Security in IPv6 Neighbor Discovery

Christian Vogt, chvogt@tm.uka.de

- IPv6 Neighbor Discovery
- Vulnerabilities
- Secure Neighbor Discovery (SEND)
- Remaining vulnerabilities

IPv6 Neighbor Discovery Protocol

- Network auto-configuration for IPv6 hosts
- Specified in RFC 2461bis, RFC 2462bis
- State managed in 4 data structures
  - Default router list: IP addresses of available first-hop routers
  - Prefix list: on-link IP address ranges
  - Destination cache: IP destination → IP next hop
  - Neighbor cache: IP next hop → MAC next hop
- ICMPv6 messages for link-local signaling
  - Neighbor Solicitation
  - Neighbor Advertisement
  - Router Solicitation
  - Router Advertisement
  - Redirect

NDP Components

- Router discovery
  - Find default/first-hop routers
  - Discover on-link prefixes ⇒ which destinations are neighbors
  - Additional information, e.g., MTU
  - Messages: Router Solicitation, Router Advertisement (exchange or periodic advertisements)
- Address auto-configuration
  - Auto-configure IPv6 addresses
  - Stateless (default) or stateful (e.g., through DHCPv6)
  - Based on prefix information delivered in Router Advertisements
  - ≥ 1 address per on-link prefix
  - DHCPv6 provides more than just addresses, such as DNS server, SIP server, NTP server, NIS server, etc.

NDP Components (2)

- Duplicate Address Detection (DAD)
  - Verify IP address uniqueness
  - Send Neighbor Solicitation, listen for (defending) Neighbor Advertisement
- Address Resolution
  - Resolves IP address into MAC address
  - Creates neighbor cache entry
  - Exchange of Neighbor Solicitation, Neighbor Advertisement
- Neighbor Unreachability Detection
  - Re-verify bidirectional reachability of neighbors
  - Keeps neighbor cache clean
  - Uses upper-layer information when possible
  - Exchange of Neighbor Solicitation, Neighbor Advertisement

NDP Components (3)

- Redirect
  - Redirect host to better router
  - Redirect host (from first-hop router) to neighbor
  - Updates neighbor cache entry
  - Transmission of Redirect

Conceptual Security Issues in NDP

- No proof of IP address ownership
  - Attacker can claim victim's IP address
- No IP-MAC address binding
  - Facilitates combination of proxies and bridges
  - Avoiding cross-layer interaction eases implementations
  - Attacker can bind its IP address to victim's MAC address (even if there was a proof of IP address ownership)
- IPsec authentication difficult to deploy
  - IKE not applicable (NDP needed for IP connectivity, IP connectivity needed for IKE, IKE needed for NDP)
  - Requires manual key distribution
  - Potentially many to-be-configured security associations

Vulnerabilities in NDP

- Attack on address resolution
  - Attacker creates false entry in victim's neighbor cache
  - Based on spoofed Neighbor Advertisements, Neighbor Solicitations, Router Solicitations
- Bogus on-link prefix
  - Attacker makes victim believe destination is on-link
  - Attacker can respond to victim's address resolution signaling
- Attack on address configuration
  - Attacker spoofs Router Advertisement with false on-link prefix
  - Victim generates IP address with this prefix
  - Access router drops outgoing packets from victim (ingress filtering)
  - Incoming packets can't reach victim

Vulnerabilities in NDP (2)

- Attack on DAD
  - Attacker spoofs "negative acks" for victim's DAD attempts
  - Victim can't configure IP address ⇒ can't communicate
- Attack on router discovery
  - Attacker tricks victim into accepting itself as default router
  - Based on spoofed Router Advertisements
- Redirect attack
  - Attacker makes victim send packets to arbitrary MAC address
  - Based on spoofed Redirects (from victim's default router)

Vulnerabilities in NDP (3)

- Attack on neighbor unreachability detection
  - Attacker causes solicitor to keep incorrect neighbor cache entry
  - Based on spoofed Neighbor Advertisements
- Replay attacks
  - Attacker replays message with correct address ownership proof and/or signature
- Configuration attacks
  - Attacker spoofs Router Advertisement with false configuration information (e.g., false MTU)

