Malicious user identification scheme for network coding enabled small cell environment

Vipindev Adat¹, Reza Parsamehr²,³, Ilias Politis¹, Christos Tselios¹ and Stavros Kotsopoulos¹

¹Wireless Communications Laboratory, University of Patras, Greece
{vipindev,ipolitis,tselios,kotsop}@ece.upatras.gr
²Instituto de Telecomunicaciones, Aveiro, Portugal
parsamehr.r@av.it.pt
³Universidad Politécnica de Madrid, Madrid, Spain
r.parsamehr@alumnos.upm.es

Abstract—Reliable communication over the wireless network with high throughput is a major target for the next generation communication technologies. Network coding can significantly improve the throughput efficiency of the network in a cooperative environment. The small cell technology and device to device communication make network coding an ideal candidate for improved performance in the fifth generation of communication networks. However, the security concerns associated with network coding needs to be addressed before any practical implementations. Pollution attacks are considered one of the most threatening attacks in the network coding environment. Although there are different integrity schemes to detect polluted packets, identifying the exact adversary in a network coding environment is a less addressed challenge. This paper proposes a scheme for identifying and locating adversaries in a dense, network coding enabled environment of mobile nodes. It also discusses a non-reputation protocol that will prevent adversaries from deceiving the network.

Index Terms—Network coding, 5G, Small cells, Security

I. INTRODUCTION

Next-generation communication networks address the challenge of providing high throughput in a dense heterogeneous network. Initial pilots of the fifth generation of wireless networks focus on achieving high quality of service even in remote areas with low energy consumption and end to end latency [1], [2]. This also leads to the paradigm shift of networks from base station-centric to device-centric approach. Small cells, capable of being set up on demand, form an integral part of this proposed environment. Device to device communication and cooperative environment of devices ensures that even the weakest user in the network will be fairly allotted with resources [3]. Further, the device to device communication makes it favorable to have network coding enabled small cell environments for enhanced bandwidth utilization [4]. This SEcure network Coding for Reduced Energy next (SECRET) generation mobile small cells, as shown in fig. 1, proposes a small cell environment supported by an SDN based core network. It also enables the participating nodes to have a device to device communication for high quality of experience and reduced overhead over the macro cell. Moreover, the SECRET small cells try to harness the advantages of network coding in the small cell environment for a resilient and efficient communication network. Network coding [5] advises the intermediate nodes to mix up the packets they receive and forward them to the next hops instead of the traditional store and forward approach. This helps to enhance the throughput and also reduces the latency in communication. Furthermore, the mixing of packets induces some inbuilt resistance to eavesdropping and wiretapping [6]. However, network coding introduces other security challenges like pollution attacks and entropy attacks due to this unique nature of mixing packets on the go [7]–[9]. These security challenges need to be tackled completely before going to the fully-fledged network coding enabled small cell deployments. Pollution attacks are considered as one of the most dreadful attacks in network coding environment. Preventing polluted packets from spreading across the network is of utmost priority because of the drastic reduction in throughput and efficiency that pollution attacks can cause in the network [10]. Furthermore, identifying the source of the attacker and isolating that malicious user in the network is the logical action to be performed after detecting a pollution attack. However, most of the integrity schemes [11]–[15] prevents polluted packets from being mixed with genuine packets and dropped by the genuine node who identify the attack but doesn’t make any effort to report the adversary node who send the malicious packet and inform other peer nodes about the presence of an adversary in the network. Very few integrity schemes [16]–[19] go to the extent of locating malicious users and publicize the presence of an adversary in the network. In a dense and mobile environment that 5G world addresses, it is important to identify adversary nodes and isolate them in the network as soon as possible and there are different scenarios of proactive and reactive malicious user mitigation schemes in related domains [20]–[22]. However, locating the adversary node becomes extremely challenging in this scenario. This paper addresses the challenge of locating malicious users...
trying to inject a polluted packet to the network coding enabled small cell environments and make the network aware of such adversary nodes. The integrity scheme proposed in [13] is followed for the identification of pollution attacks and then extended by modifying the non-repudiation transmission protocol (NRTP) [17] to the highly mobile and fluctuating network architecture. The proposed scheme ensures that even an intelligent and moving adversary node will also be properly identified if it continues to show malicious behavior. The remaining sections of the paper are structured as follows. Section II describes the motivation of the work and some of the related works relevant to the detection and identifying of adversaries in network coding. Section III presents the locating scheme we are proposing and followed by section IV analyzing different scenarios and verifying the proposed idea. Section V concludes the paper.

II. MOTIVATION AND RELATED WORKS

This section focus on describing the severity of pollution attacks and challenges in detection and identifying of adversaries. Further, it also discusses some of the relevant integrity schemes and locating schemes.

