RESEARCH ARTICLE OPEN ACCESS

Design and Implementation of Ipv6 Address Using Cryptographically Generated Address Method

Ashish Singh*
**Department of Computer Science, Samrat Ashok Technological Institute, Vidisha, M.P, India*

ABSTRACT
There is always a tradeoff between privacy and the desired level of security for any internet user in the contemporary cyber world. Cyber security, of late, is paramount and its breach could lead to untoward consequences, at times, disastrous. The advent of the IPv6 provides a hope to resolve this tradeoff satisfactorily. Included in the IPV6 suite is a method for devices to automatically configure their own addresses in a secure manner. This technique is called Cryptographically Generated Addresses (CGAs). CGA provides the ownership proof necessary for an IPv6 address without relying on any trust authority. However, the computation involved in CGAs is very high, especially for a high security level defined by the security parameter (Sec). The sheer cost involved here may pose to be an inhibiting factor for any user to continue with this security regime and may tempt her not to change her address on a frequent basis. Thus, the way forward could be to modify the standard CGA to make it more applicable across applications and scenarios and at the same time not to let it compromise with the optimum security level. We propose to reduce the CGA granularity of the security level from 16 to 8, which make it more feasible for use in most applications and scenarios. And the privacy part is taken care of by changing addresses over time which protects users from being tracked. Here, we strive to implement and evaluate these extensions to the standard CGA.

Keywords: Cyber security, IPv6, security parameter, granularity

I. INTRODUCTION

Today many people talk about Internet Privacy[1] and Security[1], particularly since the deployment of Internet Protocol version 6 (IPv6), which is the next generation of Internet Protocol replacing IPv4. In the IPv6 address, 64-bits from the left of the address of 128 bits form the subnet prefix and 64 bits from the right forms the interface identifier [3]. Privacy is a very important element in everyone’s daily life. Simply the attempt to characterize the contrast in the middle of protection and security turns into a troublesome undertaking as they are so firmly related. This is the reason the importance of protection is not conclusive among the specialists on the grounds that there is no settled separating line in the middle of security and security. Most clients might not want to have their information presented to other individuals on the web [4].

1.1. Background

The rate at which the consumption of IPv4 address space which is going on along with its inefficient performance as compared to others would eventually finish the resources within a short time span. Hence it was needed to develop a new version which could have larger address space. Henceforth a great clash of opinions occurred with many different proposals. But eventually owing to its multiple advantages IPv6 was declared to be the standard.

IPv6 which is Internet Protocol version 6 was started to come in use and it was a great improvement to the previously usedIPv4. Initially IPv4 used 32 bit address scheme and by using that scheme it could accommodate up to around 4 billion addresses. Hence to come over these shortcomings the internet engineering task force (IETF) started to develop in early 90’s the new set of rules the IPv6 IP protocol. This new protocol will thereby use 128 bit address space. And the major striking feature is that it will support a large number of unique addresses.

1.2. Features Of Ipv6

IPv6 consist of 128 bits as compared to 32 bits in IPv4. With the help of IPv6 address it is now possible to support 2^{128} unique IP addresses.

The features of IPv6 protocol are listed as follows:

- Minimized header size
- Greater space for address
- Efficient and hierarchical addressing and routing infrastructure
- Built-in security
- New protocol for neighboring node interaction
- Extensibility
-

1.3. IP SECURITY

IPSec provides a combined set of cryptographic protocols which gives the security to data communication and security to exchange keys. IPSec has two wire-level Protocols namely

Authentication Header (AH) and Encapsulated Security Payload. IPSec also provides a third suite of protocol and internet key exchange. It also maintains the information whether the communication is secure or not all times. The division of IPv6 locations into particular topology and interface identifier segment raises an issue new to IPv6 in that an altered part of an IPv6 address (i.e. interface identifier) can contain an identifier that remaining parts steady notwithstanding when the topology segment of a location changes (e.g. as the after effect of associating with an alternate piece of the internet) [4]. The conceivable way to deal with abstain from having a static non-changing location would be to change the interface identifier part of a location over the long haul and create new addresses from the interface identifier for some location scopes. One such approach is Cryptographically Generated Addresses (CGAs), RFC3972, random identifier is generated based on the node's public.