How SEND Tackles This

- Address ownership proof
  - Makes stealing IPv6 addresses "impossible"
  - Used in router discovery, DAD, address resolution
  - Based on Cryptographically Generated Addresses (CGA)
  - Alternative: non-CGAs with certificates (not specified in SEND; future work)
- Message signatures
  - Message integrity protection + sender authentication
  - Used in all NDP messages
  - RSA signature only (ensure compatibility and reduce implementation) complexity

How SEND Tackles This (2)

- Authorization of router functionality
  - Authorizes access router to provide…
    - network prefixes
    - packet forwarding
    - other info distributed by router discovery
  - Based on router certificates
- Replay protection
  - Timestamps for (unsolicited) multicast messages
    - Does not require state, but (loose) clock synchronization
  - Nonces for solicitation-advertisement exchanges
    - State determined by solicitor, echoed by advertiser

What's a CGA?

- IPv6 address with IID = hash of public key (+ parameters)
- CGA = identifier
- Natural binding public key ↔ identifier
  - Public key → identifier: simple hash
  - Identifier → public/private key: brute force
  - Combined with signature + nonce/timestamp verifies knowledge of public/private key pair

CGA Generation

1. SEC == security parameter (0..7 = 3 bits)
2. modifier := random()
3. hash2 := first(112, SHA1(modifier | 9 zero-octets | public key* | optional extension fields))
4. IF first(16*SEC, hash2) ≠ 0 THEN modifier++; goto step 3
5. collision_count := 0
6. cga_parameters := modifier | subnet prefix | collision_count | public key* | optional extension fields
7. hash1 := first(64, SHA1(cga_parameters))
8. IID := hash1; IID[0..2] := SEC; IID[7] := 0 (u-bit); IID[8] := 0 (g-bit)
9. IF DAD positive THEN collision_count++; goto step 6

* DER-encoded ASN.1 structure of type SubjectPublicKeyInfo

CGA Generation (2)

- If SEC > 0, CGA generation not guaranteed to stop after certain iterations
- Generation of CGA with high SEC infeasible with today's technology ⇒ scalable to future advances
- CGAs generated from same public key unlinkable, b/c initial modifier value random ⇒ privacy
- Modifier reusable during network renumbering

CGA Verification

1. Validate cga_parameters
   a. IF collision_count ∉ {0,1,2} THEN return FAILURE
   b. IF prefix ≠ CGA prefix THEN return FAILURE
2. hash1 := first(64, SHA1(cga_parameters))
3. IF hash1[3..5,8..63] ≠ IID THEN return FAILURE
4. SEC := IID[0..2]
5. hash2 := first(112, SHA1(modifier | 9 zero-octets | public key* | optional extension fields))
6. IF first(16*SEC, hash2) ≠ 0 THEN return FAILURE
7. return SUCCESS

New NDP Options in SEND

- CGA option
  - Required in all NDP messages hosts may originate
  - Not required in Router Advertisements and Redirects
- RSA Signature option
  - Required in all NDP messages with CGA
  - Not required in Router Solicitations with unspecified IP source address (no way to bind public key to CGA)

Authorizing Router Functionality

- Authorize routers
  - to forward packets
  - to advertise certain prefixes
- Access routers have certificates
- Certification path from router to trusted party
- Hosts pre-configured with trust anchors
- Trust anchors delegate network prefixes
  - to routers
  - to someone who delegates them further, e.g. ISP

Authorization Delegation Discovery

- Initiated by host when certification path is unknown upon receipt of Router Advertisement
- Exchange of Certification Path Solicitation, Certification Path Advertisement
- Uses separate messages, rather than options to existing NDP messages
  - Potentially much data (= long certification path)
  - ADD infrequently executed
- Certification paths cached by hosts

No Authorization for Other Param's

- Information not authorized includes…
  - Router's IP address
  - Router lifetime
  - Prefix lifetimes
  - Address configuration mode (stateful or stateless)
- But forwarding and prefixes most fundamental
  - Prefixes determine address configuration and topological location
  - Authorization for forwarding ⊃ authorization for router lifetime
  - Authorization for prefix ⊃ authorization for prefix lifetime and address configuration mechanisms

Certification Path Example

Certification Path →

Issuer: isp_group_example.net
Validity: Jan 1, 2004 - Dec 31, 2004
Subject: isp_foo_example.net
Extensions:
IP address delegation extension:
Prefixes: Q1, ..., Qr

... possibly other extensions ...
... other certificate parameters ...

