

MANET Autoconfiguration (AUTOCONF)
Internet-Draft
Intended status: Informational
Expires: May 6, 2009

C. Bernardos
M. Calderon
UC3M
H. Moustafa
France Telecom
November 2, 2008

Ad-Hoc IP Autoconfiguration Solution Space Analysis
draft-bernardos-autoconf-solution-space-02

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on May 6, 2009.

Abstract

This draft aims at analysing the solution space for the ad hoc IP autoconfiguration problem, based on the problem statement draft and the MANET architecture draft. Some evaluation considerations, are also taken into account. This draft classifies, at a generic level, the solution space of the possible approaches that could be followed to solve the IPv6 autoconfiguration for MANETs problem. The various approaches of IPv6 autoconfiguration for MANETs are illustrated, and the benefits and tradeoffs in different aspects of IPv6 autoconfiguration are explored.

Internet-Draft

AUTOCONF Solution Space

November 2008

Table of Contents

1.	Introduction	3
2.	Scenarios	3
3.	Issues of IP autoconfiguration in MANETs	4
3.1.	Additional signalling overhead	4
3.2.	Increased protocol complexity and processing load	5
3.3.	Scalability	5
3.4.	Security considerations	5
3.5.	Convergence time	5
3.6.	Routing protocol dependency	6
3.7.	IP address space assignment efficiency	7
4.	IP autoconfiguration solution space analysis	7
4.1.	Which entities are involved?	8
4.1.1.	MANET Routers (distributed approach)	8
4.1.2.	MANET Routers and Border Routers	9
4.1.3.	MANET Routers and distributed servers	9
4.1.4.	MANET Routers and centralised server(s) (centralised approach)	10
4.2.	What type of IP delegation: addresses or prefixes?	10
4.3.	How are IP addresses obtained?	11
4.4.	How is IP address uniqueness guaranteed?	12
4.4.1.	How is address uniqueness detection performed?	12
4.4.2.	When address uniqueness detection is performed: pre-service and/or in-service?	14
4.4.3.	How are address conflicts resolved?	15
4.5.	How is signalling performed?	15
4.6.	Are existing protocols modified?	16
4.7.	What are the security considerations?	17
5.	Security Considerations	17
6.	IANA Considerations	17
7.	Acknowledgements	18
8.	References	18
8.1.	Normative References	18
8.2.	Informative References	18
Appendix A.	Change Log	20
	Authors' Addresses	20
	Intellectual Property and Copyright Statements	22

[1.](#) Introduction

Due to the spontaneous nature of mobile ad hoc networks, there is a need to automatically provide IP configuration for MANET Routers, allowing them to establish communications. Each MANET Router needs a local address that can be used for intra-MANET connectivity, and an external address allowing for Internet connectivity. An IP autoconfiguration solution should be able to satisfy the self-deployment requirements of ad hoc networks and should be flexible to accommodate special features for different ad hoc environments. However, the dynamic and random characteristics of MANETs, the type of scenario, and the type of application make it difficult to have one single standard IP autoconfiguration solution.

Based on the MANET architectural concepts presented in [\[1\]](#), the autoconf problem statement draft [\[2\]](#) describes the issues that should be addressed during the development of IP autoconfiguration solutions. Some evaluation considerations for IP autoconfiguration solutions are also presented in [\[3\]](#) highlighting some important behaviours for the different autoconfiguration solutions to help solution developers during design and to help implementers during the choice of autoconfiguration mechanisms. A number of proposed solutions has been made; however, there is no existing classification or standard for these different proposed solutions. The present draft aims at providing an analysis of the solutions space for the current IP autoconfiguration proposed solutions. The draft discusses the different approaches that an IP autoconfiguration solution could take, giving a generic level classification for the solution space. Also, the main key features, benefits and scope for the different IP autoconfiguration approaches are illustrated.

[2.](#) Scenarios

The ad-hoc term is highly overloaded nowadays and it usually comprises many different network architectures, with disparate

characteristics and requirements. As an example, today we can find several instances of ad-hoc networks: Mobile Ad-hoc Networks (MANETs), sensor networks, Vehicular Ad-hoc Networks (VANETs), Wireless Mesh Networks (WMNs), etc. All of them share their multi-hop, unmanaged and decentralised nature, but also present very important differences.

