       Distributed Denial-of-Service Open Threat Signaling (DOTS) Telemetry
                              Specifications
                       draft-doron-dots-telemetry-00

Abstract

   This document aims to enrich DOTS Signaling with various telemetry
   attributes allowing optimal DDoS/DoS attack mitigation. The nature
   of the DOTS architecture is to allow DOTS Agents to be integrated in
   highly diverse environments. Therefore, the DOTS architecture
   imposes a significant challenge in delivering optimal mitigation
   services. The DOTS Telemetry covered in this document aims to
   provide all needed attributes and feedback signaled from DOTS Agents
   such that optimal mitigation services can be delivered based on DOTS
   Signaling.

Copyright Notice

Table of Contents

1. Introduction

   The DOTS signaling architecture (see [I-D.ietf-dots-architecture]) is
   designed to allow anti-DoS services for a vast number of networking,
   security and operational scenarios aimed to operate in diverse
   environments. This is a multi-dimensional challenge DOTS needs to
   meet in order to provide all the signaling requirements as derived
   from each environment's unique characteristics. The DOTS Client can
   be integrated within various elements with large diversity on their
   security capabilities. In a simple use case, the DOTS Client can be
   integrated in entities with a very basic understanding of the current
   security conditions, for example a customer portal with a user that
   is just realizing that something is "going wrong" with his service
   but is not aware of the main cause of the service degradation. Here,
   the DOTS Client can basically signal the need for mitigation along
   with its identification attributes. In a more advanced use case, the
   DOTS Client can be integrated within DDoS/DoS attack mitigators (and
   their control and management environments) or network and security
   elements that have been actively engaged with ongoing attacks. The
   DOTS Client mitigation environment determines that it is no longer
   possible or practical for it to handle these attacks. This can be
   due to lack of resources or security capabilities, as derived from
   the complexities and the intensity of these attacks. In this
   circumstance the DOTS Client has invaluable knowledge about the
   actual attacks that need to be handled by the DOTS Server. By
   enabling the DOTS Client to share this comprehensive knowledge of an
   ongoing attack, the DOTS Server can dramatically increase its
   abilities to accomplish successful mitigation. While the attack is
   being handled by the DOTS Server associated mitigation resources, the
   DOTS Server has the knowledge about the ongoing attack mitigation.
   The DOTS Server can share this information with the DOTS Client so
   that the Client can better comprehend and evaluate the actual
   mitigation realized. Both DOTS Client and DOTS Server can benefit
   this information by presenting various information in relevant
   management, reporting and portal systems.

   "DOTS Telemetry" is defined as the collection of attributes
   characterizing the actual attacks that have been detected and
   mitigated. The DOTS Telemetry is an optional set of attributes that
   can be signaled in the various DOTS protocol messages. The DOTS
   Telemetry can be optionally sent from the DOTS Client to Server and
   vice versa.

   This document aims to define all the required DOTS Telemetry
   attributes in order to use DOTS Signal and Data Channels for DOTS
   Telemetry signaling. Due to the diversity of environments DOTS
   Agents are designed to be integrated within, the DOTS Telemetry
   attributes (all of them as a whole, or some of them) are not

mandatory fields in any type of DOTS protocol message. Nevertheless,
when DOTS Telemetry attributes are available to the DOTS Agent it MAY
signal the attributes in order to optimize the overall service
provisioned using DOTS. Other basic minimum set of DOTS mandatory
signaling attributes (like "targeted entity", Targeted IP address and
so on), that are covered in other DOTS documents, are not reiterated
in this document. No assumption is made regarding the DOTS
Telemetry's actual collection methodology.

The document is divided into three logical parts: The first outlines
the need for DOTS Telemetry. The second covers the actual telemetry
attributes needed for providing comprehensive mitigation services.
The third describes the telemetry attributes needed for each of the
DOTS Signaling stages. Several typical use cases are also discussed
in detail.

1.1. Requirements Language

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 RFC 2119 [RFC2119].

1.2. Definition of Terms

This document uses the various terms defined in DOTS reuirements
document, see [I-D.ietf-dots-requirements].