A. Pollution Attacks and Countermeasures

Intermediate nodes being capable of coding the received packets and sending them to the child nodes account for the increased network throughput in the network coding environment. More specifically in random linear network coding, a minimum number of linearly independent packets are required to get any meaningful decoding of the packets being received. This also makes it difficult for an eavesdropper to get any significant information by monitoring any particular link for a short time. However, this leads to the possibility of a malicious node changing some packets during transmission. This type of attack is commonly called pollution attacks or byzantine modification attacks. A single polluted packet can corrupt all the packets it is being mixed with during the transition and it can spread across the network resulting in a significant degradation in the performance. This makes it necessary to identify polluted packets at the earliest possible node. However, it is difficult for generally used cryptographic techniques to ensure the integrity of packets since the packets are being randomly coded in the intermediate nodes. Integrity schemes with homomorphic properties are used to defend any modification of packets in network coding.

Homomac [23] was one of the initial studies where the scope of homomorphic message authentication codes (MACs) are studied for integrity schemes. Agrawal and Boneh proposed an integrity scheme based on these homomorphic MACs which can verify any linear combination or the span of the original packets. Homomorphic MACs require less compu-
tational overhead compared to homomorphic signatures or hashes and there onwards inspired different integrity schemes based MACs. However, using MACs or tags to defend pollution attacks lead to another type of pollution, called tag pollution, where the adversary doesn’t pollute the data packets but instead attach bogus tags to the packet. This will lead to the dropping of genuine packets by genuine nodes due to incorrect tags which will again result in reduced throughput. Zhang et al. [11] proposed an integrity scheme which modified the idea of homomac and applied a signature over the MACs to ensure the tags can not be modified. Further, they considered a specific key distribution scheme to improve the security against colluding adversaries and reduced overhead for verification. Esfahani et al. [12] proposed a dual HMAC scheme for reducing the computational and communication overheads of previous schemes. A blockchain enhanced secure small cell environment was proposed in [14], [24] where the MAC scheme was adapted to the small cell environment with reduced overheads and simplified key distribution scheme. [13] presents an SDN based secure network coding enabled small cell where the central controller is used for tag sharing to prevent tag pollution attack as well as to nullify any advantage to colluding adversaries. In this paper, we extend this work to locate the adversary node and notify the presence of the adversary to the network.

B. Locating schemes for malicious users in network coding

Even though there are many schemes for detecting polluted packets, precisely identifying the malicious nodes and advertising the presence of adversary to other participating nodes is a less explored area. Most of the integrity schemes identify the polluted packets and drop them but leave it to other participating nodes to identify polluted packets by following the same steps on their own. In a dense environment of mobile nodes, it will be beneficial to notify the participating nodes about the presence of adversary so that other genuine nodes can immediately drop the incoming packets from such malicious nodes. However, it also presents some other challenges where an adversary can disarange a genuine node. Siavoshani et al. [16] proposed an integrity scheme which also extends to locating the malicious users using a central controller collecting information from all the participants and use network error-correcting codes to decide the presence and location of byzantine attackers. Le and Markopoulou proposed SpaceMac in [18] where a cooperative approach is used to locate the adversary nodes and homomorphic MAC schemes to ensure the integrity of packets in transition. Wang et al. [17] proposed an approach to identify malicious users in a peer to peer streaming network. They use the video formats to verify the packets integrity and concentrate on locating the adversaries unambiguously. This Malicious node Identification Scheme (MIS) was supported by the Non-Repudiation Transmission Protocol (NRTP) which prevents adversary nodes from disparaging benign nodes and also a checksum to precisely locate the adversaries. However, the NRTP proposed by Wang et al. requires the central controller to get specific secret keys from all participating nodes and provide all parent nodes with some shared secret to create a one-way hash function which the child node can verify. This system can introduce unbearable overheads in a mobile environment where the paths are unstable and the neighboring nodes will vary frequently. In a recent work, Parsamehr et al. [19] proposed a location aware IDPS for network coding enabled mobile small cell environment. This IDPS scheme requires extra communication between the participating nodes with the hotspot on top of the D2D passage of packets to ensure the security of the network and reduce considerable communication overhead. In this paper, we modify the non-repudiation transmission protocol to suit the requirements of a mobile environment and merge with the SDN based integrity scheme for network coding enabled mobile small cells.

III. SECURITY SCHEME FOR NETWORK CODING ENABLED MOBILE SMALL CELLS

The security scheme for network coding enabled mobile small cells consist of the integrity scheme based on homomorphic MACs and the adversary locating scheme.

A. Integrity scheme against pollution attacks

Any integrity scheme for the small cell environment needs to address a dense heterogeneous network of mobile devices. This makes it difficult to have a complex key distribution like c-cover free system. It has practical difficulties to pre-distribute the keys because of the mobile nodes and their frequently changing neighborhood. Further, the adversary can get the required keys by introducing a few new nodes to the network. Thus depending on key distribution to prevent colluding adversaries from making an advantage over the benign node is not advisable for dense networks. However, [13] proposes an SDN based integrity scheme where the tags are securely shared between the participant nodes through an alternative path along with the communication channel. The software-defined core network as discussed in the architecture can be considered as a secure entity supported by different security measures [25], [26] and the centralized SDN controller as a secure trusted authority for the whole network. Fig. 2 shows a simplified scenario for the security scheme. The security scheme follows the homomorphic MAC protocol to create tags over packets in a generation. These tags are attached to the packets as well shared with the central controller. Every receiving node verifies the tags using the keys they hold and further verifies the authenticity of tags by comparing the tags received through the channel with the tags stored in the central controller. This approach prevents both tag pollution attacks and data pollution attacks at the immediate genuine node. Further, it also prevents colluding adversaries from making any significant advantage over the security scheme. A detailed security analysis of the SDN based integrity scheme is discussed in [13].

