Moon to Earth DTN Communications Through Lunar Relay Satellites

Carlo Caini, Vincenzo Fiore
DEIS - Department of Electronics, Computer Science and Systems
University of Bologna
40136 Bologna, Italy
carlo.caini@unibo.it, vincenzo.fiore@studio.unibo.it

Abstract— **Although Delay-/Disruption- Tolerant Networking, which originated from research on an Interplanetary Internet, has enlarged its scope to encompass all challenged networks, space applications are still one of its most important application fields. This paper deals with DTN communication from Moon to Earth, based on the use of a lunar satellite acting as a "data-mule" to collect data from a Lander located on the far side of the Moon. To make the scenario more interesting and complex from the point of view of possible security threats, we assume that data must be transferred to a non-institutional user connected to the Space Agency Control Centre via Internet. In particular, the paper investigates the state-of-the-art ability of ION, the NASA implementation of the DTN Bundle Protocol (BP), to cope with the many challenges of the space scenario under investigation, such as intermittent links, low bandwidth, relatively high delays, network partitioning, DTN routing, interoperability between LTP and TCP BP Convergence layers and security threats. To this end, the first part of the paper contains three brief overviews of the DTN architecture, the Bundle Security Protocol and the ION implementation. These facilitate comprehension of the following sections, dedicated to a detailed description of the experiment scenario and, most essentially, to the in depth discussion of the numerical results obtained with the latest ION version (3.0).**

Keywords; **Delay-/Disruption- Tolerant Networking (DTN), Interplanetary Internet, ION, satellite communications.**

I. INTRODUCTION

Delay-/Disruption- Tolerant Networking originated from Interplanetary Internet research, where it was clear that the characteristics of deep space links would prevent the use of the ordinary Internet protocol stack [1]. Once it was recognized that the space challenges were actually common to other "challenged networks", the research was enlarged to include all of these [2]. Although no more exclusive, space communications (including satellite [3], [4]) are still one of the most interesting applications of the DTN concept [5], [6], [7].

The most important challenges in space environments are long delays and network partitioning. In particular, space links are usually intermittent and unavailable for most of the time, due to planet and spacecraft motion. To overcome the lack of a continuous path between source and destination, the only possible solution is to store data on intermediate nodes. This store-and-forward mechanism is at the basis of DTN Bundle Protocol (BP) architecture [8], [9], where the new Bundle layer, acting as an overlay, is added to selected nodes, and large blocks of data, called "bundles", are moved from DTN source to destination through DTN intermediate nodes with storage capacity. In challenged networks security plays an important role and space applications are no exception [10]. To ensure security, the Bundle Security Protocol (BSP) has been defined [11]. The bundle structure now may contain security blocks for both end-to-end and hop-by-hop security. The main characteristics of both DTN BP architecture and BSP security are summarized in the paper. A third brief overview describes ION, the BP implementation developed by NASA [12] used in our experiments. ION contains an implementation of Contact Graph Routing (CGR), an algorithm specifically designed to cope with intermittent scheduled links, typical of space environments [13]. In the latest version [14], ION also supports most BSP features.

This paper aims to assess the ability of DTN BP, and more specifically of ION, to cope with the many challenges of a Moon to Earth communication scenario, such as intermittent links, low bandwidth, relatively high delays, network partitioning, DTN routing, convergence layer interoperability and security threats.

The scenario studied here aims to be realistic but also as challenging as possible in terms of both routing and security. It considers the transmission of bundles (e.g. image files or other sensor data) from a Lander on the far side of the Moon, to Earth, through a lunar satellite acting as a "data-mule" [3], [8], a DTN application where data must not only be stored at intermediate nodes, but also carried from one geographical location to another, to make their transfer possible. Transfer from the satellite to the Earth destination involves a simple but also challenging routing problem, to test CGR. In order to include security threats and assess the effectiveness of BSP features, we have also assumed that the final destination is not under control of the space agency.

Numerical results of both routing and security experiments were obtained on a GNU/Linux testbed based on five KVM virtual machines. The latest ION version was used, in order to provide the reader with a study of the state-of-the-art ION implementation. Results are thoroughly reviewed before conclusions are drawn.