As described in [3], there are many evaluation considerations and characteristics that a particular IP autoconfiguration mechanism might present. The requirements that an IP autoconfiguration solution should meet will greatly depend on the application scenarios that are considered. Some representative examples of scenario's

characteristics that may have an impact on the adopted IP autoconfiguration solution include, but are not limited to, the following:

- o Connected vs Standalone MANET. The technical requirements that a connected MANET scenario imposes on the solution (such as delegation of global IPv6 addresses/prefixes, MANET Border Router -- also known as Internet Gateway -- discovery, etc.) differ from the ones posed by a standalone MANET scenario (only local IPv6 addressing is required). It should also be noticed that a transition can take place in the connected scenarios, where they may become standalone scenarios from time to time.
- o Node mobility. The mobility of MANET Routers has also a big impact on the requirements of an IP autoconfiguration solution. For example, in high dynamic environments, solutions should present low convergence time values and should not require a high signalling load in order to deal with MANET topology changes (e.g., a node joining/leaving the network, partitioning and merging), since this would limit the applicability or at least the overall performance of the network.
- o Power consumption. An IP autoconfiguration solution for network deployments composed of battery-limited devices should pay special attention to energy-related issues, such as signalling and processing load.
- o Network size. Solutions that aimed at supporting the IP autoconfiguration of large MANETs in terms of number of participant nodes, should carefully look at issues related to convergence time, signalling load and scalability.

[3.](#) Issues of IP autoconfiguration in MANETs

Although IP autoconfiguration is a necessary step to enable ad-hoc networks to benefit from IP and Internet connectivity, there are some tradeoffs -- posed by the different possible approaches that can be used -- that are worth looking at. This section explores some general issues that may impact -- e.g., in terms of applicability or performance -- an IP autoconfiguration mechanism.

[3.1.](#) Additional signalling overhead

The nodes involved in performing IP autoconfiguration could require to exchange additional signalling messages in order to achieve its goal. The required amount of signalling depends on the particular solution, but it could range from no overhead at all (e.g., passive autoconfiguration), local message exchange/piggybacking, to additional message flooding (that could be limited in scope and/or optimised). The amount of signalling is likely to increase with the number of ad-hoc nodes participating in the autoconfiguration

process. Special care should be taken in order to avoid these overhead scale to unacceptable heights, specially to resource-limited devices with low power and/or processing capacity.

[3.2.](#) Increased protocol complexity and processing load

Due to the multi-hop, unmanaged and decentralised nature that usually characterise ad-hoc networks, it is expected that IP autoconfiguration in MANETs will be more complicated than in infrastructure-based networks using standard IPv6 autoconfiguration protocols (e.g., RFCs 4861 [\[4\]](#), [RFC 4862](#) [\[5\]](#)). Therefore, nodes participating in an IP autoconfiguration mechanism would be more complex than current legacy IPv6 hosts. In addition to this additional complexity, MANET Routers will likely have to bear an increased processing load. Again, this increased protocol complexity and processing load may be overwhelming for certain types of MANET Routers (e.g., limited devices in terms of power and processing capacity).

[3.3.](#) Scalability

IP autoconfiguration solutions will likely make use of additional

signalling and/or require to keep track of the IP prefixes/address that are currently being used within the MANET. This may lead to scalability issues, especially in the case of centralised solutions, in which a single node (or a reduced set) play a special role in the IP autoconfiguration process.

[3.4.](#) Security considerations

Depending on the particular scenario, it might be possible that MANET Routers do not belong to the same administrative domain, and therefore it is not possible to assume the existence of security associations between participants. For this reason, the security protection of IP autoconfiguration mechanisms could be harder than that of standard IPv6 autoconfiguration mechanisms (based for example on SeND [6]) or it could be accepted that a "weaker" protection is provided in these environments.

[3.5.](#) Convergence time

The convergence time of a MANET Router is the time required to have a unique IPv6 address since it joins the MANET or a address conflict is detected (e.g., a duplicated), while the convergence time of the whole MANET is the time required to have all its nodes configured with the correct addresses. The convergence time of an IP autoconfiguration mechanism depends on the MANET scenario and the application type, and hence can greatly impact the mechanism's

efficiency. Long convergence times may be expected when there is a sudden large scale change in the structure of the network. On the other hand, as the density of the network increases and the mobility decreases, the convergence time may become shorter.

The convergence time can also differ according to the type of the IP autoconfiguration mechanism. For example, in IP autoconfiguration mechanisms based on signalling flooding or employing periodic procedures, the convergence time can highly impact the mechanisms' scalability, especially for high mobility scenarios. In IP autoconfiguration mechanisms, in which MANET Routers request IP addresses firstly for joining the MANET and then whenever needed, the convergence time has lesser impact on the mechanisms' scalability.

[3.6.](#) Routing protocol dependency

IP autoconfiguration mechanisms require special signalling in order to allow each node to be assigned a usable and unique IP address. Such signalling constitutes additional overhead, as mentioned in [Section 3.1](#). Depending on the amount of signalling, which itself depends on the MANET scenario, the convergence time of IP autoconfiguration mechanisms and hence their scalability can be impacted.

