DOTS   A. Mortensen
Internet-Draft   Arbor Networks
Intended status: Informational   R. Moskowitz
Expires: August 12, 2018   Huawei
                                                                T. Reddy
                                                                  McAfee
                                                       February 08, 2018

Distributed Denial of Service (DDoS) Open Threat Signaling Requirements
                    draft-ietf-dots-requirements-14

Abstract

   This document defines the requirements for the Distributed Denial of
   Service (DDoS) Open Threat Signaling (DOTS) protocols enabling
   coordinated response to DDoS attacks.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

1.1. Context and Motivation

   Distributed Denial of Service (DDoS) attacks afflict networks of all
   kinds, plaguing network operators at service providers and
   enterprises around the world. High-volume attacks saturating inbound
   links are now common, as attack scale and frequency continue to
   increase.

   The prevalence and impact of these DDoS attacks has led to an
   increased focus on coordinated attack response. However, many
   enterprises lack the resources or expertise to operate on-premises
   attack mitigation solutions themselves, or are constrained by local
   bandwidth limitations. To address such gaps, service providers have
   begun to offer on-demand traffic scrubbing services, which are
   designed to separate the DDoS attack traffic from legitimate traffic
   and forward only the latter.

   Today, these services offer proprietary interfaces for subscribers to
   request attack mitigation. Such proprietary interfaces tie a
   subscriber to a service while also limiting the network elements

capable of participating in the attack mitigation. As a result of
signaling interface incompatibility, attack responses may be
fragmented or otherwise incomplete, leaving operators in the attack
path unable to assist in the defense.

A standardized method to coordinate a real-time response among
involved operators will increase the speed and effectiveness of DDoS
attack mitigation, and reduce the impact of these attacks. This
document describes the required characteristics of protocols that
enable attack coordination and mitigation of DDoS attacks.

DDoS Open Threat Signaling (DOTS) communicates the need for defensive
action in anticipation of or in response to an attack, but does not
dictate the implementation of these actions. The requirements in
this document are derived from [I-D.ietf-dots-use-cases] and
[I-D.ietf-dots-architecture].

1.2. Terminology

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 [RFC2119].

This document adopts the following terms:

DDoS: A distributed denial-of-service attack, in which traffic
   originating from multiple sources is directed at a target on a
   network. DDoS attacks are intended to cause a negative impact on
   the availability and/or other functionality of an attack target.
   Denial-of-service considerations are discussed in detail in
   [RFC4732].

DDoS attack target: A network connected entity with a finite set of
   resources, such as network bandwidth, memory or CPU, that is the
   target of a DDoS attack. Potential targets include (but are not
   limited to) network elements, network links, servers, and
   services.

DDoS attack telemetry: Collected measurements and behavioral
   characteristics defining the nature of a DDoS attack.

Countermeasure: An action or set of actions focused on recognizing
   and filtering out specific types of DDoS attack traffic while
   passing legitimate traffic to the attack target. Distinct
   countermeasures can be layered to defend against attacks combining
   multiple DDoS attack types.

Mitigation: A set of countermeasures enforced against traffic
   destined for the target or targets of a detected or reported DDoS
   attack, where countermeasure enforcement is managed by an entity
   in the network path between attack sources and the attack target.
   Mitigation methodology is out of scope for this document.

Mitigator: An entity, typically a network element, capable of
   performing mitigation of a detected or reported DDoS attack. The
   means by which this entity performs these mitigations and how they
   are requested of it are out of scope. The mitigator and DOTS
   server receiving a mitigation request are assumed to belong to the
   same administrative entity.

DOTS client: A DOTS-aware software module responsible for requesting
   attack response coordination with other DOTS-aware elements.

DOTS server: A DOTS-aware software module handling and responding to
   messages from DOTS clients. The DOTS server enables mitigation on
   behalf of the DOTS client, if requested, by communicating the DOTS
   client's request to the mitigator and returning selected mitigator
   feedback to the requesting DOTS client.

DOTS agent: Any DOTS-aware software module capable of participating
   in a DOTS signal or data channel. It can be a DOTS client, DOTS
   server, or, as a logical agent, a DOTS gateway.

