A Signature Agility solution for CGA & SEND

draft-cheneau-csi-cga-pk-agility-00
draft-cheneau-csi-send-sig-agility-00
draft-cheneau-csi-ecc-sig-agility-00

76th IETF

-

CGA & SEND maIntenance WG

Tony Cheneau
email: tony.cheneau@it-sudparis.eu
Michaela Vanderveen
Maryline Laurent
Sean Shen

Overview

- Advantages of ECC/ECDSA
- Recommendations for hosts and routers
- Changes since the last version
  - New options
  - Negotiation process
- Authorization Delegation Discovery

Advantages of ECC/ECDSA

- Faster Public Key and signature generation
- Shorter Public Key size:
  - A 1024 bits long, DER encoded, RSA Public Key's size is 160 octets
  - Equivalent, DER encoded, ECC P-256 curve's size is ~88 octets
- Shorter Signature size:
  - PKCS#1 v1.5 signature when using 1024 bits long RSA Public Key is 128 octets
  - ECDSA signature is ~71 octets long

Recommendations for hosts

- Ability to generate CGA based on Multiple Keys (as a transitional mechanism, not mandatory)
- Ability to verify additional signature types besides the one used to generate a signature (not mandatory)
- Ability to communicate with RFC 3971 nodes (not mandatory)

Recommendations for routers

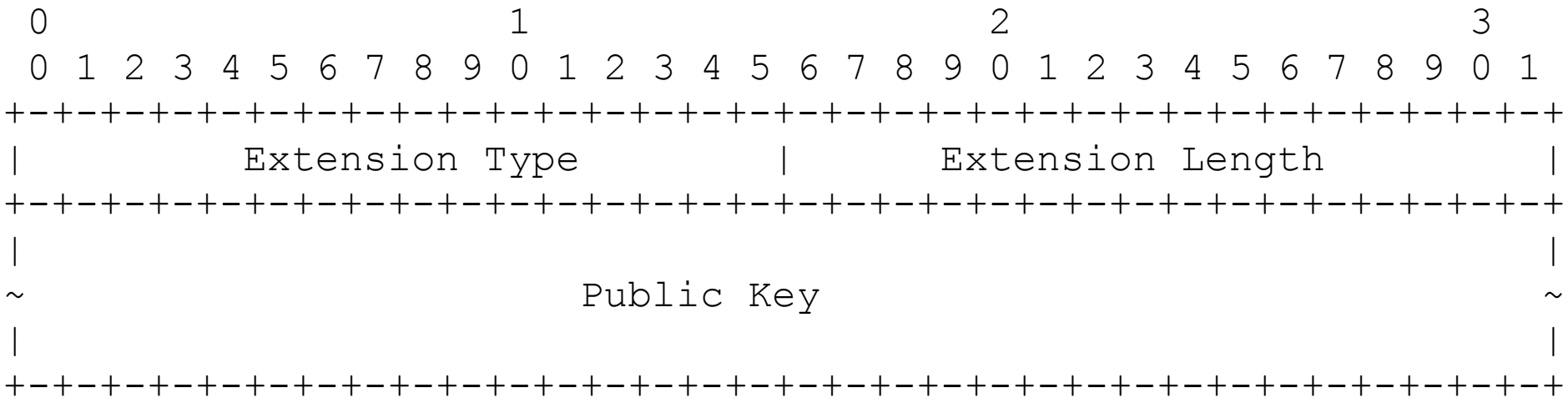
- Routers are likely to verify more Signature Algorithm types (even when they don't use M-CGA)
- When using one or many M-CGA, router can add multiple “Signature” options (for each Public Key available) to ND messages
- Routers should have their certificate(s) signed with the same algorithm they use to generate their (M-)CGA
- A single router may have multiple CGA of different types. It implies that the router will likely have multiple Certification Path (one for each addresses)