II. CRYPTOGRAPHICALLY GENERATED ADDRESSES

Cryptographically Generated Address [4], IPv6 addresses validations are offered by CGA and keep vindictive hubs from asserting the responsibility for others' addresses. For using it, sender center point needs to pick the Security Parameter (Sec). Sec is an unsigned 3-bit entire number having a value some place around "0" and "7". It grows the computational cost for both the assailant and the area generator. CGA period time depends on after enrolling device CPU speed.

2.1 Algorithm For The Formation Of Cga

In CGA, cryptographic hash is used for the location of interface identifier [5] of the location proprietor's open key and also many other helper parameters. As the last 64-bit are insufficient to give adequate security as compare to beast power assaults in the advancing future, the x CGA utilizes the Hash Extension[3] to build the privacy &security quality over 64-bit.

2.1.1 Steps For Cga Generation Algorithm:

I. CGA era Algorithm starts with deciding the location proprietor's open key and selecting the best possible Sec quality.

II. The Hash2 [4] processing circle then proceeds until discovering the last Modifier. The Hash2 worth is a hash of the blend of the public key and the modifier which are linked with a zero-quality for collision count and subnet prefix.

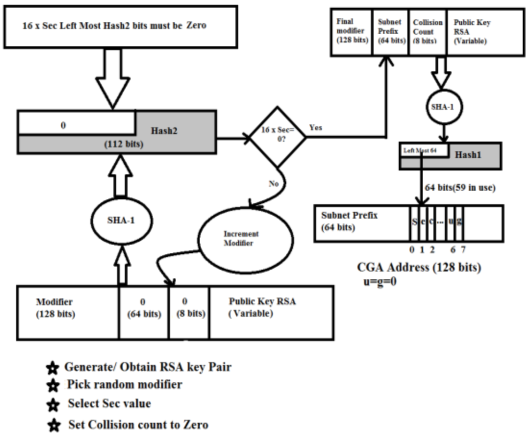


Figure 1 Algorithm for CGA Generation [3]