Consequently, one approach is to encapsulate the IP autoconfiguration signalling into the routing protocol messages, where in this case IP autoconfiguration mechanisms can be either routing protocol dependent or routing protocol partially dependent. The former necessitates the existence of a particular routing protocol in order to function, where the IP autoconfiguration mechanism in this case is considered to be tailored to a specific routing protocol (for instance, extending an existing routing protocol to support IP autoconfiguration). In the latter, the IP autoconfiguration mechanism needs the routing protocol messages in order to transfer its signalling, but is adaptive to any existing routing protocol (especially when there are no special constraints in the IP autoconfiguration mechanism, for instance, periodic messages exchange, proactive approach, reactive approach).

On the other hand, IP autoconfiguration mechanisms can be routing protocol independent, where in such case they do not consider at all the existence of any particular routing protocol, however they function in a complete independent manner. Although the autoconfiguration mechanisms function in an independent manner of the routing protocol in this case, it may happen that the routing protocol messages can be used (if a routing protocol is found) for optimised functioning of autoconfiguration mechanisms.

[3.7](#). IP address space assignment efficiency

IP autoconfiguration mechanisms have different approaches for the IP address space assignment, which in turn leads to different IP assignment techniques. One approach is to keep the existing address space centralised for all MANET Routers. Another approach is to split the existing IP address space among the MANET Routers. The first approach can suffer from IP address waste if the information

about addresses' release is not updated frequently by MANET Routers. Even the process of IP address release synchronisation in this case can require considerable exchange of messages between MANET Routers and thus scalability can be impacted, especially for large scale MANETs scenarios. The second approach is useful in the sense of being fully distributed; however it can be less efficient in some MANET scenarios especially those having frequent topology changes, where some nodes may suffer from address space leak while others may have address space waste. Consequently, this may impact the convergence time and hence the scalability of IP autoconfiguration mechanisms.

A third approach appears to be independent of any address space assignment, where each node derives its IP address, for instance using special stateful functions or using its subnet prefix and the EUI-64 interface address. This approach seems interesting in the sense of not taking care of any address space assignment process, however, address uniqueness should be assured and the process of subnet prefix assignment should not cause much signalling.

[4.](#) IP autoconfiguration solution space analysis

There are various different approaches to enable IP autoconfiguration in ad-hoc networks [\[7\]](#). In this section, we attempt to analyse the vast solution space of MANET IP autoconfiguration by asking the following questions:

1. Which entities are involved?
2. What type of IP delegation: addresses or prefixes?
3. How are IP addresses obtained?
4. How is IP address uniqueness guaranteed?
5. How is signalling performed?
6. What are the security considerations?

7. Are existing protocols modified?

[4.1.](#) Which entities are involved?

There are several combinations of entities involved in IP autoconfiguration process. Below is a list of combinations to be discussed in the following sub-sections:

- o MANET Routers (distributed approach).
- o MANET Routers and Border Routers.
- o MANET Routers and distributed servers.
- o MANET Routers and centralised server(s) (centralised approach).

[4.1.1.](#) MANET Routers (distributed approach)

A MANET can be IP autoconfigured in a fully distributed way, without any nodes having special responsibilities in the IP autoconfiguration process. In this scenario, the responsibility of the task is shared among all the participant nodes.

A possible approach is that each MANET Router chooses randomly an IP address and then checks that there is no address conflict, by asking the other nodes in the MANET (e.g., [8] follows this approach). Additionally, the responsibility of detecting conflicts may be distributed, having all the nodes have the potential to detect conflicts. This can be done, for example, by analysing incoming routing protocol messages and looking for inconsistencies [9], [10], [11].

Some solution assign an IP address pool to every new node that enters into the MANET and as of that moment, this node has the potential to split their own IP pool and assign it to another new node [12] [13] (e.g., all nodes collectively perform the functionality of a DHCP server).

The main advantage of this distributed approach is that the existence of a single point of failure is avoided and, therefore, in general this kind of solution would be more robust and scalable than a centralised approach. On the other hand, the main concern with this approach is that the probability of an address conflict to happen is higher, since there is no server that can assure that two nodes are not assigned the same address.

[4.1.2.](#) MANET Routers and Border Routers

Some solution may involve Border Router(s) (also known as Internet Gateways) playing an active role in the IP autoconfiguration process (besides its role of gateways bounding the MANET and providing connectivity to other routing domains). The most common approach is that Border Routers (BRs) announce within the MANET the global IPv6 prefixes that can be used by MANET Routers in the configuration of their IPv6 addresses [14]. MANET Routers would still play an important role in the IP autoconfiguration process, and may for example be responsible for the detection and resolution of address conflicts.

