutamiento anónimos para redes Ad-Hoc

1ª Parte

María Mercedes Rodríguez García

DISEÑO AVANZADO DE REDES

MÁSTER EN INVESTIGACIÓN EN INGENIERÍA DE SISTEMAS Y DE LA COMPUTACIÓN

ANODR

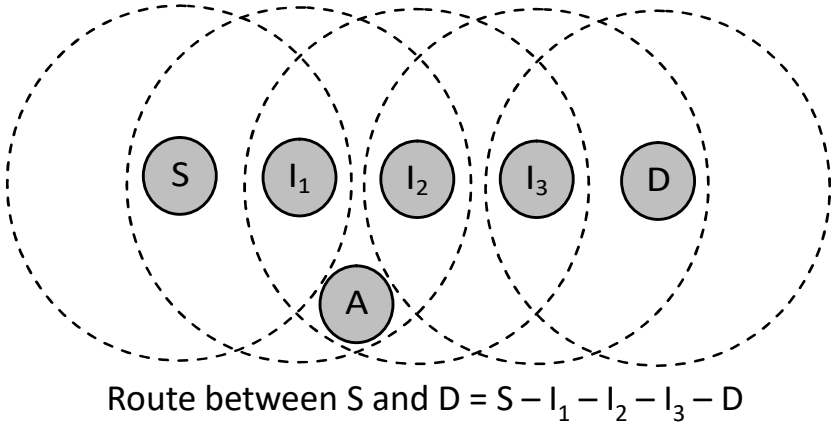
(ANonymous On Demand Routing
with Untraceable Routes for Mobile Ad-hoc Networks)

2003

Los autores proponenen varias versiones del protocolo:

- Anonymous-only ANODR-PO (Section 3.2)
- Anonymous-only ANODR-BO (Section 3.2)
- **Anonymous-only ANODR-TBO (Section 3.2)**
- Anonymous+Untraceable ANODR-PO (Section 3.3)
- Anonymous+Untraceable ANODR-BO (Section 3.3)
- Anonymous+Untraceable ANODR-TBO (Section 3.3)

ANODR



Fase **RREQ**:

- Fase iniciada por el nodo origen S
- Broadcast (con inundación)
- <RREQ, seqnum, $trapdoor_D$, $onion_S$>
  - seqnum: número de secuencia global único
  - $trapdoor_D = E_{K_{S-D}}(ID_D, \mathbf{N_S})$
    - K_{S-D}: clave secreta compartida entre nodos S y D
    - ID_D: identidad del nodo destino
    - N_S: nonce del nodo origen que se utilizará en la fase RREP como $proof_D$
  - $onion_S = E_{K_S}(ID_S)$

Conceptos criptográficos

Nonce:

- Número aleatorio de un solo uso.

Trapdoor:

- Trampa o función de una vía $f : X \rightarrow Y$
- Sin la clave secreta es computacionalmente imposible obtener X a partir de Y.
- P.e. $trapdoor_D$ solo puede ser abierto por D porque es este nodo el que posee la clave secreta K_{S-D}
- ver Section 2.5 del artículo de ANODR.

Conceptos criptográficos

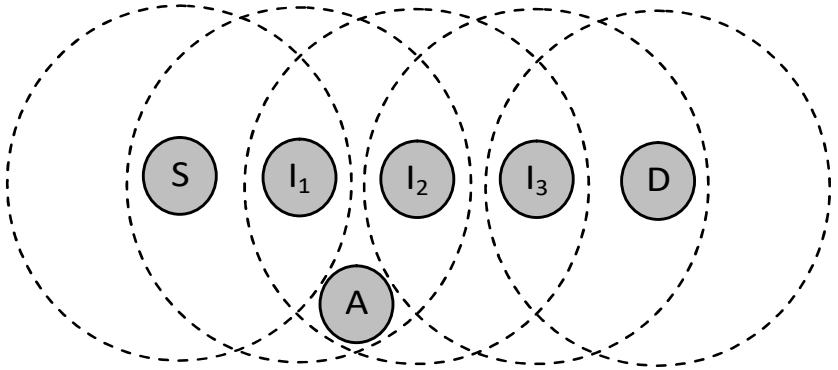
Onion:

- Estructura multicapa encriptada.
- Cada nodo de la ruta añade su capa encriptada.