2.1.2 Flow Chart Of Cga Generation Algorithm:

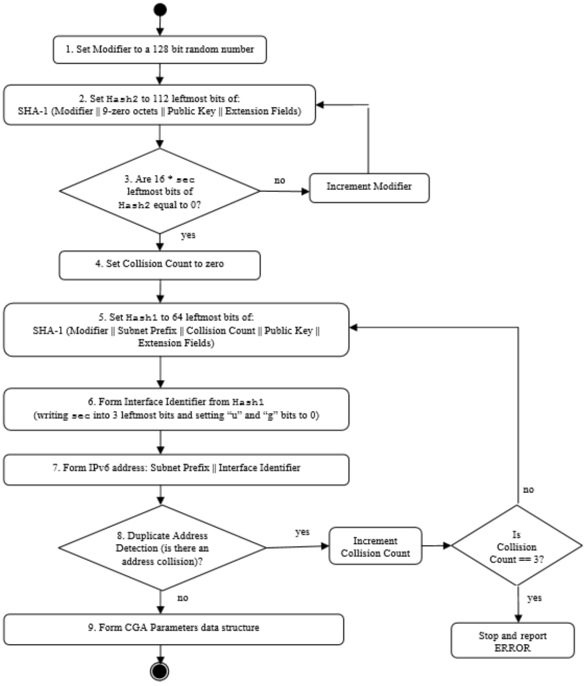

1. Set Modifier to a 128 bit random number
2. Set Hash2 to 112 leftmost bits of: SHA-1 (Modifier || 9-zero octets || Public Key || Extension Fields)
3. Are 16 * sec leftmost bits of Hash2 equal to 0?
no
Increment Modifier
yes
4. Set Collision Count to zero
5. Set Hash1 to 64 leftmost bits of: SHA-1 (Modifier || Subnet Prefix || Collision Count || Public Key || Extension Fields)
6. Form Interface Identifier from Hash1 (writing sec into 3 leftmost bits and setting "u" and "g" bits to 0)
7. Form IPv6 address: Subnet Prefix || Interface Identifier
8. Duplicate Address Detection (is there an address collision)?
yes
Increment Collision Count
Is Collision Count == 3?
no
yes
no
9. Form CGA Parameters data structure
Stop and report ERROR

Figure 2: Flow chat of CGA Generation Algorithm[2]

III. The location generator tries diverse estimations of the modifier till 16×Sec-furthest left bits of Hash2 turn into zero.

IV. When a suit is discovered, the circle for the Hash2 reckoning ends. At that point the last Modifier worth is spared and utilized as a data for the Hash1 processing.

V. The Hash1 [4] quality is a hash of the blend of the entire CGA parameters. The Interface identifier is then gotten from Hash1 by composing the estimation of sec into three left-most bits and by setting bits 6 and 7 (i.e. , "u " and "g" bits) to zero.

VI. Finally, the DAD algorithm [6] is keep running by the customer to guarantee that the location is extraordinary inside of the same subnet. On the off chance that a location crash happens, increase the Collision Count and register Hash1 again to get the Interface identifier. On the other hand, after three crashes, CGA calculation stops and reports a blunder.

2.2 Computational Cost Of Cga Generation:

It is difficult for both the address generator and the attacker to afford the cost for computation. The location generator required $O(2^{16\times Sec})$ brute-force inquiry to fulfill Hash2 requirement and discovering the exact Modifier. The assailant will do a brute force attack against a (16×Sec+59)-bit hash esteem which costs $O(2^{16\times Sec+59})$[5].

2.3 *Setting A Lifetime For Temporary Cga Address:*

We are presenting to change CGA addresses periodically to protect the users' privacy. Each CGA address has an associated lifetime that indicates how long the address is bound to an interface [5]. Once the lifetime expires, the CGA address is deprecated.

Necessary notations:-

TG: The average time required for a node to generate a CGA.
TA: The average time required for the impersonation of address by attacker.
T1: The time needed to compute Hash1.
T2: The time needed to compute Hash2.
b: The number of bits that are available within the address, that is the truncated output of Hash1 (IID).
g: Security level in CGA.
S: The number of bits needed to satisfy the Hash2 condition$(s = g \times Sec)$, which is the truncated output of Hash2.

The address generator needs on average (2^sx T2) to fulfill the condition of Hash2, adding T1 to build IID from Hash1. Therefore, the cost of generating address is:

$$TG = \left(2^{(g\times Sec)} \times T2\right) + T1 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

The total time for the impersonation when we begin with Hash1 (H1) is given by

$$TA\,(H1) = (2^b \times T1 + T2) \times 2^S \quad \ldots\ldots\ldots\ldots(2)$$

When the attacker starts from Hash2

$$TA : (H2) = (2^S \times T2 + T1) \times 2^b \quad \ldots\ldots\ldots\ldots(3)$$

The attacker has choice between the two ways for the cost of attack to be minimized.

Hence, the time for impersonation an address (TA) is:

$$TA = min\{(2^{59} \times T1 + T2) \times 2^{g\times Sec}, (2^{g\times Sec} \times T2 + T1)2^{59}\} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4)$$

2.4 Drawbacks Of Cga:

With the help of CGA a node has the quality to generate interface identifier portion of its IP address, with the use of hashing securely. It not just accommodates generation of a random interface identifier, additionally it give the hub evidence of location proprietorship, RFC3172.