DOTS gateway: A DOTS-aware software module resulting from the
   logical concatenation of the functionality of a DOTS server and a
   DOTS client into a single DOTS agent. This functionality is
   analogous to a Session Initiation Protocol (SIP) [RFC3261] Back-
   to-Back User Agent (B2BUA) [RFC7092]. A DOTS gateway has a
   client-facing side, which behaves as a DOTS server for downstream
   clients, and a server-facing side, which performs the role of DOTS
   client for upstream DOTS servers. Client-domain DOTS gateways are
   DOTS gateways that are in the DOTS client's domain, while server-
   domain DOTS gateways denote DOTS gateways that are in the DOTS
   server's domain. DOTS gateways are described further in
   [I-D.ietf-dots-architecture].

Signal channel: A bidirectional, mutually authenticated
   communication channel between DOTS agents that is resilient even
   in conditions leading to severe packet loss, such as a volumetric
   DDoS attack causing network congestion.

DOTS signal: A concise authenticated status/control message
   transmitted over the signal channel between DOTS agents, used to
   indicate the client's need for mitigation, as well as to convey
   the status of any requested mitigation.

Heartbeat: A message transmitted between DOTS agents over the signal
    channel, used as a keep-alive and to measure peer health.

Data channel: A bidirectional, mutually authentication
    communincation channel between two DOTS agents used for infrequent
    but reliable bulk exchange of data not easily or appropriately
    communicated through the signal channel under attack conditions.

Filter: A specification of a matching network traffic flow or set of
    flows. The filter will typically have a policy associated with
    it, e.g., rate-limiting or discarding matching traffic [RFC4949].

Blacklist: A list of filters indicating sources from which traffic
    should be blocked, regardless of traffic content.

Whitelist: A list of filters indicating sources from which traffic
    should always be allowed, regardless of contradictory data gleaned
    in a detected attack.

Multi-homed DOTS client: A DOTS client exchanging messages with
    multiple DOTS servers, each in a separate administrative domain.

2. Requirements

This section describes the required features and characteristics of
the DOTS protocols.

The DOTS protocols enable and manage mitigation on behalf of a
network domain or resource which is or may become the focus of a DDoS
attack. An active DDoS attack against the entity controlling the
DOTS client need not be present before establishing a communication
channel between DOTS agents. Indeed, establishing a relationship
with peer DOTS agents during normal network conditions provides the
foundation for more rapid attack response against future attacks, as
all interactions setting up DOTS, including any business or service
level agreements, are already complete. Reachability information of
peer DOTS agents is provisioned to a DOTS client using a variety of
manual or dynamic methods. Once a relationship between DOTS agents
is established, regular communication between DOTS clients and
servers enables a common understanding of the DOTS agents' health and
activity.

The DOTS protocol must at a minimum make it possible for a DOTS
client to request aid mounting a defense, coordinated by a DOTS
server, against a suspected attack, signaling within or between
domains as requested by local operators. DOTS clients should
similarly be able to withdraw aid requests. DOTS requires no
justification from DOTS clients for requests for help, nor do DOTS

clients need to justify withdrawing help requests: the decision is
local to the DOTS clients' domain. Multi-homed DOTS clients must be
able to select the appropriate DOTS server(s) to which a mitigation
request is to be sent. The method for selecting the appropriate DOTS
server in a multi-homed environment is out of scope.

DOTS protocol implementations face competing operational goals when
maintaining this bidirectional communication stream. On the one
hand, DOTS must include protections ensuring message confidentiality,
integrity and authenticity to keep the protocols from becoming
additional vectors for the very attacks it is meant to help fight
off. On the other hand, the protocol must be resilient under
extremely hostile network conditions, providing continued contact
between DOTS agents even as attack traffic saturates the link. Such
resiliency may be developed several ways, but characteristics such as
small message size, asynchronous, redundant message delivery and
minimal connection overhead (when possible given local network
policy) will tend to contribute to the robustness demanded by a
viable DOTS protocol. Operators of peer DOTS-enabled domains may
enable quality- or class-of-service traffic tagging to increase the
probability of successful DOTS signal delivery, but DOTS does not
require such policies be in place, and should be viable in their
absence.

The DOTS server and client must also have some standardized method of
defining the scope of any mitigation, as well as managing other
mitigation-related configuration.

Finally, DOTS should be sufficiently extensible to meet future needs
in coordinated attack defense, although this consideration is
necessarily superseded by the other operational requirements.

2.1. General Requirements

GEN-001 Extensibility: Protocols and data models developed as part
   of DOTS MUST be extensible in order to keep DOTS adaptable to
   operational and proprietary DDoS defenses. Future extensions MUST
   be backward compatible. DOTS protocols MUST use a version number
   system to distinguish protocol revisions. Implementations of
   older protocol versions SHOULD ignore information added to DOTS
   messages as part of newer protocol versions.