Número de secuencia global único:

- ver nota al pie en Section 3.2 del artículo de ANODR

ANODR



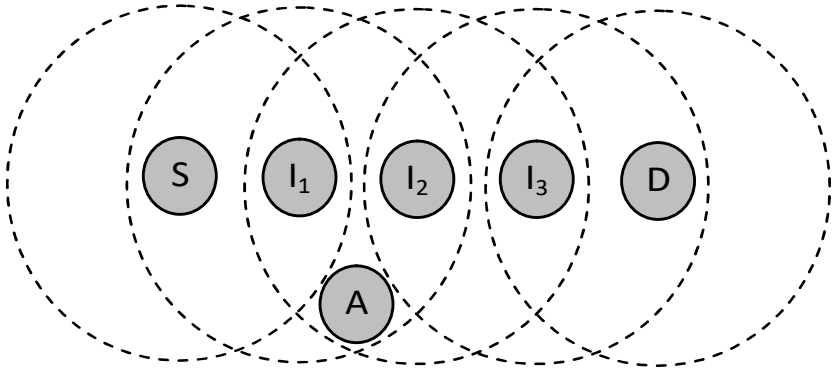
Route between S and D = S – I_1 – I_2 – I_3 – D

¿Qué hace un nodo cuando recibe un mensaje RREQ?
Por ejemplo, ¿qué haría el nodo I_2? <RREQ, seqnum, $trapdoor_D$, $onion_{I1}$>
$trapdoor_D = E_{K_{S-D}}(ID_D, N_S)$

1. Verificar si es la primera vez que recibe ese RREQ ¿cómo?
2. Verificar si es el destino ¿cómo? Intentado abrir el $trapdoor_D$
3. Si I_2 no es el destino,
    1. I_2 genera un nonce $\mathbf{N_{I_2}}$ que utilizará como prueba en RREP.
    2. I_2 añade su capa onion al mensaje RREQ antes de difundirlo.

    <RREQ, seqnum, $trapdoor_D$, $onion_{I2}$>
    $onion_{I2} = \mathbf{E_{K_{I_2}}(N_{I_2},}\ E_{K_{I1}}(N_{I1}, E_{K_S}(ID_S))\mathbf{)}$

ANODR

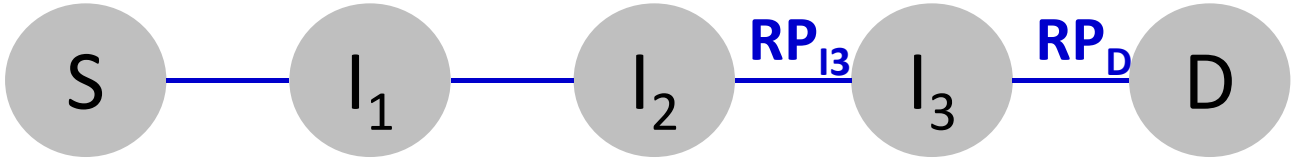


Route between S and D = S – I_1 – I_2 – I_3 – D

Fase **RREP**:

- Fase iniciada por el nodo destino D
- Broadcast (sin inundación)
- <RREP, **RP_D**, $proof_D$, $onion_{I3}$>
  - RP_D o $RoutePseudonym_D$: pseudónimo de ruta local. Es un número aleatorio generado por D y publicado en el salto D-I_3
  - $proof_D$ = **N_S** prueba de haber abierto el trapdoor
  - $onion_{I3} = E_{K_{I3}} (N_{I3}, E_{K_{I2}} (N_{I2}, E_{K_{I1}} (N_{I1}, E_{K_S} (ID_S))))$

ANODR



¿Qué hace un nodo cuando recibe un mensaje RREP?
Por ejemplo, ¿qué haría el nodo I_3? <RREP, $\mathbf{RP_D}$, $proof_D$, $onion_{I3}$>

1. Verificar si es un nodo de ruta ¿cómo? Intentado abrir su capa onion $E_{K_{I3}}(\mathbf{N_{I3}}, E_{K_{I2}}(N_{I2}, E_{K_{I1}}(N_{I1}, E_{K_S}(ID_S))))$ y comprobando si el nonce desencriptado $\mathbf{N_{I3}}$ coincide con el almacenado.
2. Si $I_3 \notin$ ruta, descartar el mensaje RREP.
3. Si $I_3 \in$ ruta,
   a. Registrar RP_D en su tabla de rutas:

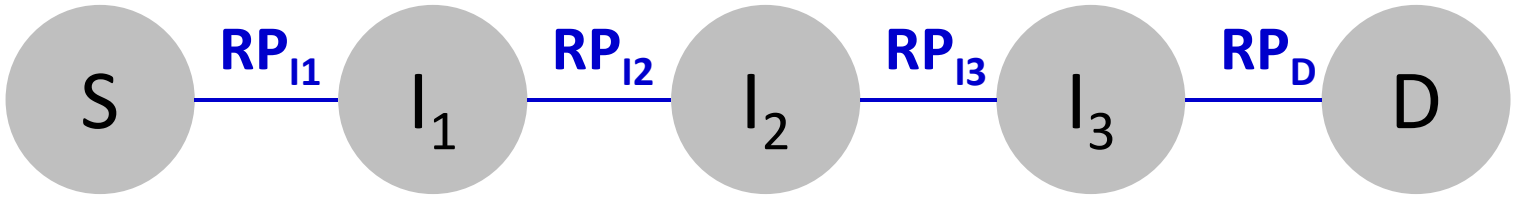
	RP_D

   b. Generar RP_{I3} y registrarlo en la tabla de rutas:

RP_{I3}	RP_D

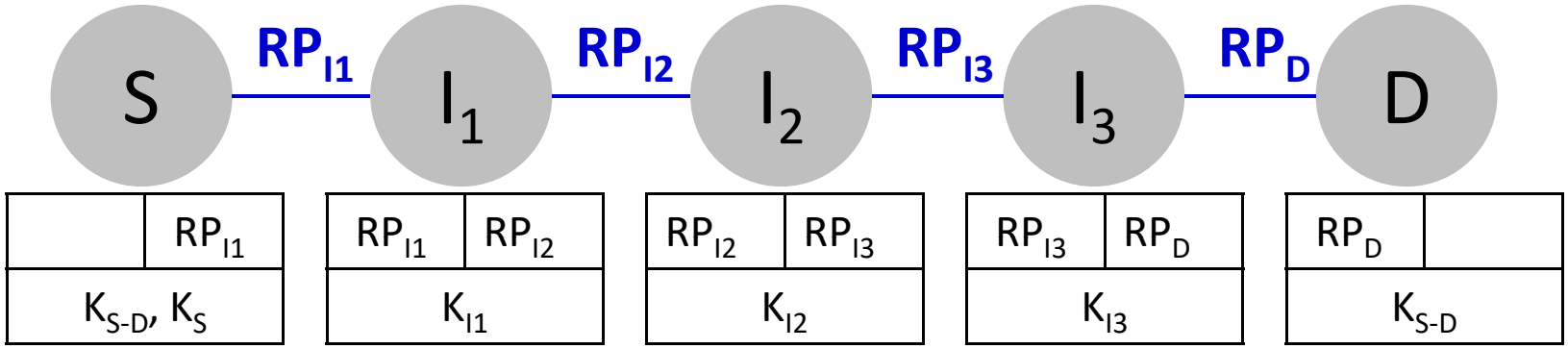
   c. Modificar el mensaje RREP antes de difundirlo.
   <RREP, $\mathbf{RP_{I3}}$, $proof_D$, $onion_{I2}$> $onion_{I2} = E_{K_{I2}}(N_{I2}, E_{K_{I1}}(N_{I1}, E_{K_S}(ID_S)))$

ANODR



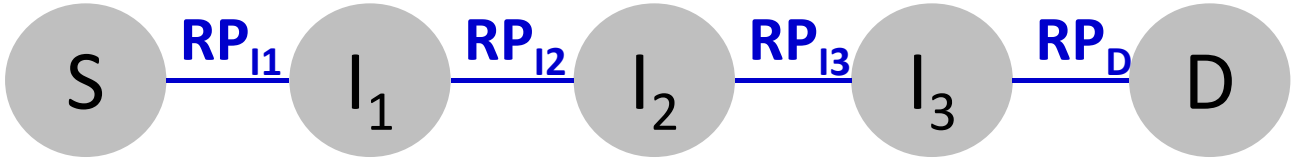
$RP_X = RoutePseudonym_X$

ANODR



$RP_X = RoutePseudonym_X$

ANODR



Fase **DATA**:

- Fase iniciada por el nodo origen S

- Bidireccional: S→D, D→S

- Broadcast (sin inundación)

- <DATA, RP_{I1}, mensaje encriptado>

¿Qué hace un nodo cuando recibe un mensaje DATA?
Por ejemplo, ¿qué haría el nodo I_1?