A security parameter "Sec" is chosen. The sec is a worth somewhere around 0 and 7. The higher the "Sec" esteem the better will be the assurance managed a hub against brute force attack. At the point when a "Sec" worth higher than zero is picked, there is no ensure that 16 by sec equivalent to zero condition will ever be met. So the issue with the utilization of the CGA calculation is the way that it is process serious. It is consequently that when the hub creates the IP location utilizing CGA, it doesn't transform it and keeps on utilizing it always, in this manner making this hub helpless against security related attacks [2].

III. PROPOSED WORK

For usability, effortlessness, and pragmatic prerequisites, it is ideal to focus Sec esteem in a programmed or aberrant route in view of unmistakable element, for example, time

3.1. Adding Condition To CGA

To ensure that CGA era procedure ends after an eventually, the adjusted CGA calculation time is taken as an information to focus the end time. On the off chance that this time limit surpasses, it will automatically stop. The calculation stays informed regarding the best found worth which has most astounding multiple zeros found in furthest left bits of Hash2. Calculation stays informed regarding the best found worth which the most astounding number of zeros is found in furthest left bits of Hash2. In fact, the greatest bearable CGA address era time relies on upon a few variables. It relies on upon the gadget registering power, the specific application prerequisites, and different elements, for example, to what extent the client is willing to sit tight for CGA era. In this way, it is expected to choose the correct end time to get a plausible Sec value [5] for CGA generation.

It is ideal to situate the security parameter (Sec) naturally in view of an end time as opposed to designing it by the location generator (proprietor) for the accompanying reasons:

- It's absurd to request that client comprehend subtle elements of the CGA calculation to choose best possible Sec quality. In any case, it is conceivable to offer the client the likelihood to choose the quality on the off chance that user knows points of interest of CGA calculation.
- It's hard to focus on best possible Sec quality, on the grounds that Sec worth has enormous

impact on computation and security related expense.

- It relies upon time that hub has got to design its address and on the pre requirements of application. When we talk about, portable correspondence, hub ought to obtain the address within limited specified time for the handover to be accomplished.
- The speed of CPU is not high for all the gadgets. Particularly, inserted & portable PCs become sluggish as compare to the desktop workstation. Regardless of the fact that client knows points of interest related to calculation of CGA, it's complicated to reasonable Sec worth for particular gadget. On the off chance that the client has the likelihood to choose fitting parameters, it respects the client any rate, harsh calculation related to normal time taken for particular Sec quality in view of his gadget details.

3.2. Selecting The Security Level Of Condition The Hash Extnsion

The temporal ceasing of conditions might squander CPU assets on grounds that the different variable "16" is moderately expansive. CGA generator [5] registers different Hash2 [4] qualities amid a period characterized when parameter data, and the yield quality is the particular case that has got the best amount of the bits corresponding to zeros. As we are aware that quantity of bits corresponding to zeros as communicated within 16×Sec, better security quality will be controlled by making Hash2 selection.

Littler numerous elements are appropriate for when it comes to TB-CGA. In this way, for the TB-CGA, choosing component "8" rather than "16" is better choice more sensible hence:-

- To reduce the wasting time.
- The opportunity to the have "8" progressive bits corresponding to zero is higher as compared to "16" progressive bits corresponding to zeros.
- Presently, Sec esteem "0" or "1" can be utilized as a part of commonsense related application. Sec=2, (CGA) era might take more time.