978-1-4673-2676-6/12/$31.00 ©2012 IEEE

II. DTN ARCHITECTURE

A. Bundle Protocol and Convergence Layer Adapters

The BP DTN architecture relies on the insertion in end points and some selected intermediate nodes of the new "Bundle layer" [8], located between Application and lower layers (usually Transport). The related protocol (the BP) [9] is in charge of the transmission of "bundles", i.e. large packets of data (e.g. an image file or a part of it), between DTN nodes. The BP interfaced with lower layers through "Convergence Layer Adapters" (CLAs), see Figure 1. Various CLAs have been defined; the most common are those for TCP, UDP and the Licklider Transmission Protocol (LTP) [15], [16], which is particularly suited to space links including cislunar ones [17]. In the DTN architecture, transport protocol end-to-end features are confined to one DTN hop, while end-to-end communication through multiple DTN hops is provided by the bundle layer, which acts as a store-and-forward overlay; DTN overlay and storage and other DTN features of interest for our experiments are analyzed below.

1) DTN as an overlay

By installing the BP in end-points and some intermediate nodes, the end-to-end path is divided into multiple DTN hops. In a heterogeneous network, e.g. a network that encompasses both terrestrial and space links, the intermediate DTN nodes are usually chosen at the border of each homogeneous network segment (namely, A, B and C in Figure 1) [3]. In this way on each DTN hop it is possible to use the transport protocol (or more generally, the protocol stack) best suited to it. This enables the use of specialized protocols, such as LTP, on the space links, where TCP or UDP could not cope with the long delays, low bandwidth and errors typical of the space environment.

2) Store-and-forward and custody option

The BP DTN architecture relies on a store-and-forward bundle transmission. Bundles, which can be much larger than ordinary IP packets, are first entirely received, then forwarded to the next "proximate" DTN node. If the link to the next node is not immediately available, as often happens in space communications, they must be locally stored even for relatively long periods of time.

The bundle source can ask next nodes to take "custody" of a bundle by setting the BP "custody option" bit in the bundle header. An intermediate DTN node may, or may not, "custody", i.e. the task of bundle retransmissions and reliable delivery to another custodian or to destination. If it accepts, it must notify the previous "custodian" (including the bundle source) and store bundles on persistent memory (e.g. on local hard disks), to prevent any loss of data caused by either software or hardware failure. Once custody is accepted by a following node, the old custodian is left free to cancel the bundle from its memory, which is advantageous when there are memory constraints or security issues.

3) Fragmentation

Fragmentation is one of the most characteristic features of DTN. It can be either "proactive", to match bundle dimension with limited contact volumes in intermittent links, or "reactive", to avoid retransmitting already acknowledged data in the presence of link disruption [8], [9].

4) Scheduled links

We will make great use of scheduled contacts, a DTN feature supported by ION for LTP CLA. Scheduled links are opened and closed by BP at precisely the beginning and end of known contacts, thus improving the link usage efficiency, especially when links are relatively short. This is essential in space communications, which are often characterized by intermittent connectivity and scarce, and therefore precious, bandwidth resources.

B. Licklider Transmission Protocol

The LTP has been designed to provide retransmission-based reliability over point-to-point links with extremely long RTT and/or frequent disruption. As such, it is suitable as convergence layer in space environment DTN architectures [15], [16]. In our experiments, we will use LTP for a variety of reasons: resilience to long RTTs, possibility of exactly matching the available bandwidth (LTP is rate controlled) and compatibility with scheduled links and CGR.

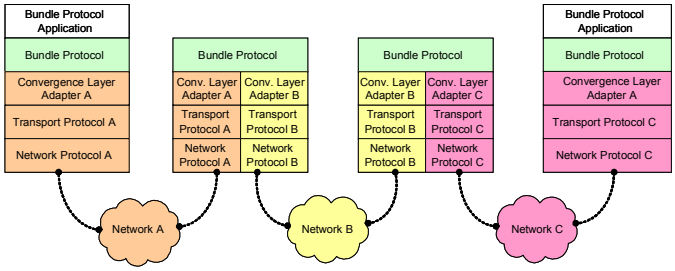

Figure 1: DTN Bundle Protocol architecture and protocol stack.