Pre-configured into host ↓

Issuer: isp_group_example.net
Validity: Jan 1, 2004 - Dec 31, 2004
Subject: isp_group_example.net
Extensions:
IP address delegation extension:
Prefixes: P1, ..., Pk

... possibly other extensions ...
... other certificate parameters ...

Issuer: isp_foo_example.net
Validity: Jan 1, 2004 - Dec 31, 2004
Subject: router_x.isp_foo_example.net
Extensions:
IP address delegation extension:
Prefixes: R1, ..., Rs

... possibly other extensions ...
... other certificate parameters ...

$P1 \ldots Pk \supset Q1 \ldots Qr \supset R1 \ldots Rs$

Deployment Model

- Centralized trust anchor
  - Single, global trusted authorization root
  - Hosts pre-configured with public key(s) of global root
- Decentralized trust anchors
  - Multiple trusted authorization roots
  - Hosts pre-configured with public keys from all roots

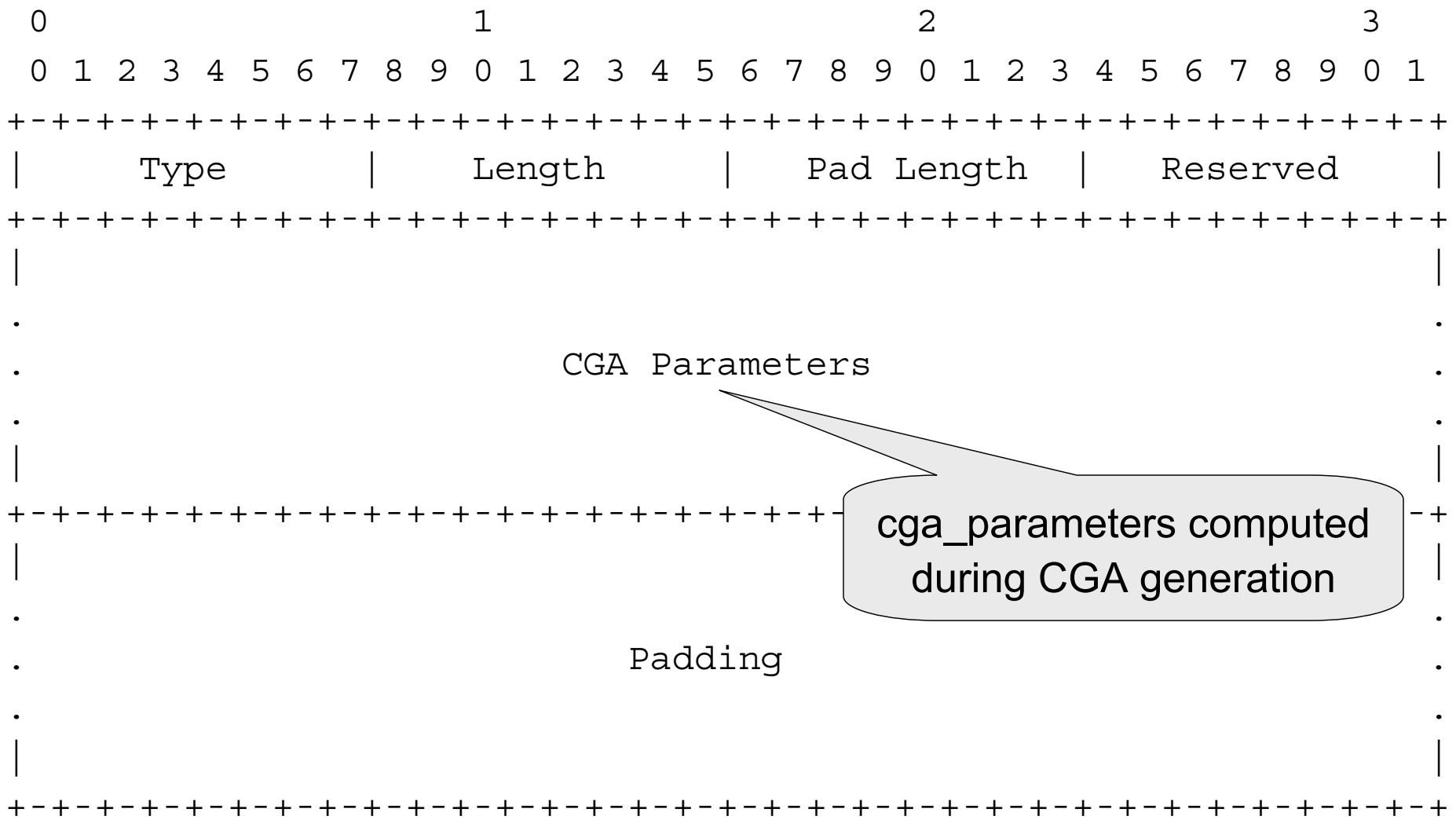
NDP Message Protection

- Neighbor Solicitation message
  - CGA option MUST; RSA Signature option MUST
  - CGA = target address during DAD, CGA = source address otherwise
- Neighbor Advertisement message
  - CGA option MUST; RSA Signature option MUST
  - CGA = source address
- Router Solicitation message with specified IP source address
  - CGA option MUST; RSA Signature option MUST
  - CGA = source address
- Router Advertisement message
  - CGA option MAY; RSA Signature option MUST
  - CGA = source address
- Redirect message
  - CGA option MAY; RSA Signature option MUST
  - CGA = source address

