Network Working Group
Internet-Draft
Expires: July 2, 2005

JJ. Puig
M. Achemlal
E. Jones
D. McPherson

January 2005

Generic Security Requirements for Routing Protocols
draft-ietf-rpsec-generic-requirements-01

Status of this Memo

By submitting this Internet-Draft, I certify that any applicable patent or other IPR claims of which I am aware have been disclosed, and any of which I become aware will be disclosed, in accordance with RFC 3668.

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/1id-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 July 2, 2005.

Copyright Notice

Abstract

Routing protocols are subject to threats and attacks that can harm individual users or network operations as a whole. This document describes a generic set of security requirements for routing protocols and routing systems.

Table of Contents

1. Introduction 4

2. Terminology 5
2.1 Path 5
2.2 Destination 5
2.3 Route Property / Route Attribute (RP / RA) 5
2.4 Forwarders 5
2.5 Neighbors / Peers / Adjacent Routers 5
2.6 Propagators 5
2.7 Route / Routing Information (RI) 6
2.8 Route correctness 6
2.9 Route validity 6
2.10 Routing Function 7
2.11 Routing Decision Process 7
2.12 Forwarding Function 7

3. General Requirements 8

4. Threats Importance 9
4.1 Threats Consequences 9

5. Security Requirements 11
5.1 Requirements Against Overclaiming 11
5.2 Requirements Against Misclaiming 13
5.3 Requirements Against Misstatement 14
5.4 Requirements Against Spoofing 17
5.5 Requirements Against Overload 18
5.6 Requirements Against Interference 19
5.7 Requirements Against Deliberate Exposure 21
5.8 Requirements Against Sniffing 21
5.9 Requirements Against Traffic Analysis 22

6. Living with Byzantine Failures 24
6.1 The Byzantine Problem 24
6.2 Byzantine General Requirements 24
6.3 Detection of the Occurrence of a Byzantine Failure 25
6.4 Byzantine Detection 25
6.5 Byzantine Robustness 26

7. Security Techniques for Routing 27
7.1 Techniques when Originating 27
7.2 Techniques when Propagating 29
7.3 Security of the Functional Parts 31
7.4 Date and Time Issues 34

8. Local Security 35

8.1 Active Participation to Security 35
8.2 Local Resources Considerations 36

9. Inter-Domain Routing Issues 40
9.1 Legitimacy 40
9.2 Propagating policies 41
9.3 Coherence 41
9.4 Confidentiality 41
9.5 Agreements involving operators 41

10. Security Considerations 43

11. References 44
11.1 Normative References 44
11.2 Informative References 44

Authors' Addresses 45

A. Acknowledgments 46

B. Revision History 47
B.1 Changes from draft-puig-rpsec-generic-requirements-02 . . 47
B.2 Changes from draft-puig-rpsec-generic-requirements-01 . . 47
B.3 Changes from draft-puig-rpsec-generic-requirements-00 . . 47

Intellectual Property and Copyright Statements 48

1. Introduction

Routing protocols are subject to threats and attacks that can harm individual users or network operations as a whole. This document describes a generic set of security requirements for routing protocols and routing systems.

Along with the "Generic Threats to Routing Protocols" document [THREATS], this work is designed to serve as a reference material for current routing protocols and routing systems analysis, for extensions design, and as a guidance for designing new, more secure, routing protocols and routing systems.

This document discusses generic requirements for routing protocols, used for both interdomain and intradomain routing. Host to router and multicast routing protocols, specifically, are out of scope.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [KEYWORDS].

Security terms are explained in [SEC-GLOSS].

This document has the following layout:

- Section 2 defines terms used within this document.
- Section 3 presents general requirements to the security of routing protocols and routing systems.
- Section 4 sorts by importance threats defined in [THREATS].
- Section 5 defines generic security requirements.
- Section 6 provides guidance for tackling the Byzantine problem.
- Section 7 describes techniques for routing security.
- Section 8 presents security considerations for the routing device.
- Section 9 introduces the inter-domain puzzle.

2. Terminology

The following terms will be used in this document:

2.1 Path

A path is a list of successive forwarders through which a destination may be reached.

2.2 Destination

Destination refers here either to a physical network or to a prefix, as announced in the routing protocol. Thus destination is only a 'hint' about eventual destination of user's traffic.

2.3 Route Property / Route Attribute (RP / RA)

Routing protocols also distribute information associated with the destination (ex: number of hops). A minimum set of such properties may be mandatory in order to avoid loops within the routing system.

Any router, while propagating routing information, may add, remove or update route attributes.

2.4 Forwarders

Forwarders are either mentioned as a route attribute, or assumed to be the peers through which the routes were acquired.

Forwarders are expected to be the first elements of paths providing packet forwarding to the destination mentioned in the route, with the associated properties.

2.5 Neighbors / Peers / Adjacent Routers

The terms "neighbors", "peers", "adjacent routers" all refer to routers which can communicate directly over the transport subsystem.

2.6 Propagators

Propagators are participants to the routing protocol. They relay the routing information between networks. They may also modify the routing information before relaying (/ flooding / re-advertising) it.

Propagators are usually set on transit networks, not on stub networks.

2.7 Route / Routing Information (RI)

Routing protocols enable routers to exchange routes or routing information.

Routes include at least the description of a destination and associated route properties. At least one forwarder is mentioned in -or associated with- the route.

2.8 Route correctness

A "correct" route is such that:

1. it exists at least one path to the destination from each system listed as a suitable forwarder.

2. (locally known) route properties are compatible with local mandatory policy requirements for the destination.

Note: This definition of correctness is local to this document; a minimal definition would only address the 2nd issue, while a stronger one would also require knowledge that route properties are consistent with actual paths properties.

According to this definition, there is no way to be sure a route is correct or not when routing decision must be taken. Thus, we will consider that a route is correct when the router has been 'convinced' of this correctness.

This 'conviction' should result from a trust measure on the way the route was acquired (ex: route signed by a business partner, static input from the management plane).

2.9 Route validity

A "valid" route is such that:

- packets forwarded to any adequate forwarder follow a path up to destination.

- (locally known) route properties are honored on the path and compatible with local mandatory policy requirements.

Note: According to this definition of route validity, there is no way to know a route is valid when routing decision is taken. There is also no certainty of knowing afterward that a route was valid or not. This definition underlines the uncertain nature of any communication though it should not be considered as `petitio principii'.

2.10 Routing Function

In this document, the routing function is a process which returns the available routes for a packet destination.

2.11 Routing Decision Process

In this document, the routing decision process selects a route from the set returned by the routing function or cancels forwarding.

Many criteria may influence selection (ex: packets source). In the field, respective rules of the routing function and of the routing decision process may be less strict.

2.12 Forwarding Function

The forwarding function performs actual forwarding according to data installed by the routing decision process.

3. General Requirements

Routing protocols are responsible for distributing information about reachability to destinations attached to the network.

First main requirement is:

MR(1) Correct routes SHOULD be made available for the routing function.

MR(1.a) When a route is unavailable or incorrect, the first main requirement means that a correct route SHOULD be made available, either through the use of the routing protocol or from the management plane.

MR(1.b) When a route is available and correct, the first main requirement means that misuse of the routing protocol SHOULD NOT jeopardize the route availability or correctness, as this would also compromise correct routing.

Note: the first main requirement does not preclude acquisition of routes whose correctness cannot be established, if available. It only sets a priority to both availability and correctness of routes.