GEN-002 Resilience and Robustness: The signaling protocol MUST be
   designed to maximize the probability of signal delivery even under
   the severely constrained network conditions caused by particular
   attack traffic. The protocol MUST be resilient, that is, continue
   operating despite message loss and out-of-order or redundant
   message delivery. In support of signaling protocol robustness,

DOTS signals SHOULD be conveyed over a transport not susceptible
to Head of Line Blocking.

GEN-003 Bulk Data Exchange: Infrequent bulk data exchange between
DOTS agents can also significantly augment attack response
coordination, permitting such tasks as population of black- or
white-listed source addresses; address or prefix group aliasing;
exchange of incident reports; and other hinting or configuration
supplementing attack response.

As the resilience requirements for the DOTS signal channel mandate
small signal message size, a separate, secure data channel
utilizing a reliable transport protocol MUST be used for bulk data
exchange.

GEN-004 Mitigation Hinting: DOTS clients may have access to attack
details which can be used to inform mitigation techniques.
Example attack details might include locally collected
fingerprints for an on-going attack, or anticipated or active
attack focal points based on other threat intelligence. DOTS
clients MAY send mitigation hints derived from attack details to
DOTS servers, in the full understanding that the DOTS server MAY
ignore mitigation hints. Mitigation hints MAY be transmitted
across either signal or data channel. DOTS server handling of
mitigation hints is implementation-specific.

GEN-005 Loop Handling: In certain scenarios, typically involving
misconfiguration of DNS or routing policy, it may be possible for
communication between DOTS agents to loop. Signal and data
channel implementations should be prepared to detect and terminate
such loops to prevent service disruption.

2.2. Signal Channel Requirements

SIG-001 Use of Common Transport Protocols: DOTS MUST operate over
common widely deployed and standardized transport protocols.
While connectionless transport such as the User Datagram Protocol
(UDP) [RFC0768] SHOULD be used for the signal channel, the
Transmission Control Protocol (TCP) [RFC0793] MAY be used if
necessary due to network policy or middlebox capabilities or
configurations.

SIG-002 Sub-MTU Message Size: To avoid message fragmentation and the
consequently decreased probability of message delivery over a
congested link, signaling protocol message size MUST be kept under
signaling Path Maximum Transmission Unit (PMTU), including the
byte overhead of any encapsulation, transport headers, and
transport- or message-level security.

DOTS agents SHOULD attempt to learn the PMTU through mechanisms
such as Path MTU Discovery [RFC1191] or Packetization Layer Path
MTU Discovery [RFC4821]. If the PMTU cannot be discovered, DOTS
agents SHOULD assume a PMTU of 1280 bytes. If IPv4 support on
legacy or otherwise unusual networks is a consideration and PMTU
is unknown, DOTS implementations MAY rely on a PMTU of 576 bytes,
as discussed in [RFC0791] and [RFC1122].

SIG-003 Bidirectionality: To support peer health detection, to
maintain an active signal channel, and increase the probability of
signal delivery during an attack, the signal channel MUST be
bidirectional, with client and server transmitting signals to each
other at regular intervals, regardless of any client request for
mitigation. Unidirectional messages MUST be supported within the
bidirectional signal channel to allow for unsolicited message
delivery, enabling asynchronous notifications between DOTS agents.

SIG-004 Channel Health Monitoring: DOTS agents MUST support exchange
of heartbeat messages over the signal channel to monitor channel
health. Peer DOTS agents SHOULD regularly send heartbeats to each
other while a mitigation request is active. The heartbeat
interval during active mitigation could be negotiable, but SHOULD
be frequent enough to maintain any on-path NAT or Firewall
bindings during mitigation.

To support scenarios in which loss of heartbeat is used to trigger
mitigation, and to keep the channel active, DOTS clients MAY
solicit heartbeat exchanges after successful mutual
authentication. When DOTS agents are exchanging heartbeats and no
mitigation request is active, either agent MAY request changes to
the heartbeat rate. For example, a DOTS server might want to
reduce heartbeat frequency or cease heartbeat exchanges when an
active DOTS client has not requested mitigation, in order to
control load.