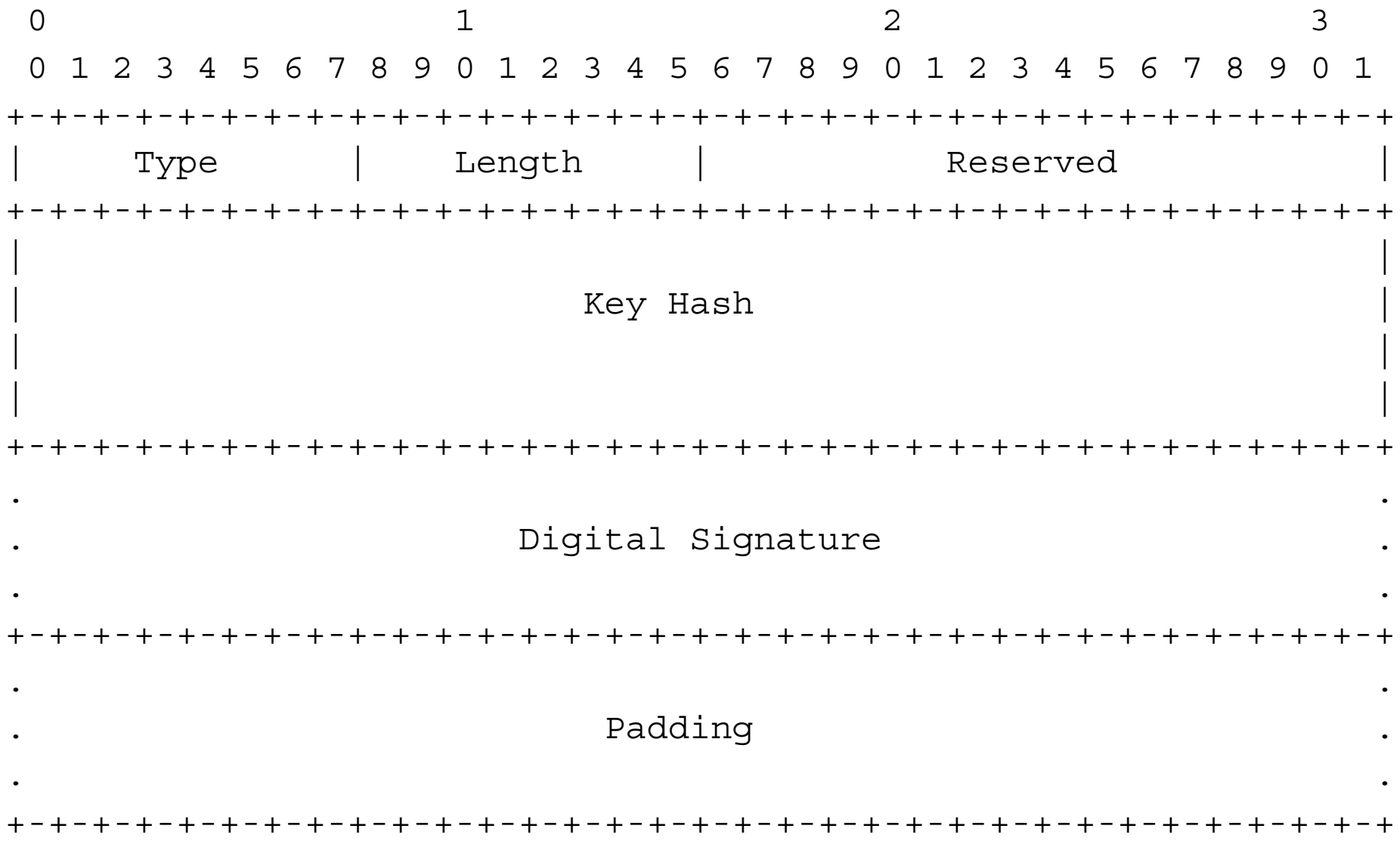
NDP Message Validation

- CGA verification
  - Tells if public key belongs to CGA owner
  - Involves only hashes
  - Little time-consuming
- RSA signature verification
  - Tells if message sender == CGA owner?
  - Only if CGA verification successful
  - Involves public-key cryptography
  - More time-consuming