Second main requirement is:

MR(2) The routing function MUST recognize and select correct routes if available for the packet properties. If such routes are unavailable or partly incorrect, severed routing processing MAY be investigated, according to a heuristic learnt from the management plane. Eventually, if it is decided that no forwarding will be achieved, the packet MUST be discarded or rejected according to local policy (this SHOULD be configurable).

MR(2.a) Packet properties analyzed by the decision process MAY include other information than destination address.

Note: the second main requirement does not preclude forwarding when full correctness or availability of routes cannot be achieved. It also focuses more on the eventual forwarding than on routing.

Routing protocols functions and misuses are documented in [THREATS].

Most (but not all) subsequent requirements are meant to raise the confidence that correct routes are available when required by the routing function.

4. Threats Importance

In the [THREATS] document, threats are described according to their sources, their consequences, and eventually the behaviors -referred as "actions"- which enable sources to trigger consequences.

4.1 Threats Consequences

In an economical perspective, primary concern is about the consequences and their potentiality for damages. We will elaborate requirements according to the following classification of consequences, sorted by importance order:

i - Usurpation. Damages cost resulting from usurpation may be extreme and may only be roughly estimated. Besides, usurpation often enables the attacker to proceed with subsequent consequences. For these reasons, usurpation is the top issue.

ii - Deception. Deception will partly result in the same damages as usurpation and is thus an important consequence.

iii - Disruption. Disruption is a significant consequence, but its range and period are usually limited and damages cost can be evaluated more accurately than for previous consequences. However, actions leading to disruption should be difficult enough to achieve so that disruption does not become a common event. Beyond a certain threshold (depending on frequency, duration, range and overall context), disruption may become more significant than usurpation or deception.

iv - Disclosure. The above consequences directly jeopardize the services expected to be provided by the routing system. Reliability and availability of the routing system is usually considered more important than confidentiality of the routing information (which is not `user data' per se and may be learnt by other means). In current protocols, it is unlikely that disclosure of routing information will lead to direct damages on routing services as a result of the information leak. In this context, concealing the services properties in order to protect against disclosure is not a priority. However, it is worth preventing against disclosure of information which would enable the attacker to trigger usurpation, deception or disruption (in-band plain text passwords are likely to be such pieces of information).

Security requirements deal with prevention against the conditions of consequences. This prevention may be against the existence of threat sources or against the occurrence of threat actions (attacks).

Subversion of the routing system may be very easy if an attacker has the ability to attack the physical link between two routers, or individual routers. Techniques that harden any of these points of attack can make attacking the routing system harder, but we consider these outside the scope of this document.

We are thus primarily interested in the threat actions that result in usurpation, secondarily in those that result in deception, thirdly in disruption, lastly in disclosure.

5. Security Requirements

In this section, we explore the requirements which will help in tackling the actions leading to the consequences of concern. First set of requirements addresses prevention against usurpation.

5.1 Requirements Against Overclaiming

"Overclaiming occurs when a subverted router advertises its control of some network resources, while in reality it does not, or the advertisement is not authorized" [THREATS].

Overclaiming is a threat from an originating router; it affects the data plane of the routing protocol.

Main issue with overclaiming is checking resources control or advertisement authorization. Several models may be designed to counter overclaiming; these models address the delegation and the authorization of network resources ownership, control and advertisement.

Delegation allows for an entity to delegate a property in part or entirely to another entity (ex: owner of some network resources delegates ownership of a part of resources to another entity, which in turn becomes owner of this part).

Authorization allows for an entity to grant rights on network resources. An owner of some network resources grants control of resources to a controller; A controller of some network resources grants authorization of advertisement to an advertiser.

In the field, depending on the context and on the instance of the routing protocol, status of owner, controller and advertiser does not necessarily imply separate entities. The same entity may own and control the resources; the same device may have been granted control and advertisement.

Whatever the model representation, a chain of variable length involving delegation and authorization of some network resources ownership, control and advertisement may exist. Overclaiming is a violation of the logic stated in this chain for these specific resources.

However, such a logic is also limited:

- In specific contexts (ex: inter-domain routing, ad-hoc routing), giving birth to such a chain for any network resource may be much too complex (it may yet be possible on a limited basis).

- The destination announced may get modified by routers during normal routing operations (ex: re-advertising with a shorter prefix); this allows for reducing the overall amount of routing information and to lessen its impact on bandwidth, storage and on routing function algorithms. On the other hand, this also legitimates overclaiming, and once original information is lost, it gets far harder to provide any trust measure regarding the new routing information. In such a situation, the usefulness of the chain is unclear because it may or may not state anything about the new super-set of network resources.

For these reasons, subsequent requirements are limited to the case in which the chain exists. The nature (hierarchical, relational, etc.) of the architectural scheme from which the chain is extracted is outside the scope of this documentation. Furthermore, the following requirements DO NOT make any statement about what should be done with -out of the chain- unauthenticated information (discard, install with lower preference, use it for the sake of the routing protocol but not for user traffic... cf. Section 7.3.3).

R(1.1) Integrity, data origin authenticity, validity at current date and availability of nodes of the chain of delegation and authorization of a specific resource ownership, control and advertisement MUST be provided when such a chain exists.

This expands to:

R(1.1.a) It MUST be possible to check that a routing device is currently authorized to advertise some network resources.

R(1.1.b) It MUST be possible to check that the entity which (directly or indirectly) granted the right of advertisement actually and currently controls the corresponding network resources.

R(1.1.c) It MUST be possible to check that the entity which (directly or indirectly) granted the control actually and currently owns the corresponding network resources.

R(1.1.d) It MUST be possible to check that delegation between entities is actually and currently valid.

R(1.2) Consumers and propagators of routing information MUST check backward the chain of delegation and authorization of resource advertisement, control and ownership.

R(1.2.a) Check depth MUST be sufficient according to the context in which the routing protocol instance is in use and to the locally available information.

Requirement R(1.1.c) MAY be limited to the scope in which the routing protocol instance is in use.

Requirements R(1.1*) imply the use of a time scale for date validity. Further discussion on this topic is presented in Section 7.4

Requirement R(1.2.a) allows for using the same chain at different scales. In internal routing operations, a router will check the chain up to the routing system controller (of which it should already be aware), while in external routing operations, a router will check the entire chain or rely on the knowledge that the check was done by another edge router of the same system. It is also possible to establish several steps of "internal" and "external" routing with regard to this specific topic.

When convergence is achieved,

- if verifiable information is available, overclaiming can be thwarted by the requirement of checking that the routing device is authorized to advertise by an administrative entity which was given control of the according network resources by their owner; in such a context, authenticated information should take precedence over any unauthenticated information.

- if no verifiable information is available, then overclaimed information is all what the device can get. What to do with it is a local policy matter.

Practical considerations related to these requirements are presented in Section 7.1.

Further elements regarding this topic are presented in Section 9.

5.2 Requirements Against Misclaiming

"A misclaiming threat is defined as an action where an attacker is advertising its authorized control of some network resources in a way that is not intended by the authoritative network administrator" [THREATS].

Misclaiming is a threat from an originating router; it affects the data plane of the routing protocol.

In our approach, the authoritative network administrator is a resource controller, higher in the chain of delegation and authorization than the routing device. Misclaiming is a corruption of properties applying to the resources as intended by their controller (cf. Section 5.1 for further information regarding this

chain).

R(2.1) Integrity, data origin authenticity, validity at current date and availability of the properties applying to the advertised resources MUST be provided.