III. BUNDLE SECURITY PROTOCOL

Security is one of the most complex and important aspects to be considered in DTN, essentially because challenged network impairments, and in particular long delays and network partitioning, make the usual security solutions impractical. Although work on BP security is still in progress, its main features have already been formally defined in the BSP RFC [11].

There are at least three original aspects in BSP that deserve attention. First, there is a distinction between DTN nodes that support BP security and those that do not. Only the former are considered in BSP security; the latter should be transparent (i.e. they should be able to treat bundles with security features like any other). Second, both hop-by-hop and end-to-end security are included, with different solutions. Third, there is a distinction between protection of bundle payload and other bundle data (e.g. metadata), which can be protected by different passwords.

The bundle structure, defined in [9], consists of a series of elements called "blocks". To implement security, the BSP adds the following four [11]:

- Bundle Authentication Block (BAB)
- Payload Integrity Block (PIB)

- Payload Confidentiality Block (PCB)
- Extension Security Block (ESB)

BAB is used for bundle hop-by-hop authentication and integrity (see Figure 2), PIB is used to provide authentication and integrity (for the payload block only) over multiple DTN hops (usually, but not necessarily, end-to-end), see Figure 3. Note that PIB may be verified by any node in between the PIB security-source and the PIB security-destination, provided that it has access to the authentication cryptographic keys. PCB provides payload confidentiality between security-source and security-destination, analogously to PIB. Finally, ESB provides security (both confidentiality and integrity are recommended in the RFC) for non-payload blocks. Note that as ESB keys are different from other security blocks, they can be made accessible to some selected intermediate nodes, such as DTN routers, without compromising end-to-end security. Careful combined use of the four BSP blocks allows a high degree of flexibility in implementing security solutions.

Threats arising specifically from the use of DTN in space missions are analysed in [10]. In this paper we will focus on the use of security blocks, and in particular BABs, to prevent non-authorized users direct or indirect access to the space links and DOS (Denial-of-Service) attacks.

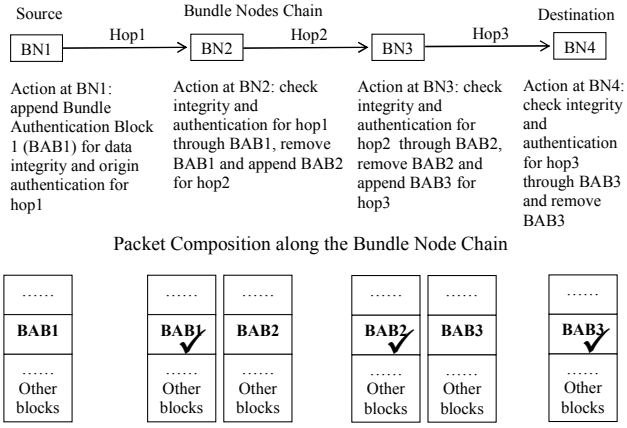

Figure 2 BAB for hop-by-hop authentication and integrity check. From [3]

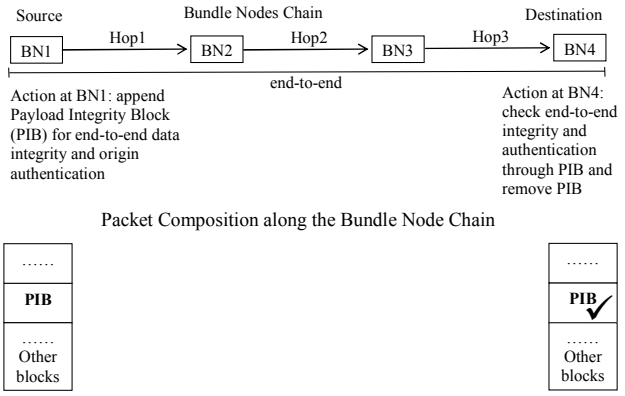

Figure 3 PIB end-to-end authentication and integrity check. From [3]

IV. ION Bundle Protocol implementation

ION (Interplanetary Overlay Network) is the DTN BP implementation developed by NASA JPL (Jet Propulsion Laboratory), with the contributions of Ohio and other Universities [12]. It is open source, available for GNU/Linux (and other platforms) [14], and is partially interoperable with DTN2, the BP reference implementation [18]. ION has been chosen here because was designed mainly for space applications and includes LTP, CGR and scheduled links, all of particular interest in our scenario. It also includes an implementation of BSP. At present the latest version fully documented is release 2.5.3. The code of 3.0 has however now released and we have used it in our experiments. Documentation is expected with release 3.1.