The main concern with this approach is the need for BRs to be deployed within the MANET. Basically, there are two possibilities:

- o BRs are fixed routers in the infrastructure (they do not present any kind of mobility) that support the attachment of MANET Routers. There could be several application scenarios in which assuming such an existence might be difficult or even impossible.
- o BRs are special MANET Routers with the ability to connect to a different routing domain (e.g., an infrastructure-based network such as the Internet). Due to the node mobility of the MANET, it could be possible that the network gets partitioned with no BR available in one or more partitions, making then impossible for MANET Routers belonging to that partition neither to communicate to the other routing domain (in case of connected MANETs) nor to provide with IP autoconfiguration support to new nodes that may arrive and join the partition (that is, these new nodes will not be able to configure a valid IP address while there is no BR reachable within the network). In case of all BRs of a particular MANET managing the same global IP prefixes, a partition of the network might result in topologically incorrect configurations and/or invalid routes towards MANET Routers.

[4.1.3.](#) MANET Routers and distributed servers

Alternatively, it may be considered the existence of some special nodes within the MANET that participate in the IP autoconfiguration process playing a predominant/special role (leader nodes). These nodes are responsible for parts of the IP autoconfiguration of some other MANET Routers [15] (e.g., by issuing Router Advertisements to nodes within their scope). In some solutions, a hierarchy is established by these special nodes.

The advantage of this approach is that may be easier to avoid address conflicts than in a completely distributed approach, because there

exist a set of servers in charge of assigning IP addresses.
Furthermore, the reliability of the solution -- when compared to a

completely centralised solution (described next) -- is improved, since there is no single point of failure. The main concern with this approach is the need for a mechanism to elect these leader nodes and to coordinate/synchronise them (in case this is required). If the leader node role cannot be played by every node (when requested to behave as leader node), then only specific ones can do it. In this case, the same issues pointed out about Border Routers also apply here.

[4.1.4.](#) MANET Routers and centralised server(s) (centralised approach)

In this case, a centralised server is in charge of the whole IP autoconfiguration process.

Centralised approaches may make use of DHCPv6 [\[16\]](#), for example by deploying a DHCPv6 server (within the infrastructure -- e.g., in case of connected MANETS -- or within the MANET itself) and configuring all MANET Routers as DHCP relays to get to the server when a new node joins the network [\[17\]](#).

Due to the centralised nature of these solutions (i.e. all the IP autoconfiguration information is managed and kept in one single entity), it becomes easier to ensure a correct IP configuration across the MANET (e.g., no duplicate addresses configured). The main concerns with this kind of approach are related to scalability and reliability (the server is a single point of failure). Besides, support of partitioning and merging becomes more complicated and the mobility management in general is not easy.

[4.2.](#) What type of IP delegation: addresses or prefixes?

One important aspect of an IP autoconfiguration mechanism, that usually has a very important impact on the mechanism operation, is the type of IP addressing resources that are delegated to MANET Routers: addresses or prefixes.

Current MANET architecture model [\[1\]](#) basically defines MANET participant nodes as MANET Routers. These MANET Routers, besides having one or more MANET interfaces, may also have non-MANET

interfaces, enabling legacy/non-MANET enabled IPv6 hosts (i.e. hosts not running a MANET routing protocol) to attach to and obtain connectivity from a MANET Router. In this particular scenario, allocating IPv6 prefixes to MANET Routers appears as an important feature to be provided. Most of the first proposals for IP autoconfiguration mechanisms only tackled the address delegation problem, whereas it has been lately when some proposals support also prefix delegation.

It is usually harder to check prefix uniqueness within a MANET than address uniqueness. Because of that, the most straightforward approach to provide prefix allocation is to do it in such a way that it is not needed to perform a prefix duplication check. Some ways of doing that is by using a centralised mechanism (for example, based on DHCPv6 [\[17\]](#)) or by using a generation/delegation approach that guarantees the prefix uniqueness before-hand.

[4.3.](#) How are IP addresses obtained?

This is an important question, since the way an IP address/prefix is obtained may also have an impact on other questions, for example those related to the uniqueness of this address/prefix within the MANET.

One of the goals that IP autoconfiguration mechanisms try to achieve is the efficient provision of valid IP addresses to nodes, requiring as less time as possible. In IPv6, there are several mechanisms currently defined for infrastructure-based networks, and they can be classified basically into two main groups, depending on how the IP address is obtained: a) the IP address is locally selected by the node (stateless autoconfiguration [\[5\]](#)), or b) the IP address is assigned by a DHCPv6 [\[16\]](#) server (stateful autoconfiguration). Additionally, the node is responsible for checking that the obtained candidate IP address is not being used in the subnet. To do that, a mechanism, called Duplicate Address Detection (DAD), has been also standardised [\[5\]](#).