R(2.2) Consumers and propagators of routing information MUST check that properties applying to the advertised resources are effectively related to the resources and as intended by the resources controllers.

Requirements R(1.*) also apply.

Misclaiming is thwarted by the requirement of checking that the properties are tied to the advertised resources and are intended by their controller.

Practical considerations related to these requirements are presented in Section 7.1.

5.3 Requirements Against Misstatement

Misstatement "is defined as an action whereby the attacker describes route attributes in an incorrect manner" [THREATS]. The attacker acts on attributes through deletion, insertion and substitution of data. He may also replay out-dated data.

Misstatement is a threat from subverted links and subverted forwarding devices; it affects the data plane of the routing protocol. However, a message replay may also be considered as a control plane violation.

There is an additional difficulty in cases in which correct operation of the routing protocol requires updates of a set of route attributes. This is a common situation in vector protocols.

We thus define the following classification of route attributes (where route attributes are defined in Section 2):

- Route attributes intended by their originator to be consumed by and to reach adjacent nodes unmodified: "constant, not propagated route attributes".

- Route attributes intended by their originator to keep constant values and to be propagated by adjacent nodes: "constant, propagated route attributes".

- Route attributes intended by their originator to reach adjacent

nodes unmodified and to be propagated after an update: "updateable, propagated route attributes".

Lastly, any router can add or delete route attributes.

5.3.1 Constant, not propagated route attributes

These route attributes must reach adjacent nodes unmodified. Possible attackers are: compromised links, subverted forwarding devices, masquerading routers.

This threat results from a lack of data integrity, data origin authentication and replay protection. Protection of data between adjacent nodes, especially anti-replay, has a tendency to focus on session management and on control plane.

The following requirements CAN be addressed either:

- at the control plane level,
- by the transport subsystem (the preferred way),
- at the data plane level.

A routing protocol design SHOULD mention at which level these requirements are fulfilled:

R(3.1) Evidence of integrity and authenticity of data exchanged between neighbors SHOULD be provided; this evidence SHOULD be dependant on data destination. When the evidence applies on data description (as opposed to applying on a per-message basis), it SHOULD also be dependant on the resource the route attributes apply to.

R(3.1.a) It SHOULD NOT be possible to impersonate a neighbor. That is: authentication of neighbors SHOULD depend on a a-priori knowledge (a public key, a shared secret, knowledge of a direct connection in a common technical room, etc). This dependency MUST be documented in the protocol design.

R(3.2) Upon reception, data integrity and authenticity SHOULD be checked. This check SHOULD also include data destination and, when the check applies directly on data description (as opposed to applying on a per-message basis), that route attributes apply to appropriate resources.

R(3.3) The routing protocol SHOULD be protected against the damages resulting from data replay. This CAN be done either by preventing replays effectiveness (ex: through integrity protected sequence numbers) or by reducing replays incidence on data (ex: through lifetime limitation of data).

Practical considerations related to these requirements are presented in Section 7.2.

5.3.2 Constant, propagated route attributes

These route attributes must be propagated but not updated. Given requirements R(3.[1-3]) above, possible remaining attackers are: subverted propagating devices.

The threat here results from a lack of route attributes integrity, origin authentication and lifetime limitation. When propagated, information protection has a tendency to apply directly at the data plane level.

R(3.4) Integrity, data origin authenticity and validity at current date of constant, propagated route attributes MUST be provided. The evidence MUST depend on the resource the route attributes apply to and on the identity of the node which add these specific route attributes.

R(3.4.a) This CAN be done in such a way that deletion, insertion or substitution of route attributes will invalidate the whole routing information or a set of route attributes when checked.

R(3.5) Consumers and propagators of routing information MUST check that constant, propagated route attributes apply to the resources and are the ones intended by the entity which set them.

R(3.6) Route attributes validity MUST be lifetime limited.

Requirement R(3.4) addresses protection against unauthenticated insertion and substitution, and against partial deletion of a route attribute.

However, full protection against deletion further depends on how route attributes and resources are related, and if propagators are allowed to delete route attributes: this is the scope of requirement R(3.4.a). A design may apply different treatments on route attributes which "must" be propagated and on route attributes which "can" be propagated or discarded along the routing protocol' propagation path. The latter must not invalidate routing information when deleted. In a sense, requirement R(3.4.a) manages a way for

more complex route attributes propagation management. Further considerations regarding this topic are not addressed in this document.

Requirement R(3.6) set a lifetime on the evidences, not on route attributes themselves.

Practical considerations related to these requirements are presented in Section 7.1.

5.3.3 Updateable, propagated route attributes

These route attributes must be propagated and updated. Given requirements R(3.[1-3]) above, possible remaining attackers are: subverted propagating devices.

The threat here results from the absence of any verifiable history of route attributes updates. In the absence of any data trace-ability, it is difficult to figure out if a misstatement occurred.

For this reason, unless route attributes are expanded in such an history and each update meets requirements for constant, propagated route attributes, we can reach to no strong requirement here.

However, depending on the semantic of specific route attributes, routers MAY evaluate whether values are realistic or not.

5.4 Requirements Against Spoofing

"Spoofing occurs when an illegitimate device assumes the identity of a legitimate one" [THREATS].

Spoofing is possible because of a lack of combined integrity and data origin authentication. When considered an attack per se, spoofing is a threat on routing protocol control plane operations. It threatens neighbor relationship formation and state maintenance.

R(4.1) Requirements R(1.*) and R(3.[1-3]) also address spoofing.

R(4.1.a) In the context of spoofing, an emphasis SHOULD be made on the transport subsystem or on the control plane when interpreting requirements R(3.[1-3]).

It is often adequate to elect an appropriate transport subsystem which would provide functionalities against spoofing (cf. Section 7.3.1).

Requirements R(1.*) through R(4.*) aim at preventing against

usurpation and deception. Following requirements address disruption and usurpation.

5.5 Requirements Against Overload

"Overload is defined as a threat action whereby attackers place excess burden on legitimate routers" [THREATS].

Overload is a threat from subverted links or devices. It may affect the data plane of the routing device, eg. as the result of link overload. It may also affect the control plane of the routing device, eg. by leading the victim router to use all its computational resources.

It is unlikely that the design of the routing protocol will suffice to prevent against this threat. However, the routing device may also include some functions which would limit the negative consequences of this threat (cf. Section 8).

At the routing protocol control plane, the following options are offered:

R(5.1) Fast rejection schemes based on tokens or cookies MAY be used. Such functionalities MAY be provided by the transport subsystem.

R(5.2) Above requirements regarding neighbors authentication may result in expensive computational checks at the control plane, even though authentication may also be of great help against data plane overload resulting from malicious messages injected on the link. A design SHOULD consider and document opportunities of overloads resulting from protection against usurpation and deception.

R(5.3) The routing protocol operations SHOULD NOT suppose the full-time availability of material (eg: a registry) whose reachability depends on the forwarding service achieved by other routers.

R(5.4) The routing protocol design SHOULD limit the amount of traffic needed for correct operation. This is greatly dependant on the context in which the protocol operates. This also implies rate control of messages sent for session setup (or recovery) when starting-up (or rebooting).

Requirements R(5.[1-2]) suppose that authorized neighbors messages can be authenticated, which helps rejecting attackers solicitations. Further cautions are nonetheless required against neighbors acting in a Byzantine manner.