3.3. Generation Algorithm For Temporal Cga

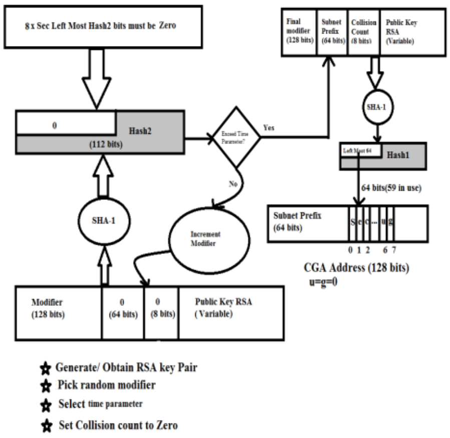


Figure 3: Illustration of Temporal CGA Algorithm[3]

Figure no. 3demonstrates a schematic of TB-CGA era calculation. Rather than Sec esteem, the parameter of time will be utilized as a data. If the condition arises that the parameters of time is still less than increase the modifier and then again calculate the value of Hash2. Amid the beast power hunt circle, the Hash2 value that corresponds to biggest number of bits corresponding to in the furthest left bits will be put away. Additionally, the relating the value of modifier that results in the best value of hash2 worth will be put away. Now suppose, the parameter of time exceeds and also the loop gets terminated , the value of modifier corresponding to highest obtained sec parameter value would now be utilized to generate CGA address along with its verification.

IV. IMPLIMENTATION DETAILS OF TB-CGA

The implementation environment viz. programming language, operating system, library packages used are mentioned. Also described is the input and output details of the algorithm along with the methodology followed to simulate and implement the proposed TB-CGA algorithm.

4.1. Implementation Evironment

Environment details of the implementation of the proposed TB-CGA algorithm are tabulated in Table 1 below.

Table 1: Implementation Environment

Programming Language	C
Operating System	Linux
Library Packages Or Apis Used	Openssl

4.2. Input

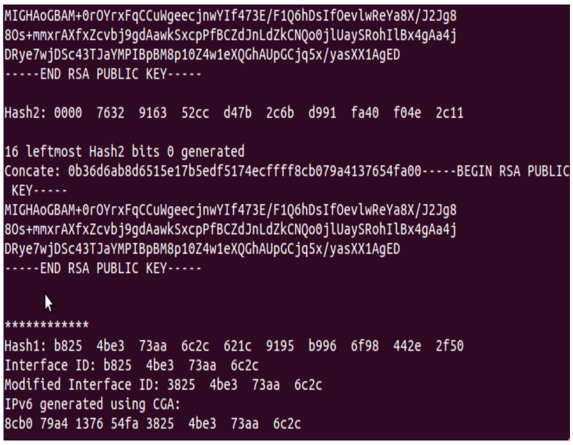
Input will be 128 bit modifier, RSA public key which is of variable length, 64 bit subnet prefix and 8 bit impact count for the uniqueness of the location which is created. For providing the security we used SHA-1 (160 bits). We have also used the security parameter (3 bit); it defines the security level of CGA against Brute force attack

4.3. Implementation Details

The output of the Time Base CGA is secure IPv6 address using CGA including the Has1, modifier ID, Interface ID.

CGA era time can be defined as the aggregate term of entire CGA task. This incorporates time to producing RSA open or private keys, time that is spend in processing Hash2 esteem corresponding to condition 16×Sec-furthest bit left of value of Hash2 will be equivalent as null, also time required to registering interface identifier that includes Hash1 figuring. Close to DAD check.