Following mutual authentication, a signal channel MUST be
considered active until a DOTS agent explicitly ends the session,
or either DOTS agent fails to receive heartbeats from the other
after a mutually agreed upon retransmission procedure has been
exhausted. Because heartbeat loss is much more likely during
volumetric attack, DOTS agents SHOULD avoid signal channel
termination when mitigation is active and heartbeats are not
received by either DOTS agent for an extended period. In such
circumstances, DOTS clients MAY attempt to reestablish the signal
channel, but SHOULD continue to send heartbeats so that the DOTS
server knows the session is still alive. DOTS servers are assumed
to have the ability to monitor the attack, using feedback from the
mitigator and other available sources, and MAY use the absence of

attack traffic and lack of client heartbeats as an indication the
signal channel is defunct.

SIG-005   Channel Redirection: In order to increase DOTS operational
flexibility and scalability, DOTS servers SHOULD be able to
redirect DOTS clients to another DOTS server at any time. DOTS
clients MUST NOT assume the redirection target DOTS server shares
security state with the redirecting DOTS server. DOTS clients are
free to attempt abbreviated security negotiation methods supported
by the protocol, such as DTLS session resumption, but MUST be
prepared to negotiate new security state with the redirection
target DOTS server.

Due to the increased likelihood of packet loss caused by link
congestion during an attack, DOTS servers SHOULD NOT redirect
while mitigation is enabled during an active attack against a
target in the DOTS client's domain.

SIG-006   Mitigation Requests and Status: Authorized DOTS clients MUST
be able to request scoped mitigation from DOTS servers. DOTS
servers MUST send status to the DOTS clients about mitigation
requests. If a DOTS server rejects an authorized request for
mitigation, the DOTS server MUST include a reason for the
rejection in the status message sent to the client.

Due to the higher likelihood of packet loss during a DDoS attack,
DOTS servers SHOULD regularly send mitigation status to authorized
DOTS clients which have requested and been granted mitigation,
regardless of client requests for mitigation status.

When DOTS client-requested mitigation is active, DOTS server
status messages SHOULD include the following mitigation metrics:

*   Total number of packets blocked by the mitigation

*   Current number of packets per second blocked

*   Total number of bytes blocked

*   Current number of bytes per second blocked

DOTS clients MAY take these metrics into account when determining
whether to ask the DOTS server to cease mitigation.

A DOTS client MAY withdraw a mitigation request at any time,
regardless of whether mitigation is currently active. The DOTS
server MUST immediately acknowledge a DOTS client's request to
stop mitigation.

To protect against route or DNS flapping caused by a client
rapidly toggling mitigation, and to dampen the effect of
oscillating attacks, DOTS servers MAY allow mitigation to continue
for a limited period after acknowledging a DOTS client's
withdrawal of a mitigation request. During this period, DOTS
server status messages SHOULD indicate that mitigation is active
but terminating.

The initial active-but-terminating period is implementation- and
deployment- specific, but SHOULD be sufficiently long to absorb
latency incurred by route propagation. If the client requests
mitigation again before the initial active-but-terminating period
elapses, the DOTS server MAY exponentially increase the active-
but-terminating period up to a maximum of 300 seconds (5 minutes).
After the active-but-terminating period elapses, the DOTS server
MUST treat the mitigation as terminated, as the DOTS client is no
longer responsible for the mitigation.

SIG-007 Mitigation Lifetime: DOTS servers MUST support mitigations
   for a negotiated time interval, and MUST terminate a mitigation
   when the lifetime elapses. DOTS servers also MUST support renewal
   of mitigation lifetimes in mitigation requests from DOTS clients,
   allowing clients to extend mitigation as necessary for the
   duration of an attack.

   DOTS servers MUST treat a mitigation terminated due to lifetime
   expiration exactly as if the DOTS client originating the
   mitigation had asked to end the mitigation, including the active-
   but-terminating period, as described above in SIG-005.

   DOTS clients MUST include a mitigation lifetime in all mitigation
   requests.

   DOTS servers SHOULD support indefinite mitigation lifetimes,
   enabling architectures in which the mitigator is always in the
   traffic path to the resources for which the DOTS client is
   requesting protection. DOTS clients MUST be prepared to not be
   granted mitigations with indefinite lifetimes. DOTS servers MAY
   refuse mitigations with indefinite lifetimes, for policy reasons.
   The reasons themselves are out of scope. If the DOTS server does
   not grant a mitigation request with an indefinite mitigation
   lifetime, it MUST set the lifetime to a value that is configured
   locally. That value MUST be returned in a reply to the requesting
   DOTS client.