5.6 Requirements Against Interference

"Interference is a threat action where an attacker uses a subverted link or router to inhibit the exchanges by legitimate routers" [THREATS].

Interference is a general threat which can be perpetrated through:

- Noise addition
- Packets replay
- Denial of forwarding
- Denial of receipts
- Delay of responses
- Break of synchronization
- Slow down of exchanges
- Flapping

Noise addition, where it affects integrity of routing exchanges, is addressed by requirements against misstatements R(3.*). Where it affects the link layer or other traffic, the nature of the threat changes to break of synchronization, overload, etc.

Packets replay is addressed by requirements against misstatements R(3.*).

Denial of forwarding (of routing protocol messages, or of lower level datagrams, packets or frames) cannot be countered. However, it can be detected if the emitter expects an evidence of correct reception (eg: reliable transport), though it is difficult in such a case to make the difference with a denial of receipt (cf. R(6.1.*) below).

Denial of receipts can be detected; even if it may prove to be difficult to figure out the cause of the threat.

R(6.1.a) An implementation SHOULD revise the level of confidence (preference, trust, stability... whatever) associated with destinations whose first hop is a neighbor with which has been detected occurrence of denial of forwarding or denial of receipt.

R(6.1.b) The protocol design MAY affect the way these data are represented, and allow for signaling / sharing stability / trust information.

R(6.1.c) Denial of forwarding or of receipts may result in breaking the states associated with neighbors sessions. Active mechanisms (ex: an in-band echo) MAY be used in order to detect such an anomaly and to set a threshold for an automatic state hygiene maintenance and for confidence revision. Section 8.2 presents resources consumption in details.

Delaying responses beyond a certain threshold is likely to break neighboring relationship, because a routing protocol implementation should time out a neighboring relationship beyond such a threshold (cf. synchronization breaks, R(6.2)). Behind the threshold, delaying may result in the same consequences as a slow down of exchanges (cf. R(6.3)).

There are no way of preventing against breaks of synchronization from subverted links or routers. However:

R(6.2) Protocol design MUST take into account possible breaks of synchronization, even when the threat may only be accidental and improbable. State hygiene and computation of confidence level SHOULD be affected by the detection of such breaks.

Slow down of exchanges may be subjective. It is likely to affect pace to convergence (either in a positive or a negative way), but slow down may also be a 'natural event', when traffic or processing is high, when queues are filling up, etc.

R(6.3) 'Reactivity' of neighbors, possibly with knowledge of the traffic load on the link, MAY be a variable of the heuristic function which computes confidence associated with a neighbor or a particular piece of routing information.

Flapping of routing information is a significant source of instabilities on global routing. It may be difficult to prevent against flapping which results from a subverted routing device. However, a routing device SHOULD lower the disturbance from this event on the network.

R(6.4) Routing information flapping SHOULD be detected through routing databases survey. Propagation of possibly flapping information SHOULD be dampened through appropriate rate control of routing information propagation.

Instabilities of particular pieces of routing information may get

absorbed through information reduction; as an instance, announcing a shorter prefix may hide the flapping on a particular route with a longer prefix. Information reduction, on the other hand, reduces traceability of information (cf. Section 5.1, Section 5.3.3).

With the general threat of interference, the routing decision process is deemed to make choices based on a heuristic evaluation of the confidence associated with a particular neighbor or piece of routing information. Requirements R(6.[1-3]) DO NOT prevent against threats actions, but aim at evaluating the cost of trust as associated with a link or a neighbor. The device may then allocate resources, invalidate routing information, etc., according to the confidence measure. This is not conditions prevention; this is consequences limitation.

Last sections address requirements against disclosure.

5.7 Requirements Against Deliberate Exposure

"Deliberate Exposure occurs when an attacker takes control of a router and intentionally releases routing information directly to devices that, otherwise, should not receive the exposed information" [THREATS].

Deliberate exposure is an information leak about the routing system. Yet, it is unclear to which extend it affects the routing protocol. If neighbors take the exposure into account, then it turns to actually be a spoofing threat, and actual consequence is deception.

There is no way a local instance of the routing protocol may protect against this action if the attacker achieves full control of the device.

This threat may be limited by hardening access to the router, enforcing privilege separations, validating through external devices on the link, etc. This is not directly related to the routing protocol.

5.8 Requirements Against Sniffing

"Sniffing is an action whereby attackers monitor and/or record the routing exchanges between authorized routers. Attackers can use subverted links to sniff for routing information" [THREATS].

As mentioned in the threat document, confidentiality is not generally a design goal of routing protocols. However, confidentiality may be desirable when collecting votes (Byzantine participants may observe others votes and set their alignment so that majority is impossible

or lead to future consequences; on the other hand, clear text communications may also help detecting failures).

R(8.1) A routing protocol design process SHOULD investigate the needs for confidentiality. Conclusions from this process MAY be documented.

R(8.2) A routing protocol CAN optionally provide confidentiality. This SHOULD be implemented on the transport subsystem unless otherwise justified (eg. it is also possible to provide optional and partial confidentiality at the data plane level, or to conceal only a subset of messages).

R(8.3) When confidentiality is in scope, deployment, scalability and performance issues related to it's use SHOULD be studied and the conclusions documented.

5.9 Requirements Against Traffic Analysis

"Traffic analysis is an action whereby attackers gain routing information by analyzing the characteristics of the data traffic on a subverted link" [THREATS].

Even if the confidentiality of the routing traffic is activated, the attacker may access some routing information by analyzing the characteristics of data traffic.

Protections against traffic analysis include traffic flow confidentiality (TFC) (inter-times padding, data padding & compression, generation of dummy packets) and anonymity. Currently, these functionalities are scarcely used on the Internet and often oppose provision of quality of service.

Protecting only the routing protocol against traffic analysis is insufficient because analysis of user traffic will also leak information about the topology and paths properties.

R(9.1) When user traffic is protected against traffic analysis, the routing protocol operations SHOULD investigate the use of a TFC & anonymity enabled transport subsystem shared with user traffic. Design of the routing protocol SHOULD be independent of this operational consideration, unless goal of the protocol is to set up the traffic flow concealing and 'anonymizing' network used by the transport subsystem.

R(9.2) When TFC & anonymity are among the design goals of the routing protocol, their effects on performance and correct operations of the routing system MUST be documented.

6. Living with Byzantine Failures

6.1 The Byzantine Problem

"A node with a Byzantine failure may corrupt messages, forge messages, delay messages, or send conflicting messages to different nodes" [BYZANTINE]. These faults may arise from routers which have been subverted by an attacker or which have faulty hardware or software [THREATS]; as a consequence, many threats are also Byzantine failures. The Byzantine general problem resolution is limited by hypotheses which are reminded here.

Byzantine resistance includes detection of Byzantine failures, Byzantine detection and Byzantine robustness, where the two latter are not necessarily correlated. Next section gives a thorough description of these forms of resistance.

The following main requirements aim at helping in the design of a Byzantine resistant routing protocol:

MR(3.1.a) Local instance of the protocol SHOULD NOT rely on correct operation of any particular neighbor.

MR(3.1.b) Operations associated with a particular neighbor SHOULD always apply a least privilege policy.

MR(3.1.c) Only traffic source and destination SHOULD be considered trustworthy.

MR(3.2) Messages MUST be authenticated when sent and checked for their authenticity when received (cf. also R(3.[1-3]). Use of cryptography simplifies the Byzantine problem.