1. Si $I_1 \notin$ ruta, descartar el mensaje DATA.
2. Si $I_1 \in$ ruta (tiene registrado RP_{I1} en su tabla de rutas), reenviar el mensaje DATA al siguiente RP.

   <DATA, RP_{I2}, mensaje encriptado>

ANODR

Scalability:

Type of operation	Operations
Symmetric key operations	- encryption - decryption
Efficient public key operations (cuando se utiliza la clave pública)	- encryption - verification of signature
Complexity public key operations (cuando se utiliza la clave privada)	- decryption - signature

Crítico: nº de operaciones que realizan los nodo intermedios

ANODR

Operaciones criptográficas realizadas por los nodos intermedios en la fase **RREQ**:

Broadcast?	Sí
Inundación?	Sí

Type of operation	Nº Operations	Operation
Symmetric key operations	2? n+1?	- Verificar el trapdoor - Añadir capa onion
Efficient public key operations	0	
Complexity public key operations	0	

ANODR

Fase **RREP**:

Broadcast?	Sí
Inundación?	No

Operaciones criptográficas realizadas por los nodos intermedios pertenecientes a la ruta:

Type of operation	Nº Operations	Operation
Symmetric key operations	1? n?	- Abrir capa onion
Efficient public key operations	0	
Complexity public key operations	0	

Operaciones criptográficas realizadas por los nodos vecinos no pertenecientes a la ruta:

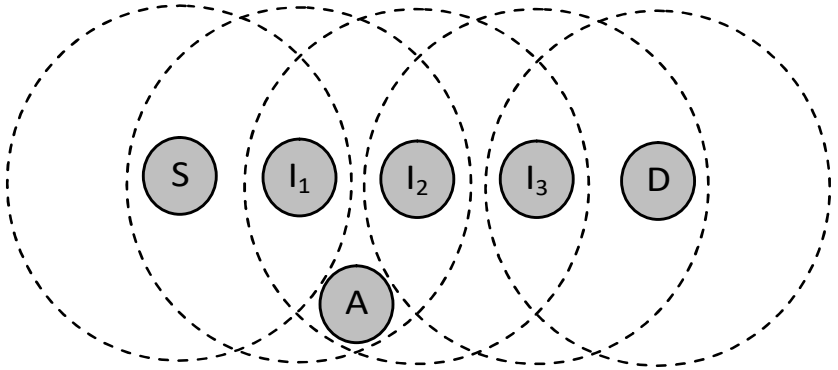
Type of operation	Nº Operations	Operation
Symmetric key operations	1? n?	- Intentar abrir capa onion
Efficient public key operations	0	
Complexity public key operations	0	

AnonDSR

(efficient ANONymous Dynamic Source Routing
for Mobile Ad-hoc Networks)

2005

AnonDSR



Route between S and D = S – I_1 – I_2 – I_3 – D

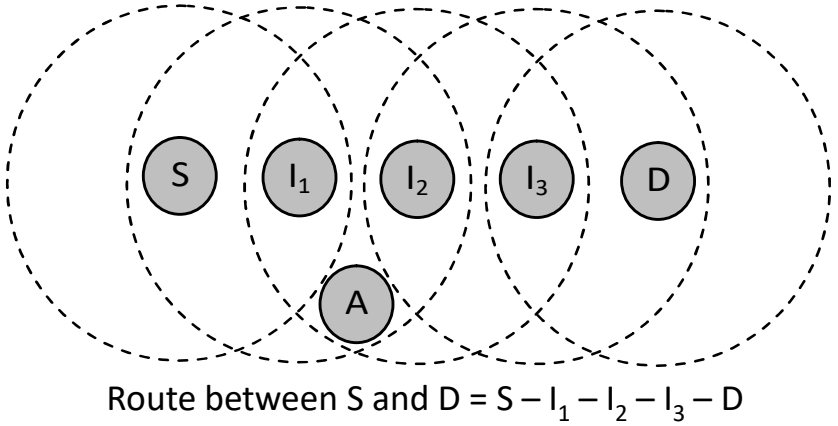
Fase **SPE**:

- Fase iniciada por el nodo origen S
- Broadcast (con inundación)
- Utilizada para intercambiar parámetros de seguridad con el nodo D:
  - K_{S-D} : clave secreta.
  - N_{S-D}: índice de la clave secreta.