SIG-008 Mitigation Scope: DOTS clients MUST indicate desired
   mitigation scope. The scope type will vary depending on the

resources requiring mitigation. All DOTS agent implementations
MUST support the following required scope types:

* IPv4 prefixes in CIDR notation [RFC4632]

* IPv6 prefixes [RFC4291][RFC5952]

* Domain names [RFC1035]

The following mitigation scope types are OPTIONAL:

* Uniform Resource Identifiers [RFC3986]

DOTS servers MUST be able to resolve domain names and (when
supported) URIs. How name resolution is managed on the DOTS
server is implementation-specific.

DOTS agents MUST support mitigation scope aliases, allowing DOTS
clients and servers to refer to collections of protected resources
by an opaque identifier created through the data channel, direct
configuration, or other means. Domain name and URI mitigation
scopes may be thought of as a form of scope alias, in which the
addresses to which the domain name or URI resolve represent the
full scope of the mitigation.

If there is additional information available narrowing the scope
of any requested attack response, such as targeted port range,
protocol, or service, DOTS clients SHOULD include that information
in client mitigation requests. DOTS clients MAY also include
additional attack details. DOTS servers MAY ignore such
supplemental information when enabling countermeasures on the
mitigator.

As an active attack evolves, DOTS clients MUST be able to adjust
as necessary the scope of requested mitigation by refining the
scope of resources requiring mitigation.

A DOTS client may obtain the mitigation scope through direct
provisioning or through implementation-specific methods of
discovery. DOTS clients MUST support at least one mechanism to
obtain mitigation scope.

SIG-009 Mitigation Efficacy: When a mitigation request is active,
   DOTS clients SHOULD transmit a metric of perceived mitigation
   efficacy to the DOTS server. DOTS servers MAY use the efficacy
   metric to adjust countermeasures activated on a mitigator on
   behalf of a DOTS client.

   SIG-010 Conflict Detection and Notification: Multiple DOTS clients
      controlled by a single administrative entity may send conflicting
      mitigation requests as a result of misconfiguration, operator
      error, or compromised DOTS clients. DOTS servers in the same
      administrative domain attempting to honor conflicting requests may
      flap network route or DNS information, degrading the networks
      attempting to participate in attack response with the DOTS
      clients. DOTS servers in a single administrative domain SHALL
      detect such conflicting requests, and SHALL notify the DOTS
      clients in conflict. The notification SHOULD indicate the nature
      and scope of the conflict, for example, the overlapping prefix
      range in a conflicting mitigation request.

   SIG-011 Network Address Translator Traversal: DOTS clients may be
      deployed behind a Network Address Translator (NAT), and need to
      communicate with DOTS servers through the NAT. DOTS protocols
      MUST therefore be capable of traversing NATs.

      If UDP is used as the transport for the DOTS signal channel, all
      considerations in "Middlebox Traversal Guidelines" in [RFC8085]
      apply to DOTS. Regardless of transport, DOTS protocols MUST
      follow established best common practices established in BCP 127
      for NAT traversal [RFC4787][RFC6888][RFC7857].

2.3. Data Channel Requirements

   The data channel is intended to be used for bulk data exchanges
   between DOTS agents. Unlike the signal channel, the data channel is
   not expected to be constructed to deal with attack conditions. As
   the primary function of the data channel is data exchange, a reliable
   transport is required in order for DOTS agents to detect data
   delivery success or failure.

   The data channel provides a protocol for DOTS configuration,
   management. For example, a DOTS client may submit to a DOTS server a
   collection of prefixes it wants to refer to by alias when requesting
   mitigation, to which the server would respond with a success status
   and the new prefix group alias, or an error status and message in the
   event the DOTS client's data channel request failed.

   DATA-001 Reliable transport: Messages sent over the data channel
      MUST be delivered reliably, in order sent.

   DATA-002 Data privacy and integrity: Transmissions over the data
      channel are likely to contain operationally or privacy-sensitive
      information or instructions from the remote DOTS agent. Theft or
      modification of data channel transmissions could lead to
      information leaks or malicious transactions on behalf of the

sending agent (see [Section 4](#) below). Consequently data sent over
the data channel MUST be encrypted and authenticated using current
IETF best practices. DOTS servers MUST enable means to prevent
leaking operationally or privacy-sensitive data. Although
administrative entities participating in DOTS may detail what data
may be revealed to third-party DOTS agents, such considerations
are not in scope for this document.