A. Contact Graph Routing

CGR is a dynamic routing algorithm designed to cope with intermittent scheduled connectivity [13]. In the space environment, communications between DTN nodes are active for only limited intervals of time, called "contact windows" in DTN terminology. Each contact offers the opportunity to transfer no more data than the "contact volume", given by the product of the link speed (in bit/s) and the contact window. Contact periods and contact volumes are assumed to be known a priori, because dependent on either DTN node motion or scheduled bandwidth allocation of space links. CGR exploits the knowledge of contacts windows and volumes to find the most suitable path from source to destination, following a complex algorithm, which is fully described in ION documentation. The key points are the following.

- Each node implementing CGR has a global knowledge of contacts (maybe because a "contact plan" has been provided to space nodes by a control centre).
- For each bundle CGR computes the full route to destination and select "the best" path; although the full route is computed, it is only used to locally select the most suitable "proximate" DTN node.
- The route is recomputed at each node implementing CGR.
- Routes are always recomputed for each new bundle, to cope with network dynamics.
- There are backoff mechanisms in case a bundle cannot actually be transferred during a contact because of any sort of impairment.
- Static routes may integrate CGR, but only if CGR is unable to find a route (because of either lack of suitable contacts or of information about a node in the contact plan).
- Mixed routing is allowed; e.g. CGR can compute route to destination for nodes in the space segment, while nodes outside can be reached computing routes to gateway nodes
- The criteria for "best" route selection may vary. In practice, they are "hard-coded" and depend on the ION release used. In the ION releases 2.5.x the "best" path is that which provides the shortest "expected delivery time" (see documentation embedded in the ION package [14]).

In releases 2.5.x there were the following limitations in CGR use. Some of them, however, have been now removed, as discussed in the Experiments section.

- CGR works only for nodes that use the ipn name scheme (mainly used by NASA); nodes with the alternative URI scheme can only by reached through static routes.
- Contacts used by CGR imply the use of scheduled links, which are available only for LTP CLA; links with TCP CLA are not supported, although nodes connected through TCP CLA can be reached via gateways.
- The decision taken by CGR in 2.5.x was not always optimal; for example, last hop continuous contacts (e.g. wired terrestrial links) were never used if there was an alternative path with an intermittent last hop.
- CGR path computation takes account of data already scheduled for transmission to "proximate" nodes. This "residual volume" check is limited to proximate nodes only, which at least in principle could lead to erroneous transmission towards nodes followed by links that are congested or with very limited capacity. Congestion control in DTN networks is a very difficult problem with only few solutions presented in the literature [19].

B. *Security features*

ION is going to offer full support to BSP. Until 2.5.3 this support was limited to BAB, now the support to PIB and PCB has been added, although documentation is still on going.

V. Moon to Earth Communication Scenario

We consider a Moon to Earth communication scenario, characterized by the presence of both space and terrestrial intermediate nodes, acting as relays. The mixed environment (space-terrestrial) considered here, although simple, is demanding from the point of view of ION because of the variety of challenges; in particular, it offers the opportunity to test the joint use of many of the most advanced ION features (CGR, scheduled links, CLA support and interoperability, security) in a case of real practical interest.

A. *Topology and applications*

The topology of our scenario is summarized in Figure 4. It consists of five DTN nodes: a Moon lander, a satellite orbiting the Moon, a Mission Control Centre (MCC), an auxiliary terrestrial Gateway Station, and finally a node, connected trough Internet, representing a possible non-institutional user. Intermittent space links are denoted by dotted lines, terrestrial wired links by continuous ones.