Aggregate time for CGA era corresponding to Sec =1 is more noteworthy as compare to normal CGA era time requirement in case of sec=0. Sec esteem "2" could be utilized as a part of the following forthcoming years. Although the computation of sec estemm3 is not feasible and cannot be attained owning to present speed of CPU. Still, Sec esteem "3" is not computationally attainable for the current CPU speeds subsequent to the Sec quality builds CGA processing enormously



```
MIGHAoGBAM+0rOYrxFqCCuWgeecjnwYIf473E/F1Q6hDsIfOevlwReYa8X/J2Jg8
8Os+mmxrAXfxZcvbj9gdAawkSxcpPfBCZdJnLdZkCNQo0jlUaySRohIlBx4gAa4j
DRye7wjDSc43TJaYMPIBpBM8p10Z4w1eXQGhAUpGCjq5x/yasXX1AgED
-----END RSA PUBLIC KEY-----

Hash2: 0000  7632  9163  52cc  d47b  2c6b  d991  fa40  f04e  2c11

16 leftmost Hash2 bits 0 generated
Concate: 0b36d6ab8d6515e17b5edf5174ecffff8cb079a4137654fa00-----BEGIN RSA PUBLIC
 KEY-----
MIGHAoGBAM+0rOYrxFqCCuWgeecjnwYIf473E/F1Q6hDsIfOevlwReYa8X/J2Jg8
8Os+mmxrAXfxZcvbj9gdAawkSxcpPfBCZdJnLdZkCNQo0jlUaySRohIlBx4gAa4j
DRye7wjDSc43TJaYMPIBpBM8p10Z4w1eXQGhAUpGCjq5x/yasXX1AgED
-----END RSA PUBLIC KEY-----


************
Hash1: b825  4be3  73aa  6c2c  621c  9195  b996  6f98  442e  2f50
Interface ID: b825  4be3  73aa  6c2c
Modified Interface ID: 3825  4be3  73aa  6c2c
IPv6 generated using CGA:
8cb0 79a4 1376 54fa 3825  4be3  73aa  6c2c
```

Figure 4: The result of TB-CGA for the Sec value 0 with RSA Key 1024 bits

4.4. Security Analysis

4.4.1 Scanning Of Nodes

At the point when a hub keeps its IID for just eventually, and when it changes its IID likewise when the prefix changes, then it turns out to be extremely troublesome for an aggressor to discover the IP location of the hub. As the beginning period of any assault is examining the system to discover the IP location of the hubs, and afterward to run port checking with a specific end goal to locate the accessible administrations running on those hubs so as to exploit the vulnerabilities of the administrations running on the hub will bring about a tradeoff of the hub's security. Thus, by utilizing our calculation, not just do we furnish the hub with a level of security certification additionally we give the hub security insurance.

4.4.2 Tracking The Location

This is the same as hub filtering so it will be exceptionally troublesome for an assailant to track the hub over the system. The purpose behind this is on the grounds that it is exceptionally troublesome for an aggressor to perceive that this hub is the same hub, yet with a recently produced IID. It is extraordinarily genuine when there is a boundless number of a hub being used on the same system.

4.4.3 Gather The Confidential Information

At the point when a hub oftentimes changes its IID inside of the system furthermore among systems, then aggressors most likely won't have enough time to get the client's private information. It will truly be troublesome for an assailant to correspond the data that he does acquire to a particular client's IP address. This implies that it will be troublesome for the aggressor to get more data about this client in light of any relationship of information.

4.5 Limitations And Deployments Considerations

There are some implications and deployment considerations for changeable Addresses.

- Changing the location as often as possible has an execution implication and will severely impact user experience.
- Protecting the user's privacy may conflict with the administrative need to effectively maintain and debug the network.
- The changeable address might cause unforeseen challenges with a few applications. The networks which reject the connection from clients whose location can't be mapped into a DNS name that additionally maps again into the same location.[7]
- The implementation needs to keep track of the addresses being used by the upper layer in order to be able to remove the deprecated addresses from the internal data structure when these addresses are no longer used by the upper protocols, but not before.

V. CONCLUSION AND FUTURE WORK

5.1. Conclusion

It is very important to be sure that the increasing deployment of IPv6 is to be done in a secure way without compromising the Internet users' privacy. CGA gives a verification system in a fragmented manner. Parameter of security has an incredible effect over CGA era time. This would make the computation of CGA's unreasonable. Satisfying the condition of hash2 parameter computation is lavish piece of the CGA era. We introduced a useful and programmed path to select parameter of security related to CGA calculation. With this adjusted rendition, time is assumed an information after that the parameter of security worth would be resolved as a yield of beast power hunt so as to fulfill Hash2 quality. Level of security is resolved consequently in view of the processing gadget CPU power accessible for hash era. Quicker gadgets have the capacity to locate a superior Sec quality as compare to sluggish gadgets. We exhibited a handy and programmed path to select parameter of security to calculate CGA era. In adjusted rendition, parameter of time is considered as info also after that the parameter of security quality will be resolved to yield the beast power inquiry so as to fulfill the condition of Hash2. Level of security will be resolved consequently in view of the processing gadget CPU power accessible for hash era. Quicker gadgets have the capacity to locate a superior Sec quality than slower ones for the same time.