Some of the autoconfiguration mechanisms used by non-MANET IPv6 nodes to choose/generate an IP address can also be considered for the MANET scenario. For example, a MANET Router may generate its addresses based on the EUI-64 mechanism [\[5\]](#). This approach has two main

advantages: by re-using the same solution that has been defined for IPv6 infrastructure-based networks, the implementation of the mechanism becomes easier. Additionally, the EUI-64 procedure provides certain guarantees on the global uniqueness of the generated IPv6 address (basically, if the IEEE MAC addresses were globally unique -- which is almost true in most cases -- the EUI-64 generated IPv6 addresses would be globally unique).

An alternative approach, used by several ad-hoc IP autoconfiguration mechanisms (e.g., [8]), consists in generating a random IPv6 address out of a known prefix. This solution has the advantage of being quite simple, but special care should be taken in the implementation of the random generator, since a bad/limited one may lead to different nodes choosing the same IPv6 addresses. This could be an issue in resource-limited devices, where the implementation of a good random number generator could be hard/difficult.

Other solutions may make use of address pools, from which nodes may take IP addresses from. These pools can also be distributed within the ad-hoc network -- for example, hierarchically, by using a binary split approach [12] -- providing also certain address uniqueness guarantees. On the other hand, the management of these address pools may be complicated, especially in environments that present high mobility patterns.

Alternative ways of IP address generation can also be considered, for example those that embed certain information on the IP address. This is the case for example of the Cryptographic Generated Address (CGAs) [18], for which the interface identifier is generated by computing a cryptographic one-way hash function from the node's public key and the IPv6 prefix (among other additional information).

An additional aspect that might be also worthwhile being tackled is the address space distribution within the MANET (already briefly discussed when we described the address pool distribution). Basically, in IPv6 infrastructure-based networks, nodes are attached to subnets, where all the attached nodes share the same prefix(es). In ad-hoc networks, given its multi-hop nature, this is not the only model that can be considered (that is, all the MANET Routers within a MANET configuring their IPv6 addresses from the same prefix). We may consider for example the distribution of different IPv6 prefixes within the MANET, so different MANET Routers may configure addresses

from disjoint IPv6 prefixes. How these prefixes are distributed may be based on different aspects, such as the geographic location of the node, its relative position and distance from a MANET Border Router, its position within a particular hierarchy, etc. The extreme case of this prefix distribution approach is the delegation of a different IPv6 prefix per MANET Router.

[4.4.](#) How is IP address uniqueness guaranteed?

This question actually encompasses the three different important ones, to be discussed in the following sub-sections:

- o How is address uniqueness detection performed?
- o When address uniqueness detection is performed: pre-service and/or in-service?
- o How are address conflicts resolved?

[4.4.1.](#) How is address uniqueness detection performed?

It should be noted that a Non-unique Address Detection mechanism is not always needed, since some methods of obtaining/generating

addresses (see section [Section 4.3](#)) ensure the uniqueness of the assigned addresses/prefixes. This is the case when a centralised approach is followed (e.g., a DHCPv6 server) which keeps an up-to-date list of assigned/available addresses, or when a set of coordinated servers is deployed -- collectively perform the DHCP functionality --. For example, a new MANET Router may take IP addresses from a set of address pools (a disjoint set of available IP addresses) distributed within the ad-hoc network -- for example, hierarchically, by using a binary split approach [\[13\]](#) -- and nodes owning a pool which collectively perform the DHCP functionality, are somehow coordinated to assure the pools are collision-free. Additionally, there exist some methods of address generation which ensure the uniqueness of assigned addresses (e.g., a special type of function is used to generate a series of random numbers -- IPv6 addresses -- [\[19\]](#)).

On the other hand, some methods of obtaining/generating addresses do not ensure the uniqueness of the assigned addresses/prefixes (e.g.,

each MANET Router generates a random IPv6 address out of a known prefix [8]). In these cases, mechanisms to detect and resolve address conflicts are needed.

A first approach to detect address conflicts is to check that the tentative address (e.g., randomly chosen) is not being used by another node in the network. A first possibility is that each MANET Router, before using a tentative address, floods an Address Request message [8] in the network to query for the usage of this address. The node waits for a while after sending this query for the reception of a reply. The process is repeated if no answer is received, and if after a number of attempts no reply has been received, the node assumes that the tentatively chosen IPv6 address is unique and starts using it. A main concern with this approach is its scalability which is strongly correlated with the organisation of the network, i.e. a flat structure, or a hierarchical one. If the former case, every address acquisition results in extra traffic throughout the whole network; however, only group leaders/representatives need to take action in the latter case [15]. Additionally, this approach is not applicable in networks where message delays cannot be bounded (e.g., the use of timeouts cannot reliably detect absence of a message).