6.2 Byzantine General Requirements

Classical hypotheses for Byzantine failure resolution are:

- devices are fully connected,
- the decision that must be agreed upon is binary (yes/no),
- the network is synchronous,
- strictly less than a third of the devices are faulty.

Under these hypotheses, a distributed algorithm requires as many rounds as the number of faults to be tolerated plus one.

Further information about distributed agreement can be found in [CONSENSUS]. In the following, we will only focus on what makes the problem tractable in IP networks.

The ability to send messages to all neighbors simultaneously allow for simulation of both full connectivity and synchronization. The fact that routing information is not a agreeable binary decision has little consequences because agreement is not an absolute requirement; see Section 6.5 and [BYZANTINE].

6.3 Detection of the Occurrence of a Byzantine Failure

The protocol algorithm may detect incoherences within the correlated routing information upon algorithm termination, abnormal attractive cycles within routes computations, etc. These events may be symptoms of a Byzantine failure occurring. More trivial evidences of a possible Byzantine failure is when agreement, termination or validity of the consensus cannot be achieved.

R(10.1) It SHOULD be possible to derive from a routing protocol design a set of coherence and sanity checks. The routing protocol documentation SHOULD mention directions when incoherence occurs, and describes reactions which are of direct impact on the protocol operation.

6.4 Byzantine Detection

Byzantine detection is much more accurate than just detecting a Byzantine failure and consists in the ability to find out which participants are subverted. A part of inherent risk of Byzantine detection is that when the number of traitors grow past a limit, it may be difficult for a device to figure out which group is subverted. Sometimes, the considered device may be itself -or conclude it is itself- faulty.

R(11.1) When Byzantine detection is achieved, automatic responses MAY be triggered in order to prevent Byzantine nodes from damaging operation of the routing protocol.

R(11.1.a) Automatic responses following a Byzantine detection MUST NOT prevent subverted devices from participating again when they cease to behave incorrectly.

R(11.1.b) Automatic responses following a Byzantine detection MUST NOT deceive non-faulty neighbors in concluding that responding devices are Byzantine nodes.

Possible automatic responses that may be investigated are the simulation of a link shutdown, setup of adequate local policies, quarantine cell. Collaborative approach between detectors to limit the influence of some subverted devices may be quite hazardous.

Either at the database maintenance level or at the routing decision process level, the following SHOULD be configurable when dealing with a detected subverted device:

R(11.2.a) "Detour": Allow or deny forwarding along an alternate route (if available), possibly on a path for which a "lower quality" (much many hops, longer delay, etc) is probable. The routing protocol instance MAY also seek actively after an alternate route.

R(11.2.b) "Send & Hope": Allow forwarding to the subverted device anyway or,

R(11.2.c) "Discard": Treat destination as unreachable.

Eventually, note that sharing symmetric material for partial authentication between more than two devices would make Byzantine detection impossible to achieve in most cases (and so would do the absence of any authentication mechanism).

6.5 Byzantine Robustness

Purpose of Byzantine robustness, in the general problem context, is for any given device to achieve algorithm termination, agreement and -naturally- validity. This does not imply Byzantine detection.

However, in the routing context, what matters really is routing information correctness (cf. Section 3):

R(12.1) Routing protocols do NOT REQUIRE to achieve agreement.

R(12.2) Routing protocols do NOT REQUIRE to terminate; in fact, it is generally expected that they will not terminate during normal operation.

Some routing protocols operates in context for which reachability is more important than attributes associated with the destination. In such scenarii, Byzantine robustness aims at protecting reachability. This manages opportunities for "severed configurations" in which some local policy requirements for a traffic could not be enforced though reachability is still possible / probable (Remember that what is often expected on the Internet is a high probability of packet delivery).

7. Security Techniques for Routing

7.1 Techniques when Originating

When originating, security requirements have a tendency to focus on the data plane. Indeed, data will further get propagated through the network, out of originator's control. Security mechanisms addressing the control side will have no control on the way data are eventually propagated. Moreover, believing that other devices will propagate the information unmodified is naive. As an instance, aggregation or filtering may be threats against resources' properties.

As a consequence, it is important to know whether the originated information is authentic or not. Even though trusting unauthenticated information may appear to be a necessity in some scenarios, it is useful to set such a distinction on information so as to derive a confidence level associated with it. In order to allow for origin authentication, information may be considered as a kind of 'record', composed of sections of the kind:

- Network resources description
- Related properties set by resources' controller.
- Integrity and data-origin authenticity evidence of information, provided by the controller of the resources. This evidence should be lifetime limited.
- Properties set by propagators of routing information (possibly with a time-limited authenticity evidence).

If propagators discard authenticity evidence, then the information should acquire a lower preference level.

The division presented in Section 5.1 between the controller and the advertiser allows for granting the advertising device with a very limited control on what is advertised; this is an interesting protection against potential damages resulting from possible advertiser's subversion.

The concept of an authorization chain linking ownership, control and advertisement is nonetheless necessary in order to build confidence between neighbors from different organizations. An issue with this kind of model is the need for a definition (or furthermore: a specification and an allocation scheme) of identities.

Obviously, on a large scale, this kind of data protection requires public key operations, regardless of the actual technology eventually

used (authorization tokens, digital signature). There are quite a lot of drawbacks associated with cryptography in general and with public key cryptography in particular.

Where these drawbacks affect devices, an increase amount of memory is needed for buffering cryptographic information and for caching (cf. next paragraph). Besides, public key operations are also quite CPU consuming. A performance study SHOULD be pursued when designing a routing protocol using cryptography; threats opened because of crypto processing SHOULD NOT nullify the interest of tackling routing threats which would result in comparable consequences (eg. disruption). A performance study often requires hypotheses on the underlying hardware, which is somewhat restricting but necessary.

Where these drawbacks concern the overall architecture, they involve deployment, administration and public information reachability issues. Regarding this latter topic, in-band or stand-alone channels are necessary for the provision of public data, for revocation and for key roll-over. A routing protocol may find itself in a dead-end if such a channel is needed for authenticity check of data which are necessary to enable access to the ad-hoc channel. This is a tricky point, which may claim for a distributed caching mechanism. Caching is all the more important when scalability is a significant issue and when centralization of data creates bottleneck; on the other hand, the whole architecture is less reactive in case revocation or key roll-overs are required, even though soft key transitions should not be necessary in this context.

Further in this direction, neighbors' public material may be kept in non-volatile storage for recovery. There may be no routes available in order to retrieve this material after a reboot, though in-band provisioning within the routing protocol is also a possibility.

Whatever the path taken by an architecture specification, it's resistance against trivial denial of services must be evaluated.

Requirements related to this section are R(1.*), R(2.*), R(3.[4-6]).

All cryptographic material MUST have their lifetime limited, and both evaluated in terms of time and in terms of amount of data.

Public keys strength is a matter of context: in inter-domain operations, one may expect that public material will not change very often, and then such a material should be significantly strong. Locally, the rate of public material updates may depend on administrator's decision; he alone evaluates the risks for the network and the administrative cost. In a conference, people may build a ephemeral network by exchanging public material on an direct

IR link before roaming and participating in ad-hoc routing through wireless links; public material is such a case would only be used a few hours and may be kept voluntarily weak.

7.2 Techniques when Propagating

When propagating, security requirements have a tendency to focus on the control plane. Propagation security is that of entities communicating in a direct fashion (and perhaps interactively) over the transport subsystem. In such a situation, we're concerned with:

- Integrity: data integrity between neighbors is an obvious requirement. Note that error detection and correction codes are not integrity evidences. Means to achieve integrity are signed-hash and keyed-hash. Data integrity is always closely related to authenticity.