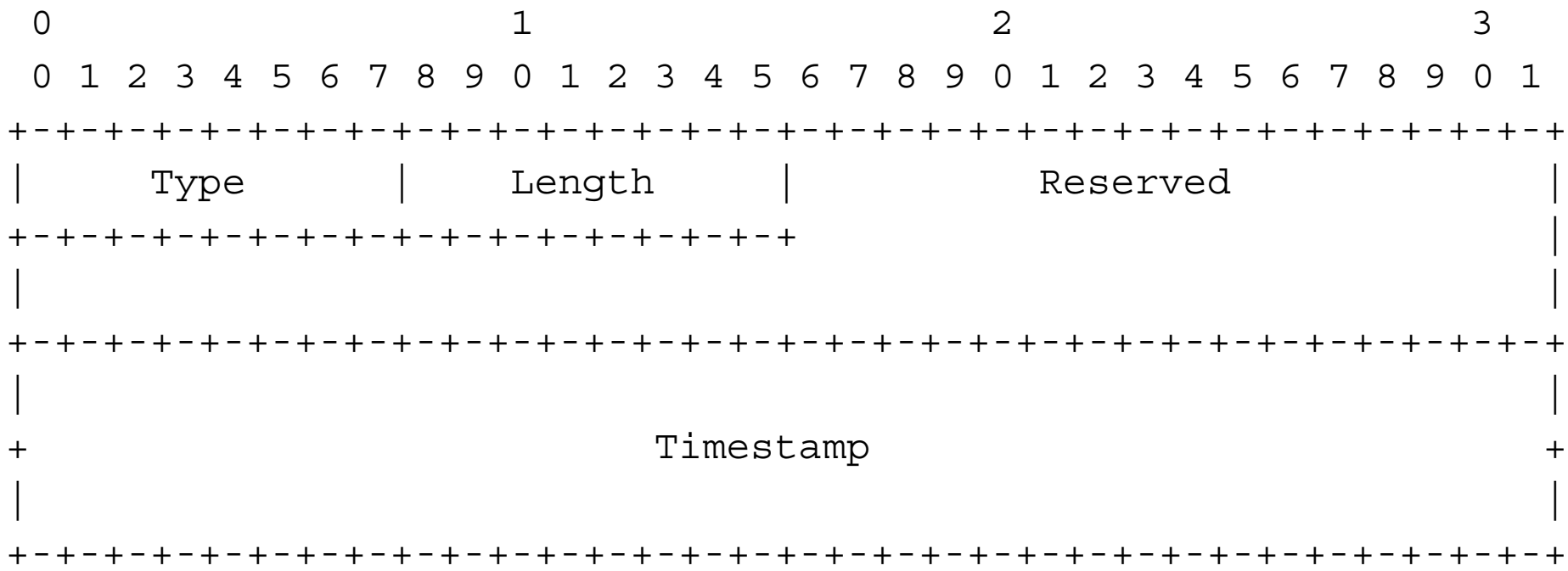
CGA Message Option



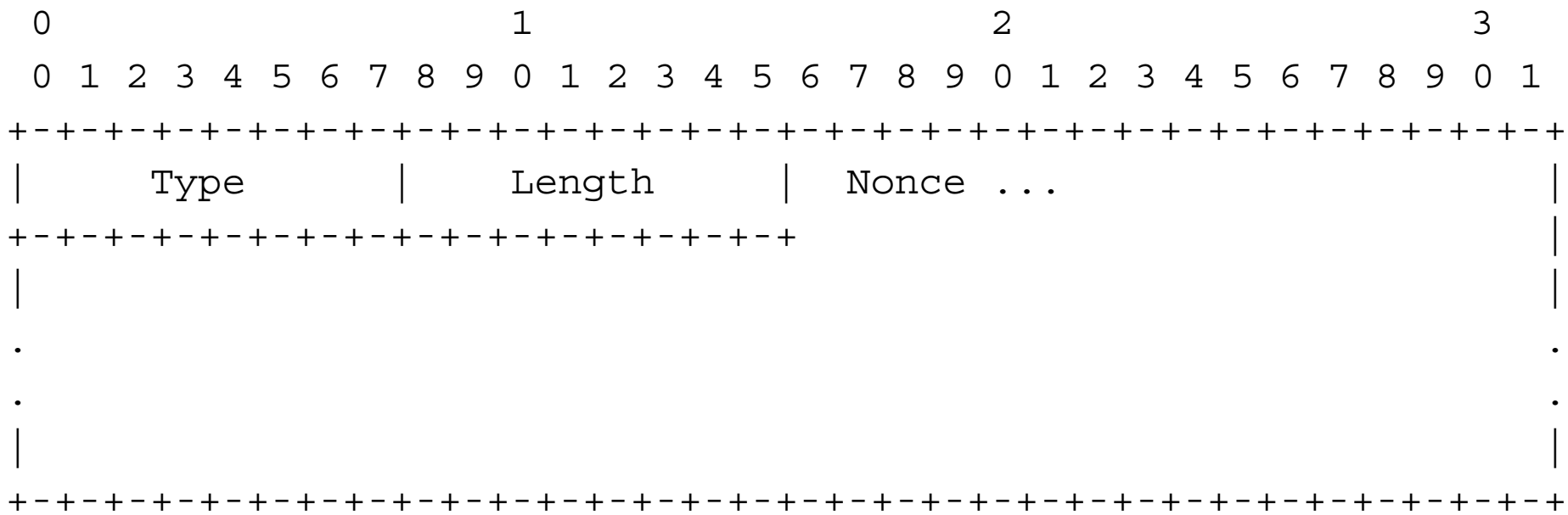
RSA Signature Message Option



Timestamp Message Option



Nonce Message Option



Remaining Vulnerabilities in SEND

- No confidentiality
- No link-layer security
  - Packets may come from attacker using false IP source CGA
- Still no cryptographic IP-MAC address binding
  - Map victim's MAC address to attacker's CGA
  - Attacker can bombard victim with large download
- Router flooding
  - Bogus packets with spoofed IP destination addresses cause routers to do (lots of) address resolution
  - Alternative solutions: rate limitations, restricting state for pending address resolution processes

Remaining Vulnerabilities (2)

- Brute force against CGA (only 59 "cryptographic bits")
  - Modifier makes brute force harder
- Computational exhaustion
  - Bogus signatures cause hosts to spend time on verification
- Authorization delegation discovery
  - Large number of bogus requests for long certification paths
  - Solution: Routers cache certification paths + negative responses
  - Send large number of (unsolicited) bogus responses
  - Cause hosts to cache this info
  - Solution: Hosts limit cache size, prioritize solicited information
- Replay during timestamp window
  - Negligible (unless advertised information changes)

Problem(s) with SEND

- Issue during design of NETLMM
  - Reuse routers' link-local addresses on different links
  - But: Link-local addresses based on public key, which must be unique per router
  - Link-local addresses can't be reused (identification problem in SEND)
- Any more…?