  } almacenados en el anillo de claves de S y D

- La subfase RREQ de la fase SPE presenta un problema: investigar sección 3.1 del artículo.

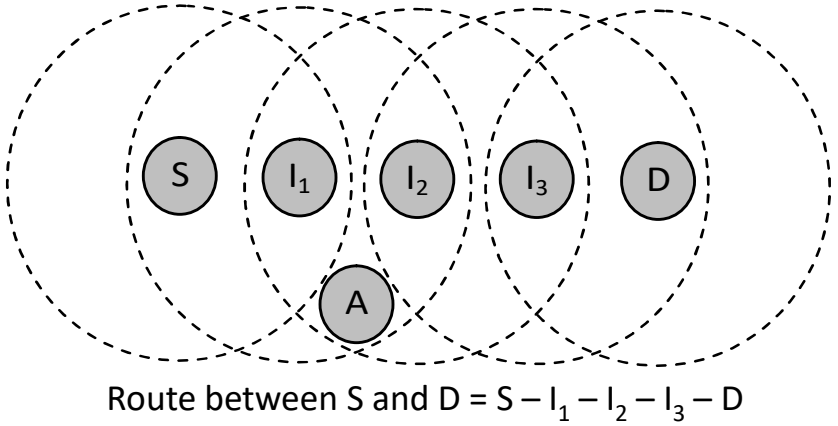
AnonDSR



Fase **RREQ**:

- Fase iniciada por el nodo origen S
- Broadcast (con inundación)
- <RREQ, PK_{temp}, $trapdoor_D$, $onion_S$>
  - PK_{temp} : clave pública de un solo uso creada por S. También utilizada como número de secuencia
  - $trapdoor_D = \{N_{S-D}, E_{K_{S-D}}(ID_D, SK_{temp})\}$
    - K_{S-D}: clave secreta compartida entre nodos S y D
    - ID_D: identidad del nodo destino
    - SK_{temp} : clave privada correspondiente a PK_{temp}
  - $onion_S = \{E_{PK_{temp}}(\mathbf{K_S}), E_{\mathbf{K_S}}(RP_S, ID_S, \ldots)\}$
    - K_S: clave de sesión que solo compartirá con D

AnonDSR



¿Qué hace un nodo cuando recibe un mensaje RREQ?
Por ejemplo, ¿qué haría el nodo I_2?

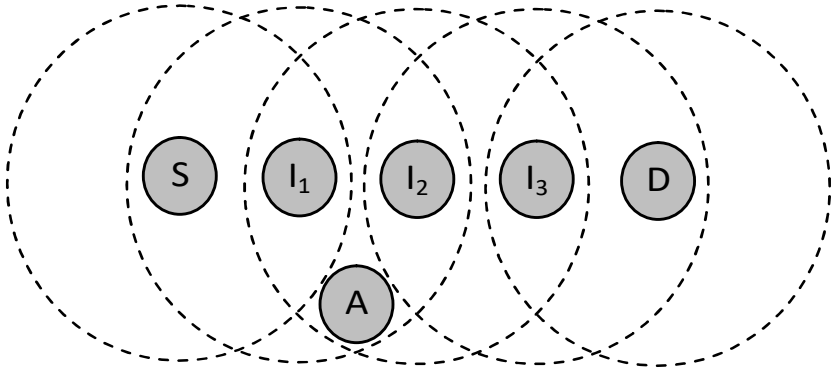
1. Verificar si es la primera vez que recibe ese RREQ
2. Verificar si es el destino ¿cómo? Intentado abrir el $trapdoor_D$
3. Si I_2 no es el destino, añade su capa onion en el mensaje RREQ antes de difundirlo.