- Authenticity: the above feature is of no use without authentication of the information producer. Authenticating correctly the messages sent from neighbors is one of the most important security requirement. Authentication techniques that can be considered are: digital signature, keyed hash.

- Anti-replay: comes here mainly for protection against active attacks from subverted Links, though this feature will also provide protection against 'natural' packets duplication. Note that underlying layers may provide an unauthenticated anti-replay feature, which would be of no use from a security point of view unless it gets also authenticated. Authentication of routing exchanges sequence numbers may bring this kind of protection to the protocol.

Other features include confidentiality and traffic flow confidentiality, which are generally out of scope in routing protocols (cf. R(8.*) and R(9.*)).

Main differences with origin-based security practices presented in the previous section include:

- message oriented protection (as opposed to data protection),

- messages are addressed (to one or many peers),

- messages are limited in time through anti-replay techniques (as opposed to limited lifetime),

- neighbors may use symmetric cryptography.

The above characteristics may be implemented by the routing protocol, or by the transport subsystem. In this latter case, a specification MUST document which security properties are provided by the transport subsystem, which are provided by the routing protocol and, eventually, how they interact.

Note that transport subsystems may experience evolutions; as a trivial instance, one may design a routing protocol which will run on wire Ethernet (802.3) with the hypothesis that physical and logical access to layer 2 infrastructure is under control. Such an hypothesis may no longer be suitable on wireless Ethernet (802.11).

Further protection may include range limiting features, enabled by the use of special addresses (link-local, limited broadcast, multicast) or of counter-based schemes (TTL). Most of these features are provided by adequate transport subsystems.

Specific issues for communications between neighbors include:

- Address protection: sometimes extra care is needed against transport subsystem's address spoofing, even though an identity has been defined at an upper layer. Address protection requires inclusion of the address in the integrity and authenticity evidence computation. [AH] may be seen as an instance of a protocol with built-in address spoofing protection.

- In 'one to many' communication contexts, sharing symmetric material opens opportunities for damages resulting from subverted insiders.

- Interactivity involves managing sessions and keeping states associated with neighbors. For the sake of state hygiene, reactivity of neighbors SHOULD be evaluated. This calls for setting delays threshold, using keep alive / heart beat mechanisms and explicitly tearing sessions down.

- Participants are vulnerable to direct computational harassment, against which DOS mitigation mechanisms are necessary. These include puzzles, cookies, tokens chains.

Requirements related to this section are R(1.*), R(3.[1-3]) and R(4.*). Section 5.3.3 is also related to this section.

When possible, methods to derive a symmetric key from public exponents should be used, given that the symmetric cryptography operations considered are less computationally expensive. Caution should be taken if the number of devices sharing the same symmetric key is greater than two.

Limiting keys lifetime and refreshing them is merely cryptographic hygiene. Therefore, a refresh mechanism is REQUIRED both for public keys and for session keys; Public keys may not require a soft transition, while refreshing session keys may require to move from the old key to the new one with no session interruption. For session keys, lifetime SHOULD be evaluated both in terms of time and of amount of data.

Actual mechanisms used to limit key lifetime MAY either be based on an explicit lifetime associated with key (ex: public key bundled with a validity date) or on roll-over. Both MAY be used simultaneously for different purposes within a single system.

7.3 Security of the Functional Parts

The threats document [THREATS] introduces a set of functions commonly shared by routing protocols: the transport subsystem, the neighbor state maintenance function and the database maintenance function.

Each of these functions may contain inner security weaknesses and simultaneously a potential for providing adequate security services for the interest of operation of the whole system.

In the following sections, the security related parts of these functions are explored.

7.3.1 Transport Subsystem

"The routing protocol transmits messages to its neighbors using some underlying protocol. For example, OSPF uses IP, while other protocols may run over TCP" [THREATS].

One may design a routing protocol independent -to a certain extent- from a specific transport subsystem, by requiring the availability of a minimal set of capabilities from this subsystem.

Yet, relevant, specific capabilities of a transport subsystem SHOULD be exploited by a routing protocol. An adequate transport subsystem provides capabilities which would be cumbersome if included in the routing protocol itself and have been -ideally- thoroughly tested. This is a net gain in complexity, even though at the expense of added complexity on protocol interactions and addresses resolution mechanisms.

FR(T.1) A routing protocol specification SHOULD document which capabilities of the transport subsystem are exploited by the routing protocol.

FR(T.2) Where issues may arise from interactions between the transport subsystem and the routing protocol, the specification MUST mention these issues (The "Security Considerations" section may be the appropriate place for IETF/IRTF documents).

The transport subsystem may already provide the following properties:

- Neighbors discovery and maintenance: A given Transport Subsystem technology may provide a way to discover and communicate with adjacent devices participating in the routing domain (neighbors). This is a critical property.

- Range limitation: the subsystem may provide a way to limit propagation of messages outside a certain range and in the same way limit intrusions from outsiders in the neighborhood. This may be achieved either through the use of an appropriate layer (likely, link layer), through special addresses (limited broadcast, multicast, link-local, site-link, etc.), through conditions expressed on TTL (see also [BTSH]). This provides a limited access control to neighborhood (yet, there are ways around these limitations: VLAN frames hopping, tunneling).

- Separate control channel: if the underlying technology provides separated channels for control traffic and user data traffic, this may help against DOS against the routing protocol. Such control channels may be provided via the same Link Layer infrastructure, or perhaps via a distinct network.

- Integrity: While the Transport Subsystem chosen by the routing protocol designer may provide error detection code, this does not provide data integrity from a security point of view. The Transport Subsystem may also provide data integrity which will still be useless from a security perspective if the secret material used by the data integrity service cannot be tied to the routing protocol participant identity.

- Authenticity: if the underlying layer both provides authenticity and integrity, many routing threats may be thwarted. Further investigations are required though, among which are studies of resistance to replay, performance, Byzantine detection and robustness, etc. In such a case, the documentation of the routing protocol MUST state which security properties are provided by the Transport Layer, which are provided by the routing protocol design and eventually how they interact (cf. FR(T.2)).

- Address spoofing protection: the subsystem is protected against address spoofing if integrity and authenticity evidence covers also the address.

7.3.2 Neighbor State Maintenance

"Neighboring relationship formation is the first step for topology determination. For this reason, routing protocols may need to maintain state information. Each routing protocol may use a different mechanism for determining its neighbors in the routing topology. Some protocols have distinct exchanges through which they establish neighboring relationships, e.g., Hello exchanges in OSPF" [THREATS].

Specifications MAY document the use of cookies or damping mechanisms in order to protect this function from trivial denial of services.

7.3.3 Database Maintenance

"Routing protocols exchange network topology and reachability information. The routers collect this information in routing databases with varying detail. The maintenance of these databases is a significant portion of the function of a routing protocol" [THREATS].

From a local perspective, and with a selfish point of view, database maintenance is what really matters for a particular device.

For this reason, resources SHOULD be 'flagged' according to trust, stability, quality... scales.

Coherence of information MAY be checked actively (with probes) and passively (observation of user traffic). In ad-hoc contexts, database may also be fed reactively. Such mechanisms MAY affect gently resources flags, according to the reliability of information acquired in this way. In some cases, it may prove advisable to consider these hints as bonus for the information preference (as an instance, when destinations are overclaimed, detecting dead networks behind the large prefix should not result in depreciating the overclaimed information [this is open to discussion, of course]).