5.2. Future Work

In a mobile environment, minimizing the time taken by CGA generation and verification algorithm is vital. This is for two reasons. Firstly, handover operations have to be completed within a few milliseconds in order to ensure an adequate quality of service. Secondly, mobile nodes have limited resources (like Battery, Bandwidth and Memory) that have to be efficiently used to prevent unacceptable delays. It is thus important to review all the work related to factors that affect the time taken by CGA Algorithm. The Factors include Sec value, Hash function (that has a shorter processing time). RSA must be replaced by a public key cryptosystem that provides comparable cryptographic strength but has a faster key generation, shorter key length and less expensive signature generation and verification. Only when this is achieved, CGA-based Authentication can be computationally feasible for mobile environment.

REFERENCES

[1]. T. Aura, "Cryptographically Generated Address", RFC3972, Internet Engineering Task Force, March 2005, http://tools.ietf.org/html/rfc3972

[2]. Hosnieh Rafiee and Christoph Meinel. "Privacy and Security in IPv6 Networks: Challenges and Possible Solutions". The 6th International Conference on Security of Information and Networks (SIN2013),ACM,November 26-28,2013 Aksaray,Turkey

[3]. Narten, T., Draves, R., Krishnan, S.: Privacy Extensions for Stateless Address Auto configuration in IPv6. RFC 4941, Internet Engineering Task Force (September 2007).

[4]. Aura, T.: Cryptographically Generated Addresses (CGA). RFC 3972, Internet Engineering Task Force (March 2005), updated by RFCs 4581, 4982.

[5]. A practical guide to ipv6 network.

[6]. Wikipedia ipv6

[7]. T. Aura, "Cryptographically Generated Addresses (CGA)," in Information Security, vol. 2851, C. Boyd and W. Mao, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 29–43.

[8]. T. Aura, "Cryptographically Generated Addresses (CGA)," RFC 3972, IETF, Mar-2005. [Online]. Available: http://tools.ietf.org/html/rfc3972.

[9]. G. O'Shea and M. Roe, "Child-proof authentication for MIPv6 (CAM)," SIGCOMM Comput. Commun. Rev., vol. 31, no. 2, pp. 4–8, Apr. 2001.

[10]. P. Nikander, "Denial of Service, Address Ownership, and Early Authentication in the IPv6 World," in Security Protocols, vol. 2467, B. Christianson, J. Malcolm, B. Crispo, and M. Roe, Eds. Springer Berlin / Heidelberg, 2002, pp. 22–26.

[11]. G. Montenegro and C. Castelluccia, "Statistically Unique and Cryptographically Verifiable (SUCV) Identifiers and Addresses," in In Proceedings of the 9th Annual Network and Distributed System Security Symposium (NDSS), 2002.

[12]. Barrera, D.; Van Oorschot, P., "Security visualization tools and IPv6 addresses," Visualization for Cyber Security, 2009. VizSec 2009. 6th International Workshop on, vol., no., pp.21, 26, 11-11Oct.2009 doi: 10.1109/ VIZSEC.2009.5375538.

[13]. Jayanthi, J.G.; Rabara, S.A., "IPv6 Addressing Architecture in IPv4 Network," Communication Software and Networks, 2010. ICCSN '10. Second International Conference on, vol., no., pp.461,465,26-28Feb.2010doi: 10.1109/ICCSN.2010.116.