2. Using the DOTS telemetry for a successful mitigation

The cyber security battle between the adversary and security
countermeasures is an everlasting fight. The DoS/DDoS attacks have
become more vicious and sophisticated in almost all aspects of their
maneuvers and malevolent intentions. IT organizations and service
providers are facing DoS/DDoS attacks that fall into two broad
categories: Network/Transport layer attacks and Application layer
attacks. Network/Transport layer attacks target the victim's
infrastructure. These attacks are not necessarily aimed at taking
down the actual delivered services, but rather to eliminate various
network elements (routers, switches, FW, transit links, and so on)
from serving legitimate user traffic. Here the main method of the
attackers is to send a large volume or high PPS of traffic toward the
victim's infrastructure. Attack volumes may vary from a few 100
Mbps/PPS to 100s of Gbps or even Tbps. Attacks are commonly carried
out leveraging botnets and attack reflectors for amplification
attacks, such as NTP, DNS, SNMP, SSDP, and so on. Application layer
attacks target various applications. Typical examples include
attacks against HTTP/HTTPS, DNS, SIP, SMTP, and so on. However, all
valid applications with their ports open at network edges can be

attractive attack targets. Application layer attacks are considered
more complex and hard to categorize, therefore harder to detect and
mitigate efficiently.

To compound the problem, attackers also leverage multi-vectored
attacks. These merciless attacks are assembled from dynamic attack
vectors (Network/Application) and tactics. Here, multiple attack
vectors formed by multiple attack types and volumes are launched
simultaneously towards the victim. Multi-vector attacks are harder
to detect and defend. Multiple and simultaneous mitigation
techniques are needed to defeat such attack campaigns. It is also
common for attackers to change attack vectors only moments after a
successful mitigation, burdening their opponents with changing their
defense methods.

The ultimate conclusion derived from these real-life scenarios is
that modern attacks detection and mitigation are most certainly
complicated and highly convoluted tasks. They demand a comprehensive
knowledge of the attack attributes, the targeted normal behavior/
traffic patterns, as well as the attacker's on-going and past
actions. Even more challenging, retrieving all the analytics needed
for detecting these attacks is not simple to obtain with the
industry's current capabilities.

With all this in mind, when signaling a mitigation request, it is
most certainly beneficial for the DOTS Client to signal to the DOTS
Server any knowledge regarding ongoing attacks. This can happen in
cases where DOTS Clients are asking the DOTS Server for support in
defending against attacks that they have already detected and/or
mitigated. These actions taken by DOTS Agent are referred to as
"signaling the DOTS Telemetry". If attacks are already detected and
categorized, the DOTS Server, and his associated mitigation services,
can proactively benefit this information and optimize the overall
service delivered. It is important to note that DOTS Client and
Server detection and mitigation approaches can be different, and can
potentially outcome different results and attack classifications.
Therefore, the DDOS mitigation service must treat the ongoing attack
details from the Client as hints, and cannot completely rely or trust
the attack details conveyed by the DOTS client. Nevertheless, the
DOTS Telemetry should support the identification of such misalignment
conditions.

A basic requirement of security operation teams is to be aware and
get visibility into the attacks they need to handle. The DOTS Server
security operation teams benefit from the DOTS Telemetry, especially
from the reports of ongoing attacks. They use the DOTS Telemetry to
be prepared for attack mitigation and to assign the correct resources
(operation staff, networking and mitigation) for the specific

service. Similarly, security operation personnel at the DOTS Client
side ask for feedback about their requests for protection.
Therefore, it is valuable for the DOTS Server to share DOTS Telemetry
with the DOTS Client. Thus mutual sharing of information is crucial
for "closing the mitigation loop" between the Client and Server. For
the Server side teams, it is important to realize that "the same
attacks that I am seeing are those that my client is asking me to
mitigate?." For the Client side, it is important to realize that the
Clients receive the required service. For example: understanding
that "I asked for mitigation of two attacks and my Server detects and
mitigates only one...". Cases of inconsistency in attack
classification between DOTS Client and Server can be high-lighted,
and maybe handled, using the DOTS Telemetry various attributes.

In addition, management and orchestration systems, at both Client and
Server side, can potentially use DOTS Telemetry as a feedback to
automate various control and management activities derived from
ongoing information signaled.

Should the DOTS Server's mitigation resources have the capabilities
to facilitate the DOTS Telemetry, the Server adopts its protection
strategy and activates the required countermeasures immediately. The
overall results of this adoption are optimized attack mitigation
decisions and actions.