7.3.3.1 Fail-back Procedures

When detecting obvious routing misbehavior which result from misuse of the routing protocol, but when sources responsible for this misbehavior cannot be identified, fail-back procedures MAY be attempted, based on previous recorded states, fail-safe states or heuristics on the routing information and on trust. Degradation of the service should often be better than no service at all, thus the device may adjust local route costs information when such events occur. The routing protocol design may document guidelines and requirements on such procedures.

Network management MUST be able to install unalterable (static) routes to allow debugging network problems without interference from routing protocols. Such routes may be pre-configured and loaded upon detection of abnormal behaviors (flapping...).

7.4 Date and Time Issues

8. Local Security

8.1 Active Participation to Security

Topics presented within this section may not be directly tied to the protocol design. However, it addresses several local considerations that are requirements for a secure operation of the routing protocol and of the device it is running on.

8.1.1 Checking

A routing device may be configured to run extra checks on the routing state, like checking databases against previous information. Some active tests may also be triggered, possibly involving device's neighbors. High caution should be taken regarding implementation of such features and they should not jeopardize the routing protocol mechanisms.

8.1.2 Reporting

A set of error messages may be designed in order to report detection of failures to other participants. Locally, a set of auditable events MUST be defined.

8.1.2.1 Auditable Events

The following events should be audited:

1. Authentication failure

2. Required public information (keys, authority) is not available

3. Errors reported by forwarders

4. Detection of a Byzantine event

5. Detection of a rebooting peer

8.1.3 Reacting

8.1.3.1 Filtering

Upon detection of subverted devices, a process may enforce security procedures such as ingress filtering or participant exclusion.

A routing device MAY be set to drop/reject routing messages if these are incorrect with current configuration of the network, e.g. if

they do not belong to the correct range of the IGP, etc.

Note that this protection is topological and partial. Extreme care should be taken not to jeopardize correct behavior of the protocol.

8.1.3.2 Correcting

8.2 Local Resources Considerations

Even though this document addresses routing protocols, these cannot operate without a platform of hardware and software to support them. All the resources belonging to this platform form what is generally referred to as a router. Thus, routers comprise all local resources of a routing daemon participating in a routing session.

This section will first highlight critical underlying components and their security issues regarding Denial of Service (DoS) vulnerabilities and then suggest suitable routing protocols' requirements addressing these issues.

8.2.1 Denial of Service Attacks

The Computer Emergency Response Team (CERT) defines in [DOS] Denial of Service attacks as being explicit attempts by attackers to prevent legitimate users of a service from using that service. Denial of Service attacks can be launched against a target for the mere purpose of preventing the victim from using a resource or can be a component of a greater attack that may ultimately aim at stealing information.

A modern router is a complex system made of several hardware and software components that interact in the effort to serve the general purpose of routing as defined in Section 3. All of these components are finite resources and therefore intrinsically prone to Denial of Service. The impact of Denial of Service attacks on certain local resources can be critical for the routing protocols running on them.

8.2.2 Hardware Resources

Almost every hardware component in a router is essential to the correct functioning of the local instances of the various routing protocols that run on it, for example - trivially speaking - without power no packets will be routed. Among others buffers/queues and CPU cycles are two of the less obvious resources that are critical for routing protocols.

8.2.2.1 Buffers/Queues

Buffers are widely used in hardware to store information that needs

to be aggregated or delayed before being consumed. In general once a buffer is full every subsequent object that needs to be stored in that queue will simply be discarded. Depending on what messages are discarded, the consequences of dropping information for routing protocols can vary from negligible to critical.

Since all messages exchanged between participants to a routing session need to reach the control-plane, the queues and buffers that support this link are critical for routing protocols. Often people are deceived by thinking that the throughput of a switching fabric is roughly the amount of bandwidth needed to launch a DoS attack against a given router; in reality, routers have smaller bandwidth links toward the control plane. The goal of an attacker could be easier in terms of resources, if he/she were to attempt to exhaust the buffers and queues on the link to the control plane with bogus control plane packets rather than trying to congest the resources serving the switching fabric. The goal of such attacks would be to cause queues and buffers to drop legitimate routing messages together with bogus ones.

8.2.2.2 CPU Cycles

Processors units, and in particular Network Processors (NPs), are a valuable resource that can perform predetermined sets of operations during a single cycle. Generally speaking, CPU cycles are a finite resource that is shared among many different processes, some of these being instances of routing protocols. As a consequence of congestion, and from an oversimplified point of view, some processes may be put "on hold" until more CPU cycles are available, or every process may be "starved a bit". Both scenarios may cause great damage to interactive processes. In particular routing protocols' instances may enter critical states where a timely reaction to an event is necessary but not available.

In general the more a CPU serves an heterogeneous pool of processes, the easier it will be for an attacker (or a faulty router) to find a single service/process that will exhaust a significant portion of the available CPU cycles, denying service to other processes, such as routing.

8.2.2.3 Buffer/Queues and CPU Cycles Requirements

Routing messages SHOULD be identifiable as coming from legitimate participants in their routing session before being directed towards the control-plane.

If any rate limiting mechanism is intended by the routing protocol to mitigate congestion of control-plane links, said solution MUST be

designed ensuring that an attacker cannot directly exploit it in the attempt to block a legitimate routing peer from exchanging routing messages.

8.2.2.4 Bandwidth

Routing protocols are based on the exchange of information between the participants to a session over network links. A link's bandwidth is finite critical resource that, if starved, can lead to Denial of Service attacks on the routing protocols. If a link is not malfunctioning, and neglecting transmission errors, then DoS attacks on a link's bandwidth can only take place at the link's ends. A router may receive an aggregate of traffic higher than it can be forwarded by a given output interface, or a receiving router may not be capable of handling the current load of traffic incoming on a given interface due to an internal scheduling priority problem or because it entered a critical or unknown state.

8.2.2.4.1 General Mitigation Techniques

Some mitigation techniques can be deployed to limit the exhaustion of bandwidth between two routing peers; two current examples are: ingress filtering, as described in [FILTERING], and solutions that rely on Quality of Service mandating that the highest priority and availability be assigned to routing messages.

8.2.2.5 Bandwidth Requirements

Routing protocols MUST be designed to easily inter-work with lower layers Quality of Service mechanisms.

8.2.3 Logic (Software) Resources

Similarly to hardware resources, logic resources can be finite and therefore exhausted thus affording attackers with the possibility of launching Denial of Service attacks. Databases are critical resources for every routing protocol and they may contain information about link-state, direct neighbors, active peers, external routes database, etc...

Routing databases have a maximum number of entries that can be stored in them and this is generally not defined by the routing protocols. This upper bound can be set by an administrator through a configuration parameter or can be restricted only by the hardware memory available to the routing platform. Either way, when this limit is approaching, for any of the databases maintained by a routing protocol, some action must be taken.

8.2.3.1 Logic (Software) Requirements

Routing protocols MUST mandate verification of every piece of information that can be verified before committing it to any underlying database.

Every piece of information that cannot be verified by the routing protocol immediately MUST be marked as temporary and means should be provided, by the routing protocol itself, to keep track of these entries, verify and discard them as soon as possible.

Every piece of information that cannot be verified by the routing protocol MUST be installed in the apposite database with the minimum time to live compatible with its function.

