

# Cryptographically Generated Addresses (CGA)

**Tuomas Aura**  
Microsoft Research  
tuomaura@microsoft.com

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      1

## Outline

1. Basic idea of CGA addresses
2. Applications
3. Solutions to various problems
4. CGA limitations and advantages

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      2

## IPv6 Address

64-bit Subnet Prefix      64-bit Interface Id

**FEDC:9773:D983:4325**    **F56C:74C4:9212:02BA**

- Nodes attached to the same gateway router have the same subnet prefix but different interface ids.
- 62 bits of the interface id can be chosen arbitrarily, e.g., randomly. 2 bits have a special semantics.

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      3

## CAM Address

- First CGA proposal (O’Shea and Roe 2000) for authentication of Mobile IPv6 binding updates.
- Interface id is created from a truncated SHA-1 hash of the address owner’s public key:

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      4

## Proof of Address Ownership

- Node sends the public key, collision count and a signed message from the CGA address.
- Receiver recomputes the hash, compares with the interface id of the source address, and verifies the signature using the public key.
  - ➔ Receiver knows that the message was sent by the owner of the source address.

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      5

## Applications

- Prevents spoofing of someone else’s IP address.
- Address autoconfiguration and duplicate address detection.
- Neighbor discovery and redirection.
- ICMP authentication.
- Mobile IP binding update authentication.
- Opportunistic IPSec.

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      6

### Problem 1: Brute-Force Attacks

- Attacker tries to find a public key (and collision count) whose hash matches someone else's address.
- 62 bits is barely enough.
- Possible to search many values in parallel.
- Pre-computation attack: create a database of  $2^{62}$  interface ids and matching public keys.

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      7

### Solution 1: Hash the Subnet Prefix

Hash (Subnet Prefix | Collision Count | Public Key)

64 bits      62 hash bits

Subnet Prefix      Interface Id

ug=00

- Attacker must create a separate database for each subnet prefix.
- Helps for globally routable addresses, not for link-local addresses.

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      8

### Problem 2: Moore's Law

- Attacker's computing power grows exponentially with time.
- A few bits in hash length (e.g. 62 vs. 64) makes little difference in the long term.
- Non-solution: adjust the interface-id and subnet-prefix lengths.

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      9

### Solution 2: Hash Extension

- A second hash must begin with k zeros:

Hash1 (Modifier, Subnet Prefix, Collision Count, Public Key)

62 hash bits

Subnet Prefix      Interface Id

ug=00

Hash2 (Modifier, Subnet Prefix, Collision Count, Public Key)  
= 0000xxxxxx

- Address generation: change modifier until the second hash begins with enough zeros.

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      10

### Hash Extension – Analysis

- Address generation requires a brute-force search for the modifier.  $\Rightarrow O(2^k)$  work.
- Brute-force attack also becomes  $2^k$  times more expensive.  $\Rightarrow O(2^{62+k})$  work.
- Cost of address use and verification is constant.
- Database attack impossible. Also link-local addresses protected.

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      11

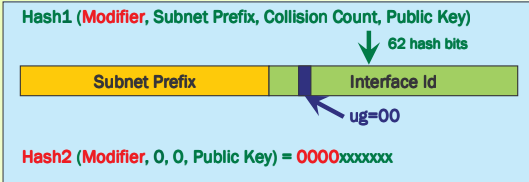
### Problem 3: Mobility

- When a mobile nodes gets a new subnet prefix, it must redo the  $O(2^k)$  search.
  - $\Rightarrow$  Hash extension too expensive for mobiles.

ISC 2003, Bristol      Tuomas Aura, Microsoft Research      12

### Solution 3: Easier Mobility

- No subnet prefix in Hash2:



- Recompute only Hash1 when the subnet prefix changes.

### Easier Mobility – Analysis

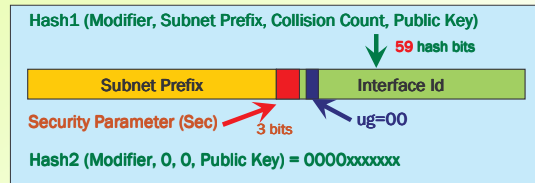
- At least as secure as hash extension without the subnet prefix in either hash.
- At least as secure as hashing the subnet prefix without hash extension.
- Effect: cost of public-key generation is multiplied by  $2^k$ .

### Problem 4: Parameterizing Security

- $k$  should be a parameter.
  - ➔ Increase  $k$  over time.
  - ➔ Servers more vulnerable than client PCs.
  - ➔ Address owner should decide its own  $k$ .
- How does the verifier learn  $k$ ?
- Non-solution: send  $k$  in a protocol message.
- Non-solution: make  $k$  a function of time.

### Solution 4: Security Parameter

- Solution: Encode  $k$  in the address bits.



- Hash2 must begin with  $k = 16 \cdot \text{Sec}$  zero bits.

### Problem 5: Bidding Down

- Which addresses are CGA and which are not?
- Cannot trust the address owner to tell. Attacker can claim that it is not using CGA and avoid verification.

### Solution 5: Type Bits

- Unused combination of “g” and “u” bits ( $g=1$  and  $u=1$ ) in the interface id.
  - ➔ Use as a type tag for CGA.
- Effectively allocates 25% of the IP address space for CGA.
- Not popular in IETF.

### Solution 5: Living without Type Bits

- **Type bits not popular in IETF.**  
Will have to set  $u=0, g=0$ . How to cope?
- Cannot use CGA and unauthenticated addresses as equals side by side.
- New protocols may require CGA addresses.
- Private networks may require CGA locally.
- Two equally strong security mechanisms (e.g. CGA and PKI) may be used side by side.
- **Our solution:**  
Accept both but give priority to CGA addresses and signed information.

ISC 2003, Bristol

Tuomas Aura, Microsoft Research

19

### CGA Limitations

- CGA-based authentication prevents spoofing of source IP addresses.  
It **does not prevent DNS spoofing.**
- Prevents spoofing of **someone else's IP** address. An attacker can generate a new address with any subnet prefix.  
➔ **CGA does not prove that the node or address exists.**

ISC 2003, Bristol

Tuomas Aura, Microsoft Research

20

### CGA Advantages

- Authentication of an IP address without PKI or other security infrastructure.
- Can prevent many DoS attacks.
- With Secure DNS, gives strong authentication.
- Particularly suitable for **authenticating IP-layer signaling.**

ISC 2003, Bristol

Tuomas Aura, Microsoft Research

21

### Conclusion

- CGA addresses enable **authentication of existing IPv6 addresses without any security infrastructure.**
- We made critical improvements to CGA addresses that make them usable. Effectively **removed the 62-bit limit on hash length.**
- Work in progress on applications and an IETF standard.

ISC 2003, Bristol

Tuomas Aura, Microsoft Research

22