The DOTS Telemetry can also be used to tune the mitigators with the
correct state of the attack. During the last few years, DDoS/DoS
attack detection technologies have evolved from threshold-based
detection (that is, cases when all or specific parts of traffic cross
a pre-defined threshold for a certain period of time is considered as
an attack) to an "anomaly detection" approach. In anomaly detection,
the main idea is to maintain rigorous learning of "normal" behavior
and where an "anomaly" (or an attack) is identified and categorized
based on the knowledge about the normal behavior and a deviation from
this normal behavior. Machine learning approaches are used such that
the actual "traffic thresholds" are "automatically calculated" by
learning the protected entity normal traffic behavior during peace
time. The normal traffic characterization learned is referred to as
the "normal traffic baseline". An attack is detected when the
victim's actual traffic is deviating from this normal baseline.

In addition, the subsequent activities toward mitigating the attack
are much more challenging. The ability to distinguish legitimate
traffic from attacker traffic on a per packet basis is complex. This
complexity originates in the fact that the packet itself may look
"legitimate" and no attack signature can be identified. The anomaly
can be identified only after detailed statistical analysis. DDoS/DoS
attack mitigators use the normal baseline during the actual

mitigation of an attack to identify and categorize the expected
appearance of a specific traffic pattern. Particularly the
mitigators use the normal baseline to recognize the "level of
normality" needs to be achieved during the various mitigation
process.

Normal baseline calculation is performed based on continuous learning
of the normal behavior of the protected entities. The minimum
learning period varies from hours to days and even weeks, depending
on the protected application behavior. The baseline cannot be
learned during active attacks because attack conditions do not
characterize the protected entities' normal behavior.

If the DOTS Client has calculated the normal baseline of its
protected entities, signaling this attribute to the DOTS Server along
with the attack traffic levels is significantly valuable. The DOTS
Server benefits from this telemetry by tuning its mitigation
resources with the DOTS Client's normal baseline. The mitigators use
the baseline to familiarize themselves with the attack victim's
normal behavior and target the baseline as the level of normality
they need to achieve. Consequently, the overall mitigation
performances obtained are dramatically improved in terms of time to
mitigate, accuracy, false-negative, false-positive, and other
measures. Mitigation of attacks without having certain knowledge of
normal traffic can be inaccurate at best. This is especially true
for DOTS environments where it is assumed that there is no universal
DDoS attack scale threshold triggering an attack across
administrative domains (see [I-D.ietf-dots-architecture]). In
addition, the highly diverse types of use-cases where DOTS Clients
are integrated also emphasize the need for knowledge of Client
behavior. Consequently, common global thresholds for attacks
detection practically cannot be realized. Each client can have his
own levels of traffic and normal behavior. Without facilitate
baseline signaling, it can be very difficult for Server to detect and
mitigate the attacks accurately. It is important to emphasize that
it is practically impossible for the Server's mitigators to calculate
the normal baseline, in cases they do not have any knowledge of the
traffic beforehand. In addition, baseline learning requires a period
of time that cannot be afforded during active attack.

As mentioned above, the task of isolating legitimate from attacker
traffic is extremely difficult to achieve. A common mechanism that
DDoS/DoS mitigators use to achieve such a distinction is to
authenticate source IP addresses that send traffic towards protected
entities. The source IP address can be authenticated as legitimate
or as a malicious BOT. Traffic from a BOT can be discarded or can be
rate-limited. Authentication can be performed using various
techniques; actively sending various challenges towards source IP

addresses is a common method. SYN Cookies, CAPTCHA, cryptographic
puzzle and others are examples of challenge-response tests used by
mitigators to determine whether the user is legitimate or a BOT.
Most certainly, building a list of authenticated source IP addresses
is a task that consumes resources and takes a long period of time to
construct. If the DOTS Client has already built a list of
authenticated IP addresses, the DOTS Server can use this list to
safely serve these IP addresses without any further need to re-
authenticate them. It is important to mention that "authenticated
IPs" are different from IP addresses in a "white list". This is
mainly because the authenticated IPs addresses are not predefined and
are not known upfront to the DOTS Agents. In addition, a source IP
address is treated as an authenticated IP address for a limited
period of time.