Starting from this topology, many applications can be considered. The reference one is the transfer of files (e.g. images or other sensor data) from the Lander to the MCC or to the non-institutional user. Bundles necessarily go through a space relay, because we assume the lander located on the far side of the Moon; bundles can reach the MCC either directly or through a terrestrial Gateway located far from the MCC, to increase the chances of contact with the lunar satellite. To add complexity and security challenges, we assume that bundles are directed to a destination node not belonging to the space institution that manages the other space and terrestrial assets. This external user is connected through Internet and could have "booked" specific Moon images or other sensor data from the space agency. It could be a University or another research centre, or even a private user.

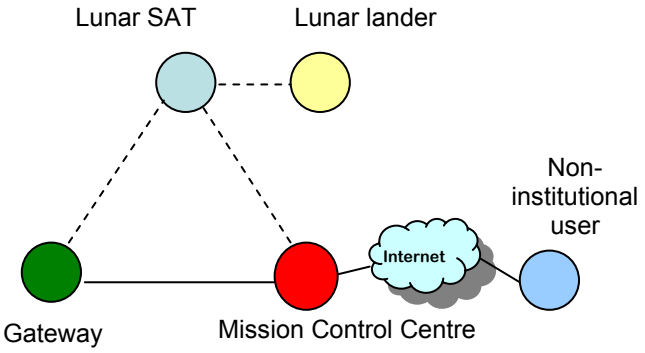

Figure 4: Topology of the Moon to Earth scenario considered. Dotted lines denote scheduled space links; continuous lines denote continuous wired links

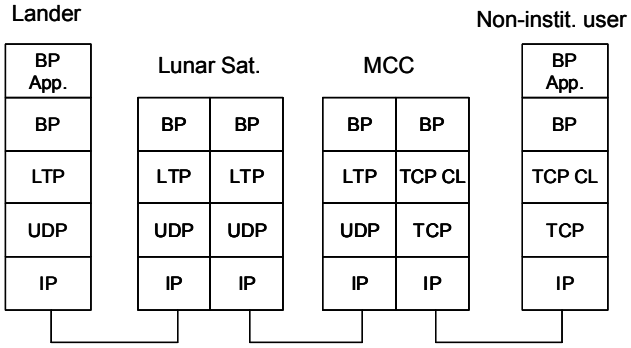

Figure 5: Protocol stack used in the experiments (in case of direct tranfer from the lunar satellite to MCC).

TABLE I. Contact Plan Excerpt

Link	*Contact#*	*Start-stop time (s)*	*Speed (downlink)*	*Volume (capacity)*
Lander-Sat	1	20-40	128kbit/s	320 kB
	2	100-120	128kbit/s	320 kB
Sat GW	1	70-80	1Mbit/	
	2	160-170	1Mbit/s	1.25 MB
Sat-MCC		150-180	1Mbit/s	3.75MB
GW–MCC	dummy	1-300 (TCPcont.)	10Mbit/s	

VI. Experiments

We will distinguish below between experiments aimed at assessing the functionality of ION features related to BP (CGR, LTP and TCP CLAs), and tests related to BSP security features. All tests have been carried out on a testbed consisting of five GNU/Linux virtual machines running the very recent ION 3.0, with CGR enhancements and stronger BSP support.

A. Contact plans, CGR routing and CLA interoperability

In the scenario summarized before we consider the transfer of ten bundles of 50 kB from the Lander to the non-institutional user. Note that from the lunar satellite to MCC two alternative routes are possible: one direct, the other through the Gateway station (Figure 4). All the nodes are DTN nodes; LTP is used on the space links (dotted lines) and TCP on the terrestrial ones (continuous lines). For the direct route, this leads to the protocol stack of Figure 5. The satellite to Earth links have a one way delay of 1.3s, while on other links is negligible; all the links are loss free. Space links are intermittent and their characteristics are summarized in the contact plan excerpt presented in Table I. Note that contact lengths and intervals have been deliberately shortened for testing purposes. Moreover, to emulate a continuous link between GW and MCC, we had to insert a sort of "dummy" contact window between these two nodes in the ION contact plan. This is necessary because CGR has not a special syntax for continuous links.