DATA-003 Resource Configuration: To help meet the general and signal
channel requirements in [Section 2.1](#) and [Section 2.2](#), DOTS server
implementations MUST provide an interface to configure resource
identifiers, as described in SIG-007. DOTS server implementations
MAY expose additional configurability. Additional configurability
is implementation-specific.

DATA-004 Black- and whitelist management: DOTS servers MUST provide
methods for DOTS clients to manage black- and white-lists of
traffic destined for resources belonging to a client.

For example, a DOTS client should be able to create a black- or
whitelist entry, retrieve a list of current entries from either
list, update the content of either list, and delete entries as
necessary.

How a DOTS server authorizes DOTS client management of black- and
white-list entries is implementation-specific.

[2.4](#). Security Requirements

DOTS must operate within a particularly strict security context, as
an insufficiently protected signal or data channel may be subject to
abuse, enabling or supplementing the very attacks DOTS purports to
mitigate.

SEC-001 Peer Mutual Authentication: DOTS agents MUST authenticate
each other before a DOTS signal or data channel is considered
valid. The method of authentication is not specified, but should
follow current industry best practices with respect to any
cryptographic mechanisms to authenticate the remote peer.

SEC-002 Message Confidentiality, Integrity and Authenticity: DOTS
protocols MUST take steps to protect the confidentiality,
integrity and authenticity of messages sent between client and
server. While specific transport- and message-level security
options are not specified, the protocols MUST follow current
industry best practices for encryption and message authentication.

In order for DOTS protocols to remain secure despite advancements
in cryptanalysis and traffic analysis, DOTS agents MUST be able to
negotiate the terms and mechanisms of protocol security, subject
to the interoperability and signal message size requirements in
Section 2.2.

While the interfaces between downstream DOTS server and upstream
DOTS client within a DOTS gateway are implementation-specific,
those interfaces nevertheless MUST provide security equivalent to
that of the signal channels bridged by gateways in the signaling
path. For example, when a DOTS gateway consisting of a DOTS
server and DOTS client is running on the same logical device, the
two DOTS agents could be implemented within the same process
security boundary.

SEC-003 Message Replay Protection: To prevent a passive attacker
from capturing and replaying old messages, and thereby potentially
disrupting or influencing the network policy of the receiving DOTS
agent's domain, DOTS protocols MUST provide a method for replay
detection and prevention.

Within the signal channel, messages MUST be uniquely identified
such that replayed or duplicated messages can be detected and
discarded. Unique mitigation requests MUST be processed at most
once.

SEC-004 Authorization: DOTS servers MUST authorize all messages from
DOTS clients which pertain to mitigation, configuration,
filtering, or status.

DOTS servers MUST reject mitigation requests with scopes which the
DOTS client is not authorized to manage.

Likewise, DOTS servers MUST refuse to allow creation, modification
or deletion of scope aliases and black-/white-lists when the DOTS
client is unauthorized.

The modes of authorization are implementation-specific.

2.5. Data Model Requirements

A well-structured DOTS data model is critical to the development of
successful DOTS protocols.

DM-001 Structure: The data model structure for the DOTS protocol MAY
be described by a single module, or be divided into related
collections of hierarchical modules and sub-modules. If the data
model structure is split across modules, those distinct modules

        MUST allow references to describe the overall data model's
        structural dependencies.

    DM-002 Versioning: To ensure interoperability between DOTS protocol
        implementations, data models MUST be versioned. How the protocols
        represent data model versions is not defined in this document.

    DM-003 Mitigation Status Representation: The data model MUST provide
        the ability to represent a request for mitigation and the
        withdrawal of such a request. The data model MUST also support a
        representation of currently requested mitigation status, including
        failures and their causes.

    DM-004 Mitigation Scope Representation: The data model MUST support
        representation of a requested mitigation's scope. As mitigation
        scope may be represented in several different ways, per SIG-007
        above, the data model MUST be capable of flexible representation
        of mitigation scope.

    DM-005 Mitigation Lifetime Representation: The data model MUST
        support representation of a mitigation request's lifetime,
        including mitigations with no specified end time.

    DM-006 Mitigation Efficacy Representation: The data model MUST
        support representation of a DOTS client's understanding of the
        efficacy of a mitigation enabled through a mitigation request.

    DM-007 Acceptable Signal Loss Representation: The data model MUST be
        able to represent the DOTS agent's preference for acceptable
        signal loss when establishing a signal channel, as described in
        GEN-002.

    DM-008 Heartbeat Interval Representation: The data model MUST be
        able to represent the DOTS agent's preferred heartbeat interval,
        which the client may include when establishing the signal channel,
        as described in SIG-003.