During a high volume attack, DOTS Client pipes can be totally
saturated. The Client asks the Server to handle the attack upstream
so that DOTS Client pipes return to a reasonable load level. At this
point, it is essential to ensure that the DOTS Server does not
overwhelm the DOTS Client pipes by sending back "clean traffic", or
what it believes is "clean". This can happen when the Server has not
managed to detect and mitigate all the attacks launched towards the
Client. In this case, it can be valuable to Clients to signal to
Server the "Total pipe capacity", which is the level of traffic the
Clients can absorb from the upstream Server. Dynamic updating of the
condition of pipes between DOTS Agents while they are under a DDoS
attack is essentially. For example, for cases of multiple DOTS
Clients share the same physical connectivity pipes. It is important
to note, that the term "pipe" noted here does not necessary represent
physical pipe, but rather represents the current level of traffic
Client can observe from Server. The Server should activate other
mechanisms to ensure it does not saturate the Client's pipes
unintentionally. The Rate Limiter can be a reasonable candidate to
achieve this objective; the Client can ask for the type of traffic
(such as ICMP, UDP, TCP port 80) it prefers to limit.

To summarize, timely and effective signaling of up-to-date DOTS
telemetry to all elements involved in the mitigation process is
essential and absolutely improves the overall service effectiveness.
Bi-directional feedback between DOTS elements is required for the
increased awareness of each party, supporting superior and highly
efficient attack mitigation service.

3. DOTS Telemetry attributes

   This section outlines the set of DOTS Telemetry attributes. The
   ultimate objective of these attributes is to allow for the complete
   knowledge of attacks and the various particulars that can best

characterize attacks. This section presents the attributes required
for each stage of the DOTS Signaling protocol (see
[I-D.reddy-dots-signal-channel] and
[I-D.nishizuka-dots-inter-domain-mechanism]). Other way of using
telemetry attributes is allowing DOTS Server to receive relevant DOTS
Telemetry before the actual attacks are launched using the DOTS Data
Channel [I-D.reddy-dots-signal-channel].

The description and motivation behind each attribute were presented
in previous sections in this document. The data model and the actual
integration within the DOTS Protocol are out of scope of this
document. It is expected that the following attributes will be
covered in any of the DOTS Protocol and DOTS Data Model standards.
As explained in previous sections, the DOTS Telemetry attributes are
optionally signaled and therefore SHOULD NOT be treated as mandatory
fields in any DOTS protocol messages.

3.1. Pre-mitigation DOTS Telemetry attributes

The Pre-mitigation telemetry attributes MAY be signaled from the DOTS
Client to the DOTS Server as part of the initiation of a DOTS service
request or during peace time using the DOTS Data Channel. Can be
signaled during a "Mitigation Request" (see also
[I-D.nishizuka-dots-inter-domain-mechanism]) session, or as part of
the "POST request" DOTS Signal (see also
[I-D.reddy-dots-signal-channel]). The following attributes are
required:

3.1.1. "Normal Baselines" of legitimate traffic

Average, x percentile and peak values of "Total traffic normal
baselines". PPS and BPS of the traffic are required.

[[EDITOR'S NOTE: We request feedback from the working group about
possible types of baselines to be signaled.]]

3.1.2. "Total Attack Traffic volume"

Current and peak values of "Total attack traffic". PPS and BPS of
attack traffic are required.

3.1.3. "Attack Details"

Various information and details that describe the on-going attacks
that need to be mitigated by the DOTS Server. The Attack Details
need to cover well-known and common attacks (such as a SYN Flood)
along with new emerging or vendor-specific attacks. The following is
a suggestion for the required fields in the Attack Details (this

definition follows the CEF Common Event Formula event definition, see
also [CEF] for more description):

Vendor |Version| Attack ID| Attack Name| Attack Severity| Extension

Where Extension is a placeholder for additional fields. Examples for
such fields are: Attack Classification, for example UDP port 80,
Layer 7 attack signature - Regex with Layer 7 attack signatures.
These can be defined as relevant key value pairs. Common and well-
known attack IDs SHOULD be standardized, and the vendor-specific IDs
SHOULD be specifically defined by each vendor.

The DOTS server should only treat the attack details as hints, and
not as a strict attribute to comply to. Please see requirement
OP-004 in [I-D.ietf-dots-requirements].

[[EDITOR'S NOTE: We request feedback from the working group about
possible alternatives for presenting "Attack Details" and various
characteristics of attacks.]]

3.1.4. "Total pipe capacity"

The limit of traffic volume, in BPS and PPS. The DOTS Server SHALL
eliminate sending this back as clean traffic. This attribute
represents the DOTS Client's pipe limits.

3.1.5. List of already "Authenticated source IPs"

List of source IP addresses that the DOTS Client has already
identified as authenticated IP addresses.