Regarding knowledge of the contact plan, we assume that it is fully known only by the MCC, the Gateway station and the satellite. The Lander knowledge is limited to its contacts with the satellite relay, which is enough for all purposes, as the Lander can only communicate through it (it actually needs contacts only for managing its LTP intermittent link to the satellite). Finally, the non-institutional user is totally unaware of the contact plan, which, as well as being convenient in actual deployment, it is also fully justified by the fact that all the traffic must be routed to MCC through a continuous TCP link. In brief, the contact plan is used in all nodes but the non-institutional user. In this and in the Lander a "group" static route is used to define their sole proximate node (the satellite or the MCC, respectively) as the gateway to the rest of the network.

The aims of this first series of experiments are

- to test CGR ability to select the best route on the basis of contact plan excerpt given in Table I;
- to test effectiveness of the static routes (group instruction);
- to test interoperability between TCP and LTP convergence layers.

Results, i.e. time series of status reports, are presented in Figure 6. At the beginning of the experiment ten bundles of 50 kB are given to the BP by a BP application. They are taken into custody ("Cst Lander" series) and wait for the next contact available to the Lunar satellite. During the first contact, only six bundles are actually transferred and taken into custody on the satellite ("Cst on Sat"), in accordance with the 320 kB contact volume (see Table I). The inclination of bundle status report series is due to the limited bandwidth available (128kbit/s) on the Lander-Sat link. Then, from the satellite to the MCC, two routes are possible. CGR, which is invoked as soon as the bundle reaches the satellite, correctly selects the route via GW, which starts first (at 70s) and is the fastest, as the GW-MCC link is continuous. Note, however, that in previous CGR versions the presence of a continuous link on the second hop, from GW to MCC, would have penalized this route. Results presented here show that this penalization has now been removed, which is a significant improvement. Having reached the MCC ("Cst MCC"), the six bundles are then immediately transferred to the non-institutional user and delivered to the BP application running on the destination ("Dlv N.I. User"). The transfer from GW to the non-institutional user is fast, as it is performed through two TCP connections on continuous terrestrial links. When the first six bundles are delivered, the remaining four are still in custody on the Lander. They are transferred to the satellite during the second Lander-Sat contact and taken into custody as before ("Cst Sat" markers starting at 100s). This time CGR selects the direct route to MCC, which is the first to open (at 150s). Note that the fact that the concurrent Sat-GW second contact closes first (at 170s instead of 180s), no longer prevents CGR from taking the right decision, unlike previous versions. Having reached the MCC ("Cst MCC" at about 150s) the bundles are immediately transferred to the final destination and delivered to the BP application (Dlv N.I. User, at about 150s).

The results presented here are successful in every way: CGR, "group" instruction, LTP and TCP convergence layer interoperability. To obtain them, however, the authors had to overcome some difficulties with the ION configuration. In all cases, we were fully supported by the ION authors.

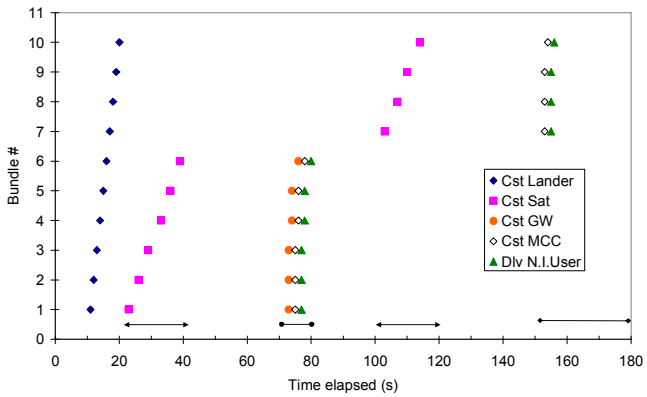

Figure 6: Status report logs of the Lander to non-institutional user bundle transfer.

B. BSP and security threats

Here the aim is to assess the BSP ability to tackle some significant security threats present in the scenario already considered (Figure 4). We will mainly focus on the use of BAB, but they also consider PIB and PCB. Note that all BSP features are as yet undocumented (but should be in the next release), which made our tests more demanding, but, hopefully, also more interesting to the reader. To the best of the authors' knowledge, the experiments described here are the first tests of the BSP ION implementation presented in the literature.