Changes since last version

- draft-cheneau-cga-pk-agility-01 → draft-cheneau-csi-cga-pk-agility-00
  - text clarification
- draft-cheneau-send-sig-agility-01 → draft-cheneau-csi-send-sig-agility-00
  - removes ECC/ECDSA related stuff
  - removes « resend » ('r') flag
  - removes « router as a notary » functionality
  - updated options
- draft-cheneau-csi-ecc-sig-agility-00
  - provides a basic skeleton for other signature algorithms

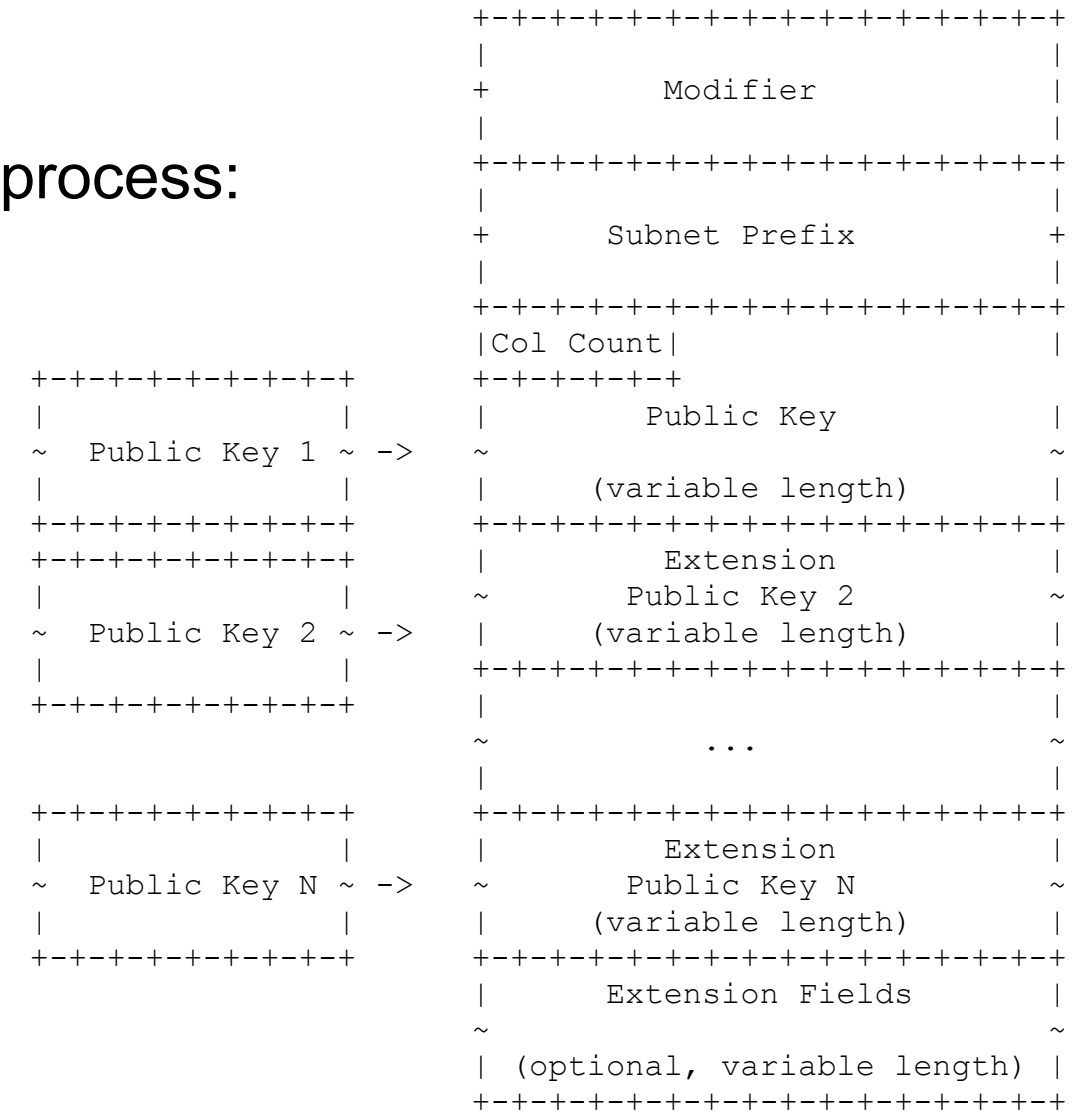
Multiple Key CGA (1 of 2)