Another possibility is that before choosing a tentative address a positive acknowledgement (ACK) is required from all known nodes in the network [20]. This approach requires each MANET Router to maintain an up-to-date list of all the nodes in the network. This list can additionally be used to detect partitions in the network (e.g., if a set of nodes do not reply, it could be due to a partition). This approach has several significant drawbacks: its scalability, the cost of updating this list of participants in highly

dynamic scenarios, and it is not applicable in networks where message delays cannot be bounded -- which is a likely occurrence in dynamic ad hoc networks -- because of its use of timeouts (e.g., the absence of a message could be misinterpreted as a partition).

Another plausible approach is to relax the requirement of avoiding duplicate addresses and focus on preventing a packet from being routed to a wrong destination even if an address conflict exists [21]. For example, a unique per MANET Router key is included in the routing control packets and in the routing table entries. So, every node is identified by a unique tuple <address, key> (e.g., virtual IP

address). Following this approach, even if two nodes happen to have chosen the same IP address, they can still be identified by the use of their unique keys. The main concern with this approach is that it implies to modify current routing protocols.

Another way of addressing the detection of duplicate addresses events is looking for the consequences/results of a potential address conflict. This can be accomplished passively by continuously monitoring routing protocol traffic (e.g., looking for inconsistencies) [9]. The basic idea of this approach is to exploit the fact that some protocol events occur in case of duplicate address, but (almost) never in case of a unique address. Most of the existing solutions following this approach work with proactive routing protocols (i.e. OLSR) but it can also be applied to on-demand routing protocols [10]. The main advantages of this approach are that it can work with current routing protocols (e.g., without any modifications) and it does not introduce any extra overhead to perform address uniqueness detection. On the other hand, the time needed to detect conflicts may be high and during this time a MANET Router may be experiencing deficiencies in their communications.

4.4.2. When address uniqueness detection is performed: pre-service and/or in-service?

Address uniqueness detection may be needed at different times of the communication. A first situation is when a MANET Router has just chosen a tentative address and, before assigning it to the interface and using it, it is checked that there is not an address conflict (i.e. pre-service detection).

Additionally address conflicts may occur at any time, mainly caused by mergers and partitions in the MANET. It may happen that nodes in different networks/partitions may independently obtain the same address, and duplicate addresses result if these networks merge later. Thus, the address uniqueness detection may be needed to take place in a continuous manner during the whole life of the MANET (in-service detection) [2].

Generally speaking, address uniqueness detection approaches commented above could be used both as pre-service and as in-service mechanisms [15] [21] [22]. Nevertheless, some of them seem to be more appropriate for just one of the situations (pre-service or in-

service). For instance, querying the rest of MANET Routers to check whether an IP address is available or not, seems to be more acceptable in the case of pre-service detection (e.g., flooding is not repeated periodically over the time). In general the overhead introduced by the mechanism is going to be a more critical issue in the case of in-service than in the case of pre-service detection. So, approaches that analyse routing protocol messages looking for inconsistencies (e.g., [21]) or uniquely identify nodes in routing protocol messages (e.g., [19]) without adding extra messages seem to be more suitable for the case of in-service detection.

[4.4.3.](#) How are address conflicts resolved?

Whenever an address conflict is detected the most common approach is to use a heuristic to decide which MANET Router keeps on using the duplicated address and which one has to look for a new IP address. In the case of pre-service detection the solution is quite straightforward the newcomer (e.g., trying to use an already-assigned address) has to look for a new address. However if the conflict has been caused by a merging (in-service detection) different heuristic can be used (e.g., the node that detects the conflict keeps on using the duplicated address [22]).

An interesting issue to be addressed is what happens in the event of an address conflict while the node has an ongoing session. Session continuity should be guaranteed after an address duplication episode. One possible way of ensuring session continuity is IP tunnelling data packets to the new assigned address (e.g., The MANET Router, keeping on using the duplicated address, tunnels packets to the other MANET Router [22]).

[4.5.](#) How is signalling performed?

In general, the IP configuration mechanism requires some extra signalling -- additional to the signalling introduced by the ad-hoc routing -- in order to reach its goal. The ways signalling is performed may have an impact on the scalability and convergence time of the IP autoconfiguration mechanism.