1) Direct access to space assets (masquerading)

Space assets are precious and thus must be carefully protected against any form of attack. For this reason only nodes in full control of the space agencies must be authorized to access them. This is why in our scenario we assume that any request for photo or data download from space assets must be directed to the MCC for approval and never to space assets directly. By contrast, direct transfer in the opposite direction, from the Lander to the non-institutional user must be possible. In particular, in our scenario it must be impossible for the non-institutional user to directly send any bundle to the satellite relay (the Lander is protected by its location on the far side of the Moon). For preventing this attack it is enough to use BAB authentication of space nodes. Any attempt to send bundles destined to the relay or to the Lander by the non-institutional user, pretending to be an authorized node like the MCC, is easily rejected by BAB checking at relay satellite.

We tested BAB on one hop, with full success both in the case of LTP and TCP convergence layer. This is an improvement over previous versions of ION, where the one hop BAB test failed in case of TCP CL, due to the difficulty in obtaining the right ipn address of the source, and thus in identifying the right key to use in BAB verification.

2) Direct access to space assets (Denial of Service attack)

Instead of trying to send bundle accepted, an attacker could try to exhaust the storage resources of space assets, to carry out a Denial of Service attack (bundle need to be stored at intermediate nodes) [20]. For this reason it is essential that bundles that fail the BAB verification rule be eliminated as soon as possible. This is done in recent ION versions, after a bug fix suggested by the paper authors. A different kind of DoS attack could try not to exhaust the memory, but to keep the node fully busy with security processing. To prevent this kind of attack, based on the sending of a large number of unauthorized bundles, a lightweight processing, like the cookie techniques presented in [20] could be used.

3) Indirect access to space assets

An attacker, like our non-institutional user, could try to exploit its authorized access to the MCC only, in order to send bundles to the satellite relay or the Lander. Such bundles, would pass the MCC BAB check, and if allowed to go further, would reach the satellite relay, which could not discard them because they would arrive with the BAB signature added by the authorized MCC node (see Figure 2) and not with that of the unauthorized source, as in the previous case of direct attack.

During our experiments we found that the incompatibility issue between BAB and TCP, now resolved on one hop tests, is unfortunately still present on two hop tests. However, with LTP all BAB tests proved successful.

Apart from residual BAB problems with TCP, there are two possible solutions to the indirect attack. Firstly, at the MCC node, to prevent forwarding of bundles sent by unauthorized nodes towards space assets. This can be done by enforcing the use of PIB, which although end-to-end, can be used by intermediate nodes (with the appropriate keys), like MCC, to check authenticity of the bundle payload. Secondly, the same PIB check could be made at destination node. This, however, would not prevent from the unauthorized use of space links. The best policy is likely to enforce PIB control and discard at both intermediate and destination nodes.

Our tests have proved that PIB functionalities work fine.

4) Tampering and eavesdropping

Unauthorized bundle payload modification (tampering) or reading (eavesdropping) can be counteracted by an appropriate use of PIB and PCB. The test of PCB was therefore needed to complete our BSP evaluation. Although we had no errors, the bundle payload appeared in plain after deciphering, independently of the deciphering key used. That could depend on the wanted exclusion of the ciphering algorithm in the open source version of ION.

VII. CONCLUSIONS

In this paper we have investigated Moon to Earth DTN communications through a lunar satellite able to collect data from a Lander on the far side of the Moon. The use of DTN BP is widely recognized as essential to cope with intermittent links and network partitioning typical of space environments. Thus, our paper mainly focused on the definition of a Moon to Earth scenario suitable for DTN BP experiments and on the study of the state-of-the-art ION BP implementation developed by NASA and Ohio University. Our experiments were carried out on ION 3.0 and evaluated many features at present undocumented, such as an enhanced CGR and the support of bundle payload authentication and encryption (PIB and PCB). Results show a major step forward with respect of previous ION versions, with significant improvements in particular on CGR reliability and on the completeness of security features. Some work is however still in order to improve compatibility of BSP with TCP CL, to update documentation and to streamline the use of configuration files for inexpert users.

ACKNOWLEDGMENTS

The authors would like to thank Scott Burleigh for his kind support during experiments with the present and previous versions of ION.

REFERENCES

[1] S. Burleigh, et al., “Delay-tolerant networking: An approach to interplanetary internet", IEEE Commun. Mag., vol. 41, no. 6, pp. 128–136, Jun. 2003.