CGA Parameters Data Structure's Public Key extension field:

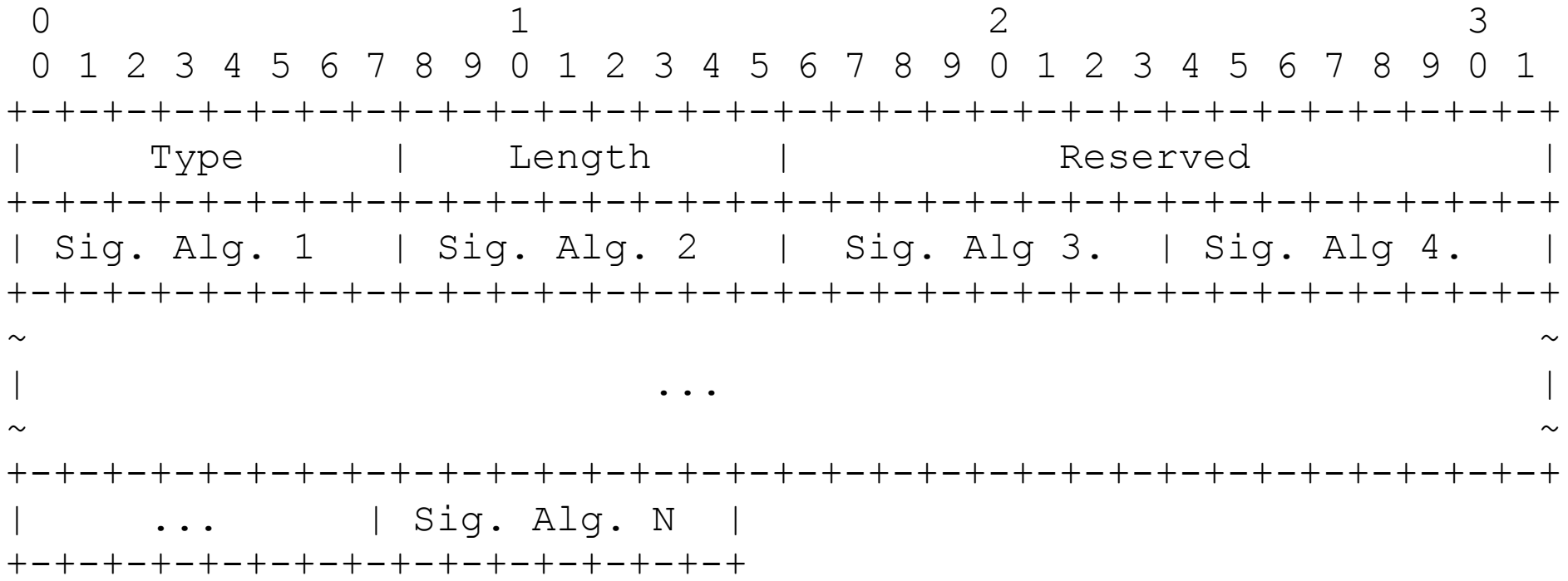


Multiple Key CGA(2 of 2)