    DM-009 Relationship to Transport: The DOTS data model MUST NOT
        depend on the specifics of any transport to represent fields in
        the model.

3. Congestion Control Considerations

3.1. Signal Channel

   As part of a protocol expected to operate over links affected by DDoS
   attack traffic, the DOTS signal channel MUST NOT contribute
   significantly to link congestion. To meet the signal channel

requirements above, DOTS signal channel implementations SHOULD
support connectionless transports. However, some connectionless
transports when deployed naively can be a source of network
congestion, as discussed in [RFC8085]. Signal channel
implementations using such connectionless transports, such as UDP,
therefore MUST include a congestion control mechanism.

Signal channel implementations using TCP may rely on built-in TCP
congestion control support.

3.2. Data Channel

As specified in DATA-001, the data channel requires reliable, in-
order message delivery. Data channel implementations using TCP may
rely on the TCP implementation's built-in congestion control
mechanisms.

4. Security Considerations

This document informs future protocols under development, and so does
not have security considerations of its own. However, operators
should be aware of potential risks involved in deploying DOTS. DOTS
agent impersonation and signal blocking are discussed here.
Additional DOTS security considerations may be found in
[I-D.ietf-dots-architecture] and DOTS protocol documents.

Impersonation of either DOTS server or DOTS client could have
catastrophic impact on operations in either domain. If an attacker
has the ability to impersonate a DOTS client, that attacker can
affect policy on the network path to the DOTS client's domain, up to
and including instantiation of blacklists blocking all inbound
traffic to networks for which the DOTS client is authorized to
request mitigation.

Similarly, an impersonated DOTS server may be able to act as a sort
of malicious DOTS gateway, intercepting requests from the downstream
DOTS client, and modifying them before transmission to the DOTS
server to inflict the desired impact on traffic to or from the DOTS
client's domain. Among other things, this malicious DOTS gateway
might receive and discard mitigation requests from the DOTS client,
ensuring no requested mitigation is ever applied.

As detailed in Section 2.4, DOTS implementations require mutual
authentication of DOTS agents in order to make agent impersonation
more difficult. However, impersonation may still be possible as a
result of credential theft, implementation flaws, or compromise of
DOTS agents. To detect misuse, DOTS operators should carefully

monitor and audit DOTS agents, while employing current secure network
communications best practices to reduce attack surface.

Blocking communication between DOTS agents has the potential to
disrupt the core function of DOTS, which is to request mitigation of
active or expected DDoS attacks. The DOTS signal channel is expected
to operate over congested inbound links, and, as described in
Section 2.2, the signal channel protocol must be designed for minimal
data transfer to reduce the incidence of signal blocking.

5. IANA Considerations

This document does not require any IANA action.

6. Contributors

Mohamed Boucadair
   Orange

   mohamed.boucadair@orange.com

Flemming Andreasen
   Cisco Systems, Inc.

   fandreas@cisco.com

Dave Dolson
   Sandvine

   ddolson@sandvine.com

7. Acknowledgments

Thanks to Roman Danyliw and Matt Richardson for careful reading and
feedback.

8. References

8.1. Normative References

[RFC0768] Postel, J., "User Datagram Protocol", STD 6, RFC 768,
           DOI 10.17487/RFC0768, August 1980,
           <https://www.rfc-editor.org/info/rfc768>.

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791,
           DOI 10.17487/RFC0791, September 1981,
           <https://www.rfc-editor.org/info/rfc791>.

   [RFC0793] Postel, J., "Transmission Control Protocol", STD 7,
              RFC 793, DOI 10.17487/RFC0793, September 1981,
              <https://www.rfc-editor.org/info/rfc793>.

   [RFC1035] Mockapetris, P., "Domain names - implementation and
              specification", STD 13, RFC 1035, DOI 10.17487/RFC1035,
              November 1987, <https://www.rfc-editor.org/info/rfc1035>.

   [RFC1122] Braden, R., Ed., "Requirements for Internet Hosts -
              Communication Layers", STD 3, RFC 1122,
              DOI 10.17487/RFC1122, October 1989,
              <https://www.rfc-editor.org/info/rfc1122>.

   [RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191,
              DOI 10.17487/RFC1191, November 1990,
              <https://www.rfc-editor.org/info/rfc1191>.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <https://www.rfc-editor.org/info/rfc2119>.