[2] K. Fall, S. Farrell, "DTN: an architectural retrospective", IEEE J. Select. Areas in Commun., vol.26, no.5, pp. 828-836, June 2008.

[3] C. Caini, H. Cruickshank, S. Farrell, M. Marchese, “Delay- and Disruption-Tolerant Networking (DTN): An Alternative Solution for Future Satellite Networking Applications”, Proceedings of IEEE, Vol. 99, N. 11, pp.1980-1997, Nov. 2011.

[4] C. Caini, R. Firrincieli, “DTN and satellite communications”, in “Delay Tolerant Networks: Protocols and Applications”, Ed. A.Vasilakos, Y. Zhang, T. Spyropoulos, pp.283-pp.318, CRC press, Nov. 2011, New York.

[5] C. J. Krupiarz, E. H. Jennings, J. N. Pang, J. B Schoolcraft, J. B. Seguí, and J. L. Torgerson, “Spacecraft Data and Relay Management Using Delay Tolerant Networking,” AIAA 9th International Conference on Spacecraft Operations (SpaceOps), Rome, Italy, June 19-24, 2006.

[6] J. Wyatt, S. C. Burleigh, R. Jones, L. Torgerson, S. Wissler, “Disruption Tolerant Networking Flight Validation Experiment on NASA's EPOXI Mission,” SPACOMM, First International Conference on Advances in Satellite and Space Communications, 2009, Colmar, France, 20-25 July 2009, pp. 187-196.

[7] A.Jenkins, S.Kuzminsky, K.K.Gifford, R.L. Pitts, K.Nichols, “Delay/Disruption-Tolerant Networking: Flight test results from the international space station”. In Proc. of IEEE Aerospace Conference, 2010, pp.1-8, 2010.

[8] V. Cerf , A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, H. Weiss “Delay-Tolerant Networking Architecture”, Internet RFC 4838, Apr. 2007.

[9] K. Scott, S. Burleigh, “Bundle Protocl Specification”, Internet RFC 5050, Nov. 2007.

[10] W. Ivancic, “ Security Analysis of DTN Architecture and Bundle Protocol specification for Space-Based Networks”, in Proc. of IEEE Aerospace Conf., Big Sky Montana, 2010, pp. 1-12.

[11] S. Symington, S. Farrell, H. Weiss, P. Lovell, “Bundle Security Protocol Specification”, Internet RFC 6257, May 2011.

[12] S. Burleigh, "Interplanetary Overlay Network (ION) an Implementation of the DTN Bundle Protocol, In the Proc. of 4th IEEE Consumer Communications and Networking Conference, 2007, pp. 222-226.

[13] S. Burleigh, “Contact Graph Routing,” Internet-Draft, July 2010. work-in-progress http://tools.ietf.org/html/draft-burleigh-dtnrg-cgr.

[14] ION code: http://www.openchannelfoundation.org/projects/ION.

[15] M. Ramadas, S. Burleigh and S. Farrell, “Licklider Transmission Protocol – Motivation,” Internet RFC 5325, Sept. 2008.

[16] M. Ramadas, S. Burleigh and S. Farrell, “Licklider Transmission Protocol – Specification,” Internet RFC 5326, Sept. 2008.

[17] Ruhai Wang, Xuan Wu, Tiaotiao Wang T.Taleb, “Experimental Evaluation of Delay Tolerant Networking (DTN) Protocols for Long-Delay Cislunar Communications”, in Proc. of IEEE GLOBECOM, 2009, pp. 1-5, 2009.

[18] Internet Research Task Force DTN Reseach Group (DTNRG) web site: http://www.dtnrg.org/ Accessed 2010-09-28.

[19] I. Bisio, M. Cello, T .de Cola, M. Marchese, “Combined Congestion Control and Link Selection Strategies for Delay Tolerant Interplanetary Networks”, IEEE GLOBECOM 2009, pp. 1-6, 2009.

[20] G. Ansa, H. Cruickshank, Z. Sun and M. Al-Siyabi “A DOS-Resilient Design for Delay Tolerant Networks”, in Proc. of IWCMC 2011 Conf., Istanbul, Turkey, 2011, pp. 424-429.