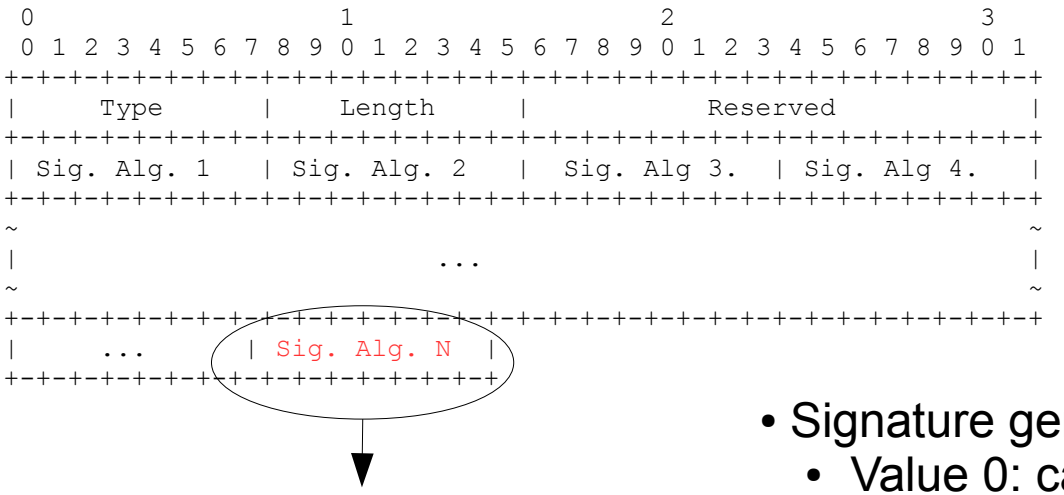
Multiple Key CGA generation process:

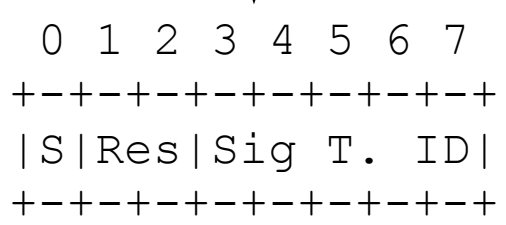


Supported Signature Algorithms Option (SSA)



Signature Algorithm field (detailed)