[[EDITOR'S NOTE: We request feedback from the working group about the
way to support this kind of IP address list under various NAT
strategies deployed in the Client's network. The same support is
required for "white lists" and "black lists" referred to in several
DOTS drafts, see [I-D.reddy-dots-data-channel].]]

3.2. Client to Server Mitigation Status DOTS Telemetry attributes

The Mitigation Status telemetry attributes MAY be signaled from the
DOTS Client to the DOTS Server as part of the periodic mitigation
status update as realized by the Server. This can be signaled during
"Mitigation Efficacy Update" (see also
[I-D.nishizuka-dots-inter-domain-mechanism] session, or as part of
the - "PUT request" DOTS signal (see also
[I-D.reddy-dots-signal-channel]).

The following attributes are required:

3.2.1. Current "Total traffic volumes"

Current values of the total traffic, in BPS and PPS, that arrive at
the DOTS Client sites. In addition, the Peak and x percentile of
traffic, in BPS and PPS, MAY also be signaled.

3.2.2. Current "Total Attack Traffic"

The total attack traffic volume, bps and pps, that the DOTS Client
still sees during the active mitigation service. In addition, the
Peak and x percentile of traffic, in BPS and PPS, MAY also be
signaled.

3.2.3. "Mitigation Efficacy Factor"

A factor defining the overall Mitigation Efficacy from the Client
perspective. By way of suggestion, the Mitigation Efficacy Factor
can be defined as the current clean traffic ratio to the normal
baseline. Network and Application Performance Monitoring attributes
can also be considered here.

3.2.4. "Attack Details"

The overall attack details as observed from the DOTS Client
perspective. The same data models that will be defined for the Pre-
mitigation DOTS Telemetry can also be applicable here.

3.3. Server to Client Mitigation Status DOTS Telemetry attributes

The Mitigation Status telemetry attributes MAY be signaled from the
DOTS Server to the DOTS Client as part of the periodic mitigation
status update. This can be signaled during "Mitigation Status" (see
also [I-D.nishizuka-dots-inter-domain-mechanism] session, or as part
of the "GET request" DOTS Signal (see also
[I-D.reddy-dots-signal-channel]).

The following attributes are required:

3.3.1. Current "Mitigation Countermeasure status"

As defined in [I-D.ietf-dots-requirements], the actual mitigation
activities can include several countermeasure mechanisms. The DOTS
Server SHOULD signal the current operational status to each relevant
countermeasure. For example: Layer 4 mitigation active/inactive,
Layer 7 mitigation active/inactive, and so on. In addition to the
status of each countermeasure, the DOTS Server SHOULD also signal: A
list of attacks detected by each countermeasure, and the statistics

for each countermeasure: Number of bytes/packets for each attack
handled, clean traffic volumes, dropped traffic.

It is important to maintain the AttackID among all DOTS
communications. The DOTS Client can further use this information for
reporting and service fulfillment purposes.

4. DOTS Telemetry Use-cases

DOTS Telemetry can certainly improve numerous DOTS Signaling use
cases. Nevertheless, DOTS Telemetry can be most beneficial when
dealing with relatively complex use cases where the DOTS Client is
integrated into environments with advanced detection and mitigation
abilities. In this section, typical use-cases are presented.
However, this list of use cases does not eliminate many other
scenarios, where the DOTS Telemetry is the pivot in bringing in
valuable use cases. It is expected that the DOTS Telemetry notions
will be added to the DOTS use cases [I-D.ietf-dots-use-cases]
documant.

4.1. Hybrid anti-DoS services use-case

In this common use case, a large enterprise deploys DDoS/DoS
mitigators as "in-line" devices on all the enterprise Internet peers.
The enterprise's security and operations teams are aware that in
cases of a large volume of attacks their Internet links can get
saturated. Therefore, having "in-line" mitigation devices deployed
will not help them in maintaining the service level that their
organization must maintain. In addition, they understand that they
are not capable of operating all the required actions to mitigate
multi-vector attacks. For these solid reasons, the enterprise IT
decides to purchase MSP (Managed Security Services) for "on demand"
DDoS/DoS mitigation services from a MSP Cloud provider. As part of
the anti-DoS service delivery, the enterprise and the MSP Cloud
provider have agreed to the required SLA. Also, they deploy a DOTS
Client at the enterprise premises and the DOTS Server at the MSP
cloud. During peace time, the enterprise mitigators build the
enterprise protected service's normal baseline. In cases of attacks
that can be mitigated "on-prem", the enterprise is able to deal with
the attack with its own resources. Should the attack become a large
volume attack and or also become multi-vector, the Internet links of
the organization, or even the mitigator links, get saturated. The
DOTS Client signals the need for aid in mitigating the on-going
attacks from the MSP's DOTS Server. In order to fulfil his SLA, the
MSP uses the DOTS Telemetry it received from the Client to assign the
adequate mitigation resources, tune the mitigators with the normal
baseline, assign the appropriate personnel to handle the enterprise
attacks, and so forth. The enterprise's security and operations team