   <RREQ, PK_{temp}, $trapdoor_D$,$onion_{I2}$>

   $onion_{I2} = \{E_{PK_{temp}}(\mathbf{K_{I2}}), E_{\mathbf{K_{I2}}}(RP_{I2}, onion_{i1})\}$

   K_{I2}: clave de sesión que solo compartirá con S y D

AnonDSR

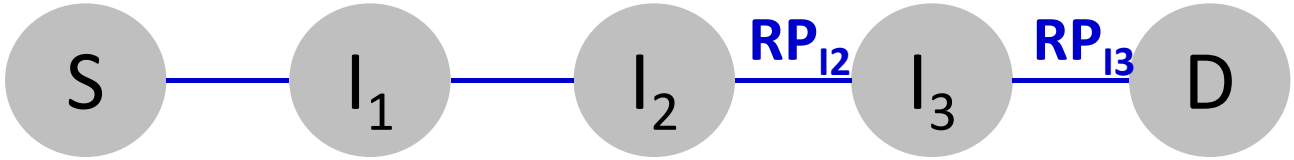


Route between S and D = S – I_1 – I_2 – I_3 – D

Fase **RREP**:

- Fase iniciada por el nodo destino D
- Broadcast (sin inundación).
- Usando SK_{temp} desencripta todo el onion.
- Registra K_{I3}, K_{I2}, K_{I1}, K_S, RP_{I3}, RP_{I2}, RP_{I1}, RP_S en su tabla de rutas
- <RREP, RP_{I3}, $onion_{I3}$>
  - $onion_{I3} = E_{K_{I3}} (RP_{I2}, E_{K_{I2}} (RP_{I1}, E_{K_{I1}} (RP_S, E_{K_S} (K_{I3}, K_{I2}, K_{I1}, RP_{I3}, RP_{I2}, RP_{I1}, \ldots))))$
  - Observar que el onion en RREP es diferente al onion RREQ.

AnonDSR



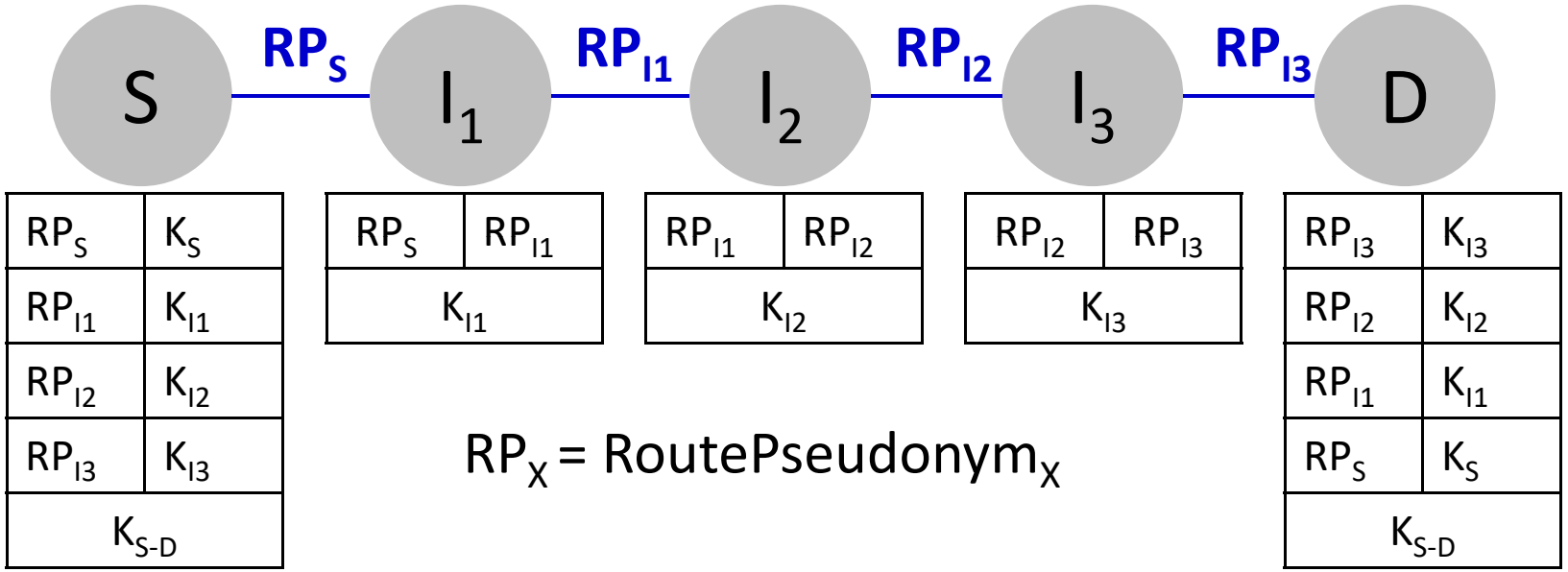
¿Qué hace un nodo cuando recibe un mensaje RREP?
Por ejemplo, ¿qué haría el nodo I_3?