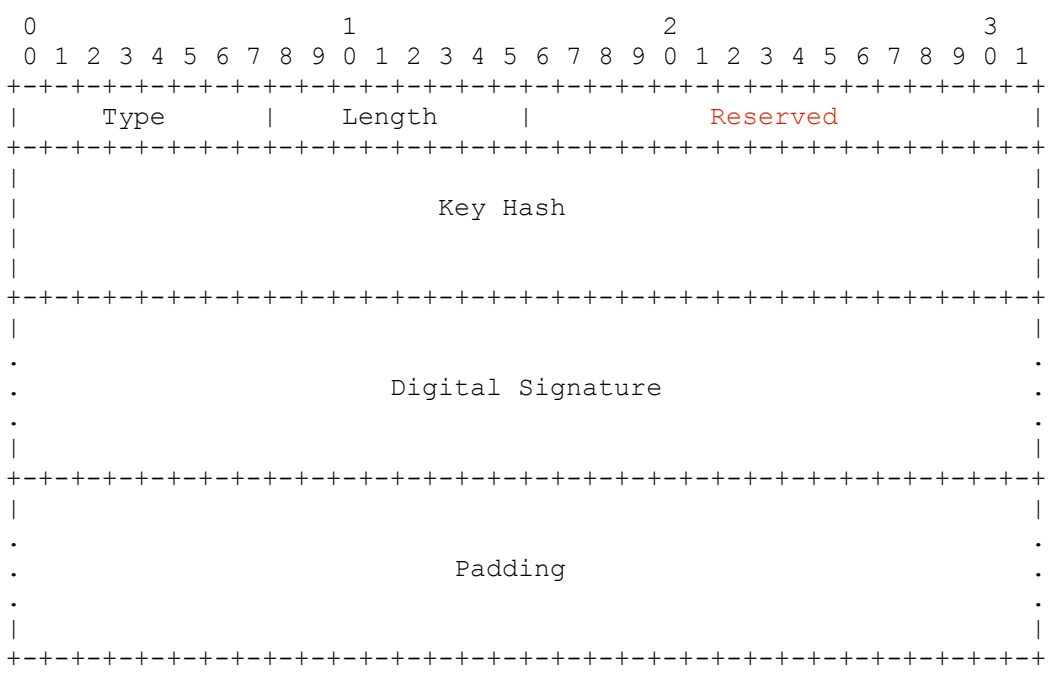




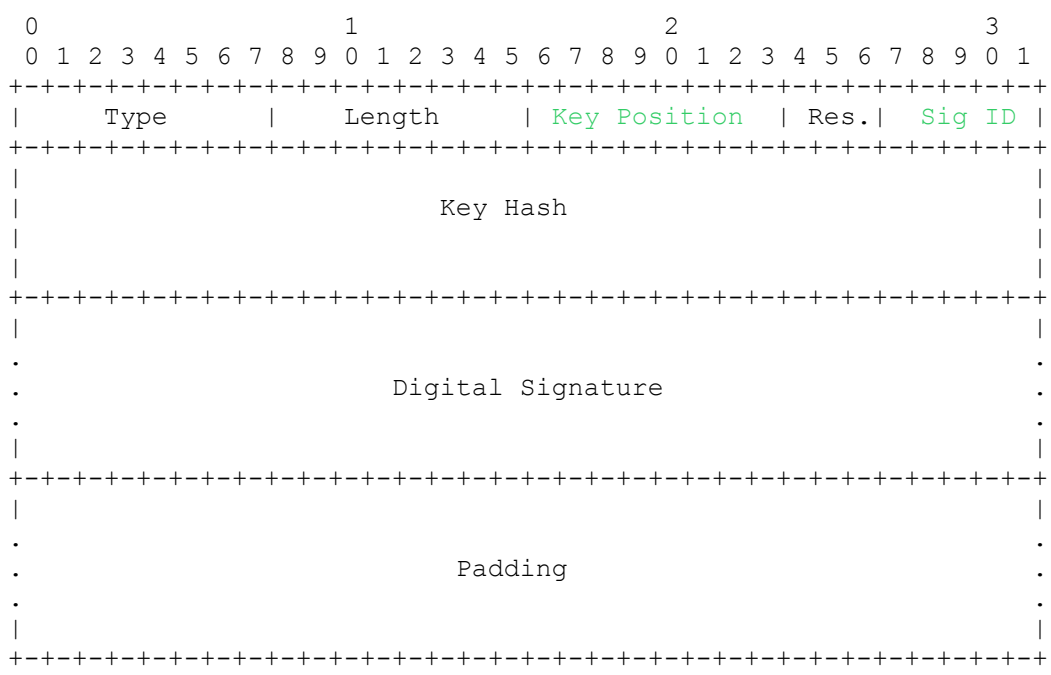
- Signature generation bit:
  - Value 0: can only verify this signature algorithm
  - Value 1: can sign and verify with this signature algorithm
- Signature Type Identifier subfield:
  - Value 0: RSA/SHA-1
  - Value 1: RSA/SHA-256
  - Value 9: ECDSA (P-256)/SHA-256
  - …

(ECC values follows registry IKEv2 authentication method)

Universal Signature Option (USO)