This extra signalling may be sent as separate messages or may be added/piggybacked to existing routing protocol messages (e.g., prefix or Border Router Information). The size of this added overhead and its periodicity may vary on the different solutions. The main concern of adding/piggybacking signalling information to the existing

routing protocol messages is that the IP autoconfiguration mechanism is routing protocol dependant (see [Section 4.6](#)). On the other hand, the main advantages of this approach are that the IP autoconfiguration mechanism may somehow take advantage of the routing discovery phase of the ad-hoc routing protocol (e.g., discovering of available prefix and border routers) and do not introduce extra messages (e.g., MANET Routers have to process less messages).

Flooding the MANET with signalling messages is required by some mechanisms, for example asking for the approval of the rest of the nodes with each new address acquisition. There exist different ways of decreasing the effects of flooding such as limiting the scope, for example organising the network in a hierarchical structure, where only group leaders need to take action with each new address acquisition.

An alternative approach consists on relying -- partially or totally -- on ad-hoc routing signalling to perform IP autoconfiguration. An example of this approach is passive address uniqueness detection [[15](#)] where conflicts are identified by analysing received routing protocol messages and detecting inconsistencies. The main advantage of this approach is that no extra signalling is introduced in the network and the routing protocol is used as-is (e.g., without modifications), on the other hand this kind of mechanism are routing protocol dependant (e.g., the mechanism may be quite different for each particular routing protocol).

[4.6](#). Are existing protocols modified?

IP autoconfiguration mechanisms should function in a compatible manner to the other underlying protocols; however some of these protocols can be modified or extended in order to allow the proper IP autoconfiguration mechanisms' functioning and signalling transfer. Autoconfiguration mechanisms can extend the IPv6 Neighbour Discovery Protocol (NDP) to work in multi-hop wireless networks (for instance extending the NDP router solicitation and advertisement messages), or employ ICMPv6 messages in a modified manner. Also, some mechanisms can modify DHCP protocol, allowing MANET Routers to act as modified DHCP proxies.

As discussed in Section [Section 3.6](#), IP autoconfiguration mechanisms can use the routing protocol messages to transfer the IP autoconfiguration signalling. This takes place whether by simply encapsulating such signalling in routing protocols control messages with no routing protocols modification, or through adding new control messages to the routing protocol ones. In the former, IP autoconfiguration mechanisms are mostly open to any existing routing

protocol, proactive or reactive according to their functioning mode.

While in the latter, IP autoconfiguration mechanisms extend the functioning of a given routing protocol to support IP autoconfiguration, which in turn limits their application scope.

[4.7.](#) What are the security considerations?

The wireless ad hoc environment attacks can lead to improper functioning of autoconfiguration mechanisms. Nevertheless, the IP autoconfiguration mechanisms proposed so far do not propose special mechanisms to secure the address autoconfiguration process.

Assuring reliable IP autoconfiguration mechanisms' signalling (i.e. secure transfer) is critical for the proper functionality of any IP autoconfiguration mechanism. In this sense secure communication should be assured between MANET Routers, where this problem can be differently treated according to the approach used by the IP autoconfiguration mechanism. For IP autoconfiguration mechanisms depending on the routing protocol, this can be done through securing the routing protocol (especially, the control messages' transfer). However, IP autoconfiguration mechanisms that are routing protocol independent needs special security mechanisms. In spite of the type of IP autoconfiguration mechanism (routing protocol dependent or not), cooperation between nodes is an important factor in order to assure the proper messages' (signalling) forwarding during the autoconfiguration process.

Generally, security considerations can differ depending on different MANET scenarios, where connected MANETs allows to have a central authority that can play the role of a trusted third party to authenticate MANET Routers for example or provide cryptographic keys.

[5.](#) Security Considerations

This draft highlights the security considerations issue during the analysis of the solution space of MANET IP autoconfiguration; however no special security mechanisms are given. The autoconfiguration problem statement draft [\[2\]](#) states some important security issues that worth considerations. Also, the IP evaluation considerations draft [\[3\]](#) discusses the issue of trust and security in order to

assure the proper functioning of IP autoconfiguration solutions.

[6.](#) IANA Considerations

This document has no actions for IANA.

Bernardos, et al.

Expires May 6, 2009

[Page 17]

Internet-Draft

AUTOCONF Solution Space

November 2008

[7.](#) Acknowledgements

The structure and rationale of this I-D has been greatly inspired by [RFC 4889](#) [23].

The work of Carlos J. Bernardos and Maria Calderon has been partially supported by the Spanish Government under the POSEIDON (TSI2006-12507-C03-01) project.

The work of Carlos J. Bernardos and Maria Calderon has also been partially supported by the EU through the ICT FP7 European Project CARMEN (INFSO-ICT-214994). Apart from this, the European Commission has no responsibility for the content of this Internet-Draft. The information in this document is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

[8.](#) References

[8.1.](#) Normative References

- [1] Chakeres, I., Macker, J., and T. Clausen, "Mobile Ad hoc Network Architecture", [draft-ietf-autoconf-manetarch-07](#) (work in progress), November 2007.
- [2] Baccelli, E., Mase, K., Ruffino, S., and S. Singh, "Address Autoconfiguration for MANET: Terminology and Problem Statement", [draft-ietf-autoconf-statement-04](#) (work in progress), February 2008.