1. Verificar si es un nodo de ruta ¿cómo? verificando si RP_{I3} es su pseudónimo.
2. Si $I_3 \notin$ ruta, descartar el mensaje RREP.
3. Si $I_3 \in$ ruta,
   a. Abrir su capa onion $E_{K_{I3}}(RP_{I2}, E_{K_{I2}}(RP_{I1}, E_{K_{I1}}(RP_S, E_{K_S}(K_{I3}, K_{I2}, K_{I1}, RP_{I3}, RP_{I2}, RP_{I1}, \ldots))))$
   b. Registrar en su tabla de rutas los caminos locales:

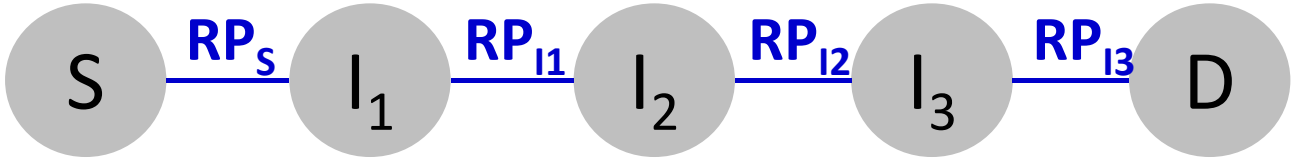
RP_{I2}	RP_{I3}

   c. Modificar el mensaje RREP antes de difundirlo.
   $<RREP, RP_{I2}, onion_{I2}>$ $onion_{I2} = E_{K_{I2}}(RP_{I1}, E_{K_{I1}}(RP_S, E_{K_S}(K_{I3}, K_{I2}, K_{I1}, RP_{I3}, RP_{I2}, RP_{I1}, \ldots)))$

AnonDSR


S
RP_S
I_1
RP_{I1}
I_2
RP_{I2}
I_3
RP_{I3}
D
RP_S K_S
RP_{I1} K_{I1}
RP_{I2} K_{I2}
RP_{I3} K_{I3}
K_{S-D}
RP_S RP_{I1}
K_{I1}
RP_{I1} RP_{I2}
K_{I2}
RP_{I2} RP_{I3}
K_{I3}
RP_{I3} K_{I3}
RP_{I2} K_{I2}
RP_{I1} K_{I1}
RP_S K_S
K_{S-D}
$RP_X = RoutePseudonym_X$

AnonDSR



Fase **DATA**:

- Fase iniciada por el nodo origen S

- Bidireccional: S→D, D→S

- Broadcast (sin inundación)

- <DATA, RP_S, onion-data$_S$> $onion_S = E_{K_{I1}}(E_{K_{I2}}(E_{K_{I3}}(E_{K_S}(data))))$

¿Qué hace un nodo cuando recibe un mensaje DATA?
Por ejemplo, ¿qué haría el nodo I_1?

1. Si $I_1 \notin$ ruta, descartar el mensaje DATA.
2. Si $I_1 \in$ ruta (tiene registrado RP_S en su tabla de rutas),
   a. Modificar el mensaje DATA antes de difundirlo.

      <DATA, RP_{I1}, onion-data$_{I1}$> $onion_{I1} = E_{K_{I2}}(E_{K_{I3}}(E_{K_S}(data)))$

AnonDSR

Operaciones criptográficas realizadas por los nodos intermedios en la fase **RREQ**:

Broadcast?	Sí
Inundación?	Sí

Type of operation	Nº Operations	Operation
Symmetric key operations	2? 1?	- Verificar el trapdoor? - Añadir capa onion
Efficient public key operations	1	- Encriptar su clave secreta
Complexity public key operations	0	

AnonDSR

Fase **RREP**:

Broadcast?	Sí
Inundación?	No

Operaciones criptográficas realizadas por los nodos intermedios pertenecientes a la ruta:

Type of operation	Nº Operations	Operation
Symmetric key operations	1	- Abrir capa onion
Efficient public key operations	0	
Complexity public key operations	0	

Operaciones criptográficas realizadas por los nodos vecinos no pertenecientes a la ruta:

Type of operation	Nº Operations	Operation
Symmetric key operations	0	
Efficient public key operations	0	
Complexity public key operations	0	