   [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform
              Resource Identifier (URI): Generic Syntax", STD 66,
              RFC 3986, DOI 10.17487/RFC3986, January 2005,
              <https://www.rfc-editor.org/info/rfc3986>.

   [RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing
              Architecture", RFC 4291, DOI 10.17487/RFC4291, February
              2006, <https://www.rfc-editor.org/info/rfc4291>.

   [RFC4632] Fuller, V. and T. Li, "Classless Inter-domain Routing
              (CIDR): The Internet Address Assignment and Aggregation
              Plan", BCP 122, RFC 4632, DOI 10.17487/RFC4632, August
              2006, <https://www.rfc-editor.org/info/rfc4632>.

   [RFC4787] Audet, F., Ed. and C. Jennings, "Network Address
              Translation (NAT) Behavioral Requirements for Unicast
              UDP", BCP 127, RFC 4787, DOI 10.17487/RFC4787, January
              2007, <https://www.rfc-editor.org/info/rfc4787>.

   [RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU
              Discovery", RFC 4821, DOI 10.17487/RFC4821, March 2007,
              <https://www.rfc-editor.org/info/rfc4821>.

   [RFC6888] Perreault, S., Ed., Yamagata, I., Miyakawa, S., Nakagawa,
              A., and H. Ashida, "Common Requirements for Carrier-Grade
              NATs (CGNs)", BCP 127, RFC 6888, DOI 10.17487/RFC6888,
              April 2013, <https://www.rfc-editor.org/info/rfc6888>.

   [RFC7857] Penno, R., Perreault, S., Boucadair, M., Ed., Sivakumar,
              S., and K. Naito, "Updates to Network Address Translation
              (NAT) Behavioral Requirements", BCP 127, RFC 7857,
              DOI 10.17487/RFC7857, April 2016,
              <https://www.rfc-editor.org/info/rfc7857>.

   [RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage
              Guidelines", BCP 145, RFC 8085, DOI 10.17487/RFC8085,
              March 2017, <https://www.rfc-editor.org/info/rfc8085>.

   [RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6
              Address Text Representation", RFC 5952,
              DOI 10.17487/RFC5952, August 2010,
              <https://www.rfc-editor.org/info/rfc5952>.

8.2. Informative References

   [I-D.ietf-dots-architecture]
              Mortensen, A., Andreasen, F., Reddy, T.,
              christopher_gray3@cable.comcast.com, c., Compton, R., and
              N. Teague, "Distributed-Denial-of-Service Open Threat
              Signaling (DOTS) Architecture", draft-ietf-dots-
              architecture-05 (work in progress), October 2017.

   [I-D.ietf-dots-use-cases]
              Dobbins, R., Migault, D., Fouant, S., Moskowitz, R.,
              Teague, N., Xia, L., and K. Nishizuka, "Use cases for DDoS
              Open Threat Signaling", draft-ietf-dots-use-cases-09 (work
              in progress), November 2017.

   [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston,
              A., Peterson, J., Sparks, R., Handley, M., and E.
              Schooler, "SIP: Session Initiation Protocol", RFC 3261,
              DOI 10.17487/RFC3261, June 2002,
              <https://www.rfc-editor.org/info/rfc3261>.

   [RFC7092] Kaplan, H. and V. Pascual, "A Taxonomy of Session
              Initiation Protocol (SIP) Back-to-Back User Agents",
              RFC 7092, DOI 10.17487/RFC7092, December 2013,
              <https://www.rfc-editor.org/info/rfc7092>.

   [RFC4732] Handley, M., Ed., Rescorla, E., Ed., and IAB, "Internet
              Denial-of-Service Considerations", RFC 4732,
              DOI 10.17487/RFC4732, December 2006,
              <https://www.rfc-editor.org/info/rfc4732>.

   [RFC4949] Shirey, R., "Internet Security Glossary, Version 2",
              FYI 36, RFC 4949, DOI 10.17487/RFC4949, August 2007,
              <https://www.rfc-editor.org/info/rfc4949>.

Authors' Addresses

   Andrew Mortensen
   Arbor Networks
   2727 S. State St
   Ann Arbor, MI 48104
   United States

   Email: amortensen@arbor.net

   Robert Moskowitz
   Huawei
   Oak Park, MI 42837
   United States

   Email: rgm@htt-consult.com

   Tirumaleswar Reddy
   McAfee
   Embassy Golf Link Business Park
   Bangalore, Karnataka 560071
   India

   Email: TirumaleswarReddy_Konda@McAfee.com