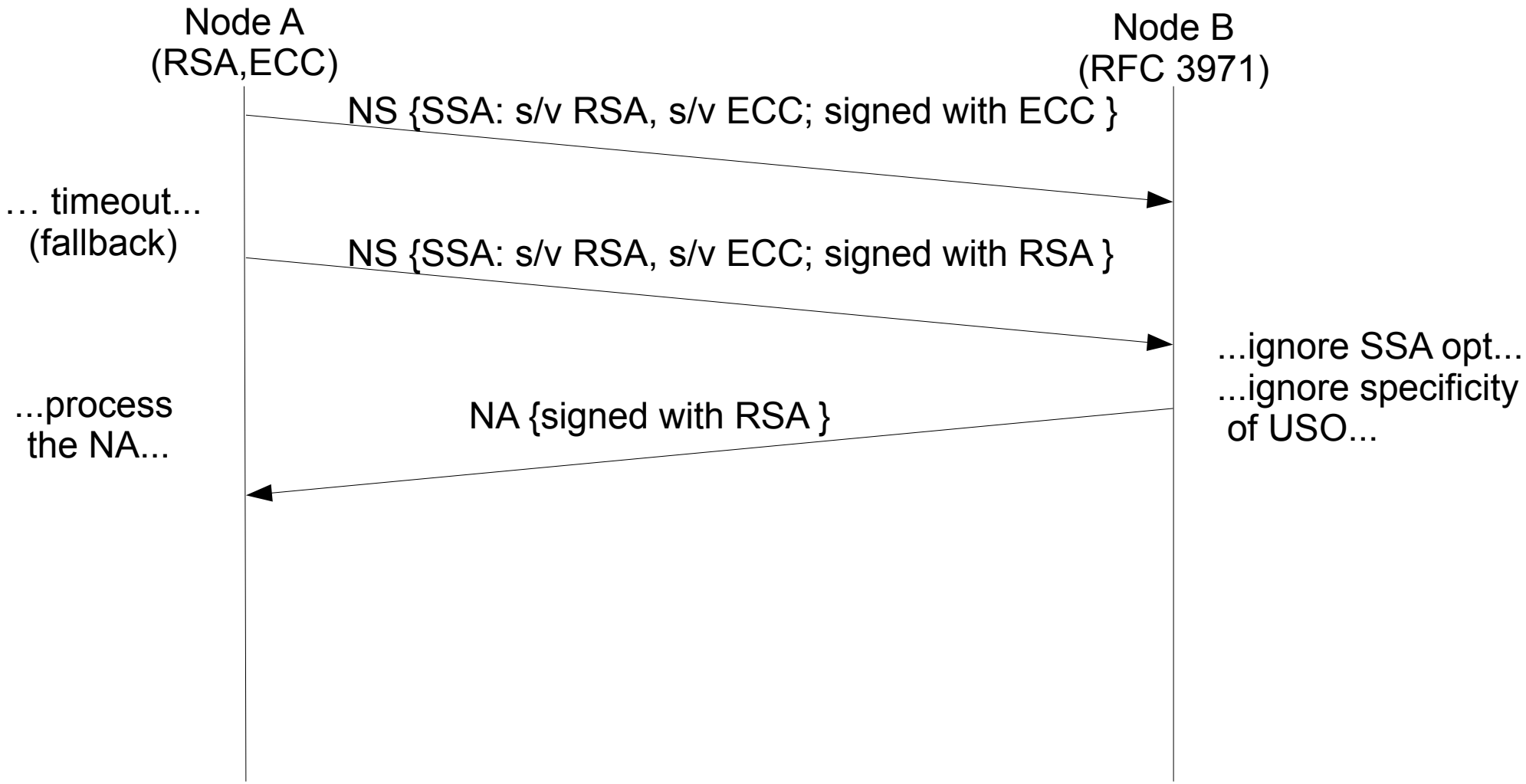
RSA Signature Option



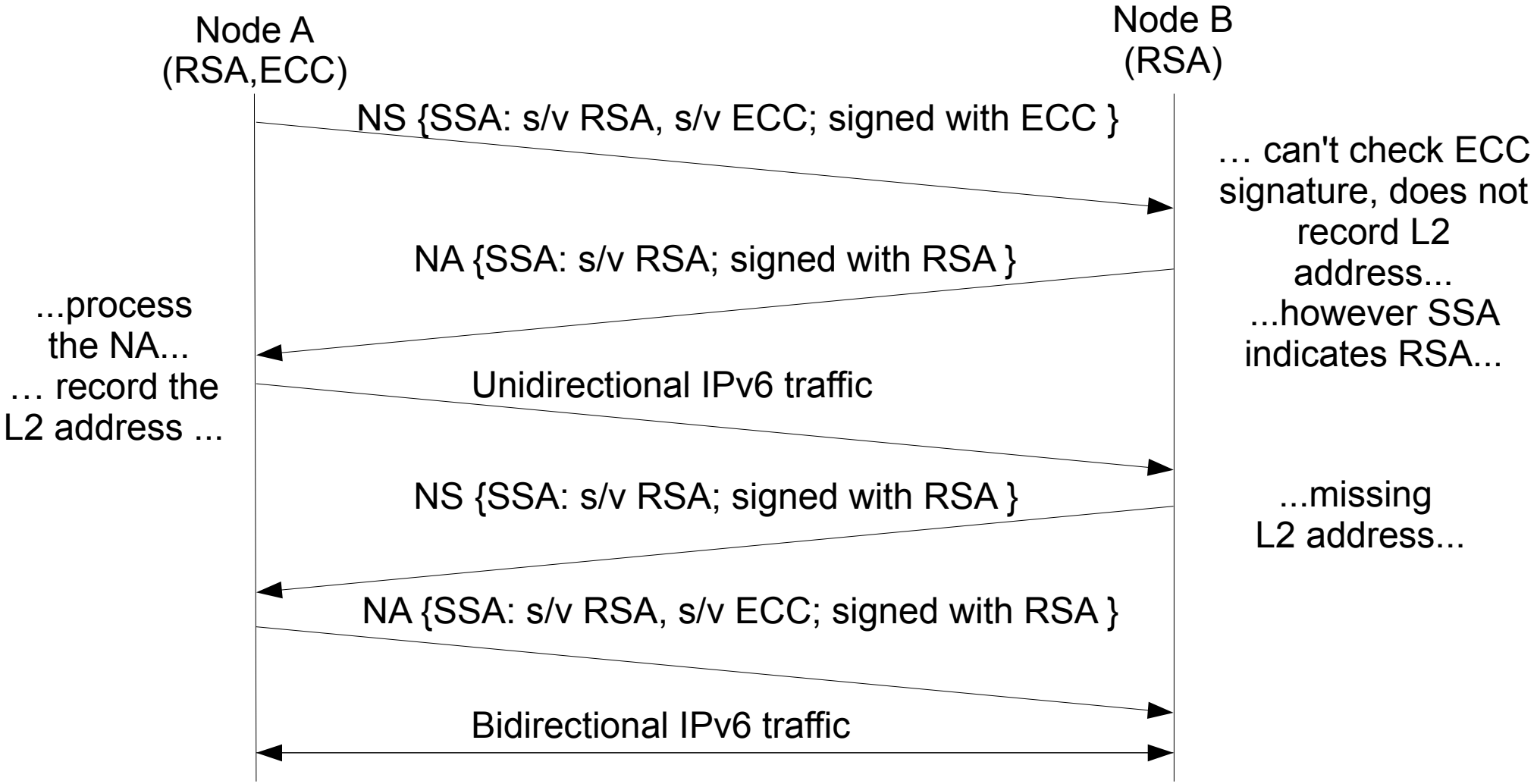
Universal Signature Option

Backward compatible with the RSA Signature Option when Key Pos=0 and Sig Id=0 (i.e. when using a single Public Key and RSA/SHA-1 signature algorithm)