Routing protocols MUST provide mechanisms for routing platforms' databases, in overflow state, to discard information that will cause minimum possible disruption to the routing session.

Routing protocols SHOULD be designed as to incorporate feed-back solutions from databases approaching overflow state so that mitigative actions can be taken.

Routing protocols SHOULD be designed with the concept of graceful degradation in mind in order to better survive in case any of the underlying databases approaches or enters overflow state.

9. Inter-Domain Routing Issues

9.1 Legitimacy

An important issue in inter-domain routing is legitimacy for claiming network resources. In fact, this is where confidence edifice starts. Requirements R(1.*) are related to this topic, though they do not address some decisions.

Parts of these decisions regard routes specialization.

Hierarchical addressing is necessary in order to aggregate entries in local routing tables; this reduces tables size and improve general performances, even though this may threaten performances on a specific path. When preferred (eg. for confidentiality reasons), some specific routes may appear in the table. A problem with hierarchical addressing is that, when used as such in the routing protocol, it may generate resources masking. This is especially obvious with operations like aggregations of destinations or removal of a specific destination: both these operations will result in the generic entry taking over the specific one.

These operations may be considered as a violation of ownership, though it is also unclear whether a shorter prefix ownership should -administratively speaking- involve authority on a corresponding longer prefix.

On the other hand, if care is taken within the routing protocol to protect specific routes against overclaiming resulting from aggregations or removal, then this involves extra architecture requirements and more bandwidth get consumed in routing protocol exchanges.

Besides, this will not prevent routing tables from aggregating or removing entries, and this kind of decorrelation between routing information and the way packets get actually forwarded may not be desirable, even though loose relation between local routing tables and routing information is common.

Another part of the problem is public information reachability.

When public material may help in establishing right to claim resources, availability of the required material is problematic. Section 7.1 presents this in further details. With regard to public cryptography, it should be clear that a light paradigm (authorizations ?) would better fit in most cases, though third parties also appear to be a necessity at this point.

9.2 Propagating policies

Policy propagation within a routing protocol which operates between administrative routing domains, exterior gateway protocols, is very difficult. This particular area of security is fraught with difficulties making it next to impossible to actually secure policy across multiple administrative domains.

Since each administrative domain can add policies to a given route, anyone can essentially insert any policy. Even if a full history of policies is available, the question: "Who's policy are we honoring ?" has to be answered. The originator's policy ? Or the AS we received the route from ? Or the AS that currently has the route ? Or some other AS ?

9.3 Coherence

Where domains are multi-homed, should operations of the edge routers be coherent ? In a nutshell: should a domain be considered as a stand-alone, non-schizophrenic, entity ? Note that coherence does not preclude edge routers from behaving differently.

9.4 Confidentiality

As was mentioned several times previously, confidentiality is usually not a design goal of routing protocols. In inter-domain operations, enabling confidentiality would require finding a balance between information that is required to be publicly available and information whose concealing is desirable. May be a possible path is not to care about concealing destination info, but about properties applying to resources. Yet, the value of a route without knowledge of according properties is certainly dubious.

9.5 Agreements involving operators

Secure EGPs operations will require kind of agreements between the involved parties. Though operators may achieve these agreements on a case by case basis, this is unlikely to be effective in the field. Emergence of trusted third parties upon which would rely the diffusion of public key material and relations to prefix ownership would fit better.

Another question is whether these pieces of information must be tied with public information related to the system ownership, such as the organization name. This may lead to specific routing policies or abuses that would introduce more complexity.

Access control also imply agreements: who's granted right to

participate to the protocol ?

10. Security Considerations

This entire draft RFC is security related. Specifically it addresses security of routing protocols and routing systems as associated with requirements to those protocols and systems. In a larger context, this work builds upon the recognition of the IETF community that signaling and control/management planes of networked devices need strengthening. Routing protocols and routing systems can be considered part of that signaling and control plane, may be the most important. However, to date, these protocols and systems have largely remained unprotected and opened to malicious attacks. This document discusses routing protocol and routing system security requirements as we know them today and lays the foundation for the design of new, more secure, routing protocols and systems.

11. References

11.1 Normative References

[AH] Kent, S. and R. Atkinson, "IP Authentication Header", RFC 2402, November 1998, <www.ietf.org/rfc/rfc2402.txt>.

[DAMPING] Villamizar, C., Chandra, R. and R. Govindan, "BGP Route Flap Damping", RFC 2439, November 1998, <www.ietf.org/rfc/rfc2439.txt>.

[FILTERING] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, May 2000, <www.ietf.org/rfc/rfc2827.txt>.

[KEYWORDS] Bradner, S., "Key Words for Use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <www.ietf.org/rfc/rfc2119.txt>.

[SEC-GLOSS] Shirey, R., "Internet Security Glossary", RFC 2828, May 2000, <www.ietf.org/rfc/rfc2828.txt>.

11.2 Informative References

[BTSH] Vijay, G., Heasley, J. and D. Meyer, "The BGP TTL Security Hack (BTSH)", Internet Draft; version 02, May 2003, <www.ietf.org/internet-drafts/draft-gill-btsh-02.txt>.

[BYZANTINE] Perlman, R., "Network Layer Protocols with Byzantine Robustness", , August 1988, <www.vendian.org/mncharity/dir3/perlman_thesis/>.

[CONSENSUS] Coulouris, G., Kindberg, T. and J. Dollimore, "Distributed Systems: Concepts and Design", Addison Wesley ISBN - 0201619180, 2000 September.

[DOS] CERT, "Denial of Service Attacks", June 2001, <www.cert.org/tech_tips/denial_of_service.html>.

[SMITH] Smith, R. and al., "Securing Distance-Vector Routing Protocols", Symposium on Network and Distributed System Security , February 1997, <www.isoc.org/isoc/conferences/

ndss/97/smith_sl.pdf>.

[THREATS] Barbir, A., Murphy, S. and Y. Yang, "Generic Threats to Routing Protocols", Internet Draft; version 06, April 2004, <www.ietf.org/internet-drafts/draft-ietf-rpsec-routing-threats-06.txt>.

Authors' Addresses

Jean-Jacques Puig
CNRS / UMR 5157 (Samovar) / Piece A-108
9, Rue Charles Fourier
Evry 91011
France

Phone: +33 1 60 76 44 65
Fax: +33 1 60 76 47 11
EMail: jean-jacques.puig@int-evry.fr
URI: http://www-lor.int-evry.fr/~puig/

Mohammed Achemlal
France Telecom R & D

EMail: mohammed.achemlal@francetelecom.com

Emanuele Jones
Alcatel Canada - R&I - Security group
600 March Road
Kanata, ON K2K 2E6
Canada

Phone: +1 613 784 5977
Fax: +1 613 784 8944
EMail: emanuele.jones@alcatel.com

Danny McPherson
Arbor Networks

EMail: danny@arbor.net

Appendix A. Acknowledgments

The authors would like to acknowledge the suggestions and contributions of:

- Russ White - CISCO

Appendix B. Revision History

B.1 Changes from draft-puig-rpsec-generic-requirements-02

Further development on requirements. Incorporation of off-list comments. Deletion of solution space paragraphs.

B.2 Changes from draft-puig-rpsec-generic-requirements-01

TOC tweaking. Phrasing simplifications. Development of the requirements.

B.3 Changes from draft-puig-rpsec-generic-requirements-00

Full TOC change.

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.