[8.2.](#) Informative References

- [3] Moustafa, H., Bernardos, C., and M. Calderon, "Evaluation Considerations for IP Autoconfiguration Mechanisms in MANETs", [draft-bernardos-autoconf-evaluation-considerations-03](#) (work in progress), November 2008.
- [4] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", [RFC 4861](#), September 2007.
- [5] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", [RFC 4862](#), September 2007.
- [6] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", [RFC 3971](#), March 2005.

- [7] Bernardos, C., Calderon, M., and H. Moustafa, "Survey of IP address autoconfiguration mechanisms for MANETs", [draft-bernardos-manet-autoconf-survey-04](#) (work in progress), November 2008.
- [8] Perkins, C., "IP Address Autoconfiguration for Ad Hoc Networks", [draft-perkins-manet-autoconf-01](#) (work in progress), November 2001.
- [9] Weniger, K., "PACMAN: Passive autoconfiguration for mobile ad hoc networks", IEEE Journal on Selected Areas in Communications, vol. 23, no. 3, Mar 2005 pp. 507-519 , 2005.
- [10] Jeong, H., "Passive Duplicate Address Detection for On-demand Routing Protocols", [draft-jeong-autoconf-pdad-on-demand-01](#) (work in progress), April 2007.
- [11] Mase, K. and C. Adjih, "No Overhead Autoconfiguration OLSR", [draft-mase-manet-autoconf-noaolsr-01](#) (work in progress), April 2006.
- [12] Mohsin, M. and R. Prakash, "IP Address Assignment in a Mobile Ad Hoc Network", MILCOM 2002 , 2002.
- [13] Tayal, A. and L. Patnaik, "An address assignment for the automatic configuration of mobile ad hoc networks", Personal

Ubiquitous Computing , 2004.

- [14] Ruffino, S. and P. Stupar, "Automatic configuration of IPv6 addresses for MANET with multiple gateways (AMG)", [draft-ruffino-manet-autoconf-multigw-03](#) (work in progress), June 2006.
- [15] Weniger, K. and M. Zitterbart, "IPv6 Autoconfiguration in Large Scale Mobile Ad-Hoc Networks", European Wireless 2002 , 2002.
- [16] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", [RFC 3315](#), July 2003.
- [17] Templin, F., "Virtual Enterprise Traversal (VET)", [draft-templin-autoconf-dhcp-20](#) (work in progress), October 2008.
- [18] Aura, T., "Cryptographically Generated Addresses (CGA)", [RFC 3972](#), March 2005.
- [19] Zhou, H., Ni, L., and M. Mutka, "Prophet Address Allocation for

Large Scale MANETs", Proceedings of INFOCOM 2003 , 2003.

- [20] Nesargi, S. and R. Prakash, "MANETconf: Configuration of hosts in a mobile ad hoc network", INFOCOM 2002 , 2002.
- [21] Vaidya, N., "Weak Duplicate Address Detection in Mobile Ad Hoc Networks", MOBIHOC'02 , 2002.
- [22] Jeong, J., "Ad Hoc IP Address Autoconfiguration", [draft-jeong-adhoc-ip-addr-autoconf-06](#) (work in progress), January 2006.
- [23] Ng, C., Zhao, F., Watari, M., and P. Thubert, "Network Mobility Route Optimization Solution Space Analysis", [RFC 4889](#), July 2007.

Changes from -01 to -02:

- o New release to keep the document alive.
- o Update of some references.

Changes from -00 to -01:

- o New release to keep the document alive.
- o Update of some references.

Authors' Addresses

Carlos J. Bernardos
Universidad Carlos III de Madrid
Av. Universidad, 30
Leganes, Madrid 28911
Spain

Phone: +34 91624 6236
Email: cjbc@it.uc3m.es

Bernardos, et al.

Expires May 6, 2009

[Page 20]

Internet-Draft

AUTOCONF Solution Space

November 2008

Maria Calderon
Universidad Carlos III de Madrid
Av. Universidad, 30
Leganes, Madrid 28911
Spain

Phone: +34 91624 8780
Email: maria@it.uc3m.es

Hassnaa Moustafa
France Telecom

38-40 rue du General Leclerc
Issy Les Moulineaux 92794 Cedex 9
France

Phone: +33 145296389

Email: hassnaa.moustafa@orange-ftgroup.com

contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.