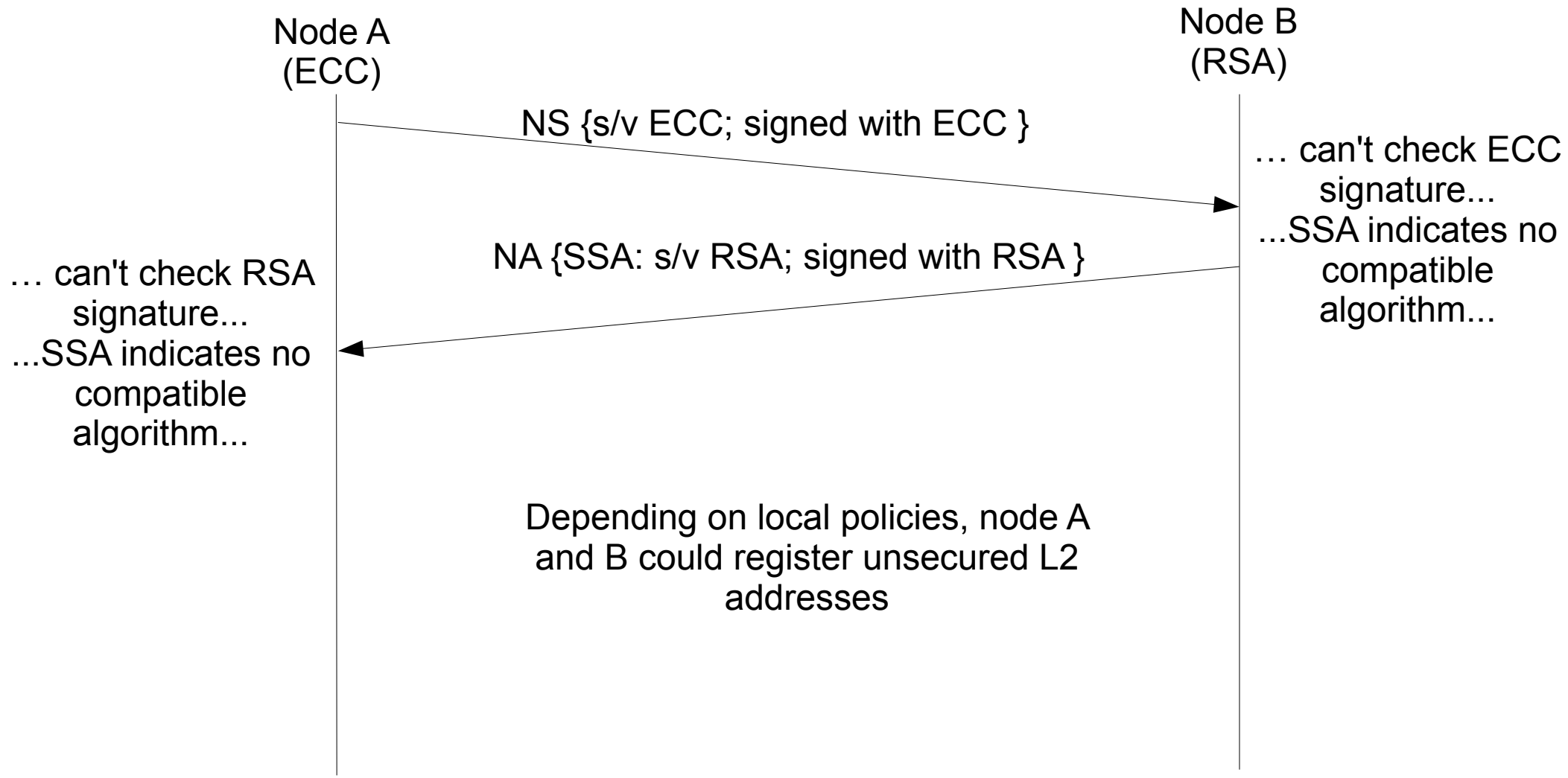
Negotiation phase with legacy RFC 3971 nodes



Negotiation phase with compatible nodes



Negotiation phase with incompatible nodes



Authorization Delegation Discovery

- Is this out of the scope of our work ?
- Should we state requirements such as:
  - « the Certification Path SHOULD only contain certificates containing verifiable content for the requesting node based on its Supported Signature Option »
  - « if the router as a CGA based on algorithm XXX, then all the Certification Path certificates must use the same XXX algorithm »

Thanks for listening

draft-cheneau-csi-cga-pk-agility-00

draft-cheneau-csi-send-sig-agility-00

draft-cheneau-csi-ecc-sig-agility-00

Questions ? Thoughts ? Volunteering for reviewing ?