uses the DOTS Telemetry they received from their DOTS Client to get
visibility into the actual mitigation performed "on the cloud" and
makes sure the service is fulfilled as expected.

4.2. MSP to MSP anti-DoS services use-case

This use case can be treated as a continuation of the previous use
case. The MSP Cloud provider operation team realizes that they have
some serious difficulties at their data centers and they are no
longer capable of serving the enterprise attacks. A back-to-back
DOTS Gateway is implemented at the MSP Cloud to allow redirection of
the attack to another MSP Cloud provider for the purpose of attack
mitigation. The same processes for using DOTS Telemetry are taken
here to ensure continuous service delivery. The same is true for
visibility into the actual service provided to the enterprise and to
the primary MSP Cloud provider.

5. Acknowledgements

Thanks to Yotam Ben-Ezra and Dennis Usle from Radware for their
contribution, careful reading and feedback.

6. IANA Considerations

This memo includes no request to IANA.

7. Security Considerations

To Be Added.

8. References

8.1. Normative References

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

8.2. Informative References

   [CEF]   ArcSight, Inc., "Common Event Format configuration
              guide.", 2009, <https://kc.mcafee.com/resources/sites/MCAF
              EE/content/live/CORP_KNOWLEDGEBASE/78000/KB78712/en_US/
              CEF_White_Paper_20100722.pdf>.

[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-00 (work in progress), July 2016.

[I-D.ietf-dots-requirements]
          Mortensen, A., Moskowitz, R., and T. Reddy, "Distributed
          Denial of Service (DDoS) Open Threat Signaling
          Requirements", draft-ietf-dots-requirements-02 (work in
          progress), July 2016.

[I-D.ietf-dots-use-cases]
          Dobbins, R., Fouant, S., Migault, D., Moskowitz, R.,
          Teague, N., and L. Xia, "Use cases for DDoS Open Threat
          Signaling", draft-ietf-dots-use-cases-01 (work in
          progress), March 2016.

[I-D.nishizuka-dots-inter-domain-mechanism]
          Nishizuka, K., Xia, L., Xia, J., Zhang, D., Fang, L.,
          christopher_gray3@cable.comcast.com, c., and R. Compton,
          "Inter-domain cooperative DDoS protection mechanism",
          draft-nishizuka-dots-inter-domain-mechanism-01 (work in
          progress), July 2016.

[I-D.reddy-dots-data-channel]
          Reddy, T., Wing, D., Boucadair, M., Nishizuka, K., and L.
          Xia, "Distributed Denial-of-Service Open Threat Signaling
          (DOTS) Data Channel", draft-reddy-dots-data-channel-00
          (work in progress), August 2016.

[I-D.reddy-dots-signal-channel]
          Reddy, T., Boucadair, M., Wing, D., and P. Patil,
          "Distributed Denial-of-Service Open Threat Signaling
          (DOTS) Signal Channel", draft-reddy-dots-signal-channel-01
          (work in progress), September 2016.

Authors' Addresses

   Ehud Doron
   Radware Ltd.
   Raoul Wallenberg Street
   Tel-Aviv 69710
   Israel

   Email: ehudd@radware.com

      Tirumaleswar Reddy
      Cisco Systems, Inc.
      Cessna Business Park, Varthur Hobli
      Sarjapur Marathalli Outer Ring Road
      Bangalore, Karnataka 560103
      India

      Email: tireddy@cisco.com

      Flemming Andreasen
      Cisco Systems, Inc.
      USA

      Email: fandreas@cisco.com

      Liang Xia (Frank)
      Huawei
      101 Software Avenue, Yuhuatai District
      Nanjing, Jiangsu 210012
      China

      Email: Frank.xialiang@huawei.com

      Kaname Nishizuka
      NTT Communications
      GranPark 16F, 3-4-1 Shibaura,
      Tokyo, Minato-ku 108-8118
      Japan

      Email: kaname@nttv6.jp