B. Locating the adversary and Modified NRTP

The main challenges associated with the locating scheme for malicious nodes are fake alerts and the adversary node...
denying about having sent the corrupted packet. An adversary can report one of its parent nodes as an adversary by creating a fake alert and try to deceive the controller. On the other hand, an adversary can also deny sending a corrupted packet once the genuine node creates an alert. One approach to prevent this can be by using some public-key cryptography to ensure the authenticity of packets and reports. However, this involves significant overheads. The Non-Repudiation Transmission Protocol [17] defined by Wang et al. proposed a one-way hash function using a shared secret between each parent-child node pair to identify suspicious nodes which have considerably less overhead. Further, they have a checksum being created and transmitted on downward streams to identify the exact location of the adversary. However, computing the shared secret for all node pairs can have tremendous overhead in a dense network of mobile nodes. Also using a checksum in downstream links to verify the presence of adversary can not be successful when the nodes are mobile creating only temporary connections. Thus we modify the NRTP and propose a different verification scheme for identifying the adversary node.

The Non-Repudiation Protocol for a dense mobile network, instead of entrusting each pair of nodes, tries to ensure that no node can repudiate the authenticity of a packet it sent with the central controller. This is performed by a secret that every node shares only with the central unit. This should be part of the authentication of a node when it joins the network. So every node $X$ will have its secret $sec_X$ shared with the central controller. Now whenever a node has to send a generation of packets, it will create evidence $\Phi(e)$ on the block $e$ using the one-way hash function. This evidence can only be produced by the node $X$ and can be verified by the central controller. Thus every outgoing block will be of the format $\Phi(e)\|ID_A$. The evidence attached to each block does not make any sense to the receiving node. However, the receiving node can verify the integrity of the received block using the tags being attached to the packets and comparing them with the tags being shared by the central controller. If a genuine node detects a polluted packet, it has to report the polluted packets with the evidence associated with the packets along with the ID of the sender. The format of the report will be $\Phi(e)\|ID_X$. Then the central unit can verify the packets and the evidence associated with it. No node can deceive the controller by sending a modified report since it can not create proper evidence which will be the same as the evidence created by another node. If the controller gets a report which involves a block-evidence pair which doesn’t match with the reported ID of the adversary, then both the reporter and reported nodes will be considered as a suspicious node. Different attack scenarios are discussed in detail in the next section. It is also to be noted that other integrity schemes can also be used for detecting the pollution attack and the non-repudiation protocol can be used to identify the attacker. However, a trusted central controller is necessary to advertise the presence of a malicious node and [13] is used to detect polluted packets in our approach since it also uses a central controller in the detection scheme.

This proposed scheme requires only those genuine nodes detecting the pollution to send a report to the central unit making it more bandwidth-efficient compared to [19]. Further, the modified NRTP makes sure the number of secrets to be shared between the users is minimum (just one secret per user with the trusted central unit) compared to a large number of secrets (one for each parent-child pair) to the NRTP proposed in [17] in a dense environment. Thus the proposed scheme ensures that the malicious users are identified efficiently making the network coding enabled small cell environments secure.

IV. ANALYSIS OF THE PROPOSED SCHEME IN DIFFERENT ATTACK SCENARIOS

This section discusses different scenarios related to the locating scheme and how the non-repudiation protocol ensures precise identification of the adversary. Since the probability of any node creating a valid block-evidence pair matching with the ID of another node, there can be two different situations of reports reaching the central controller; either the controller can verify the report and identify the adversary or it can not verify the report and consider the report as a suspicious one.

A. The controller receives a verifiable report

This is the basic scenario of an adversary node injecting polluted packets and a benign node report it correctly. Fig. 3 represents this scenario. The non-repudiation protocol ensures that no node can replicate a piece of valid evidence created by another node, so if the controller can verify the block-evidence pair, then the report is valid. Any adversary who sends a polluted packet with the evidence it created with its secret key with the controller will be reported as soon as the pollution is identified. The report will constitute $\Phi(e)\|ID_A$, where $e$ is the block with corrupted packet and tags, $\Phi(e)$ is the evidence created on $e$ by the adversary node and $ID_A$ is the identity of adversary node. The controller can verify the polluted packets by checking the tags included in the packet and also verify the identity of the attacker by verifying the evidence associated with the reported block. The controller will add
the identity of the attacker, $ID_A$ to the malicious node list and announce the presence of adversary in the network. The controller may receive a verifiable report when the reporting node is an adversary when a genuine packet with genuine evidence is reported as a polluted packet. However, in this case, the controller can find out the reported packet is not polluted by verifying the tags and identify the reporting node as an adversary.

**B. The controller receives a non-verifiable report**

There can be two different reasons for the controller to receive a non-verifiable report. Either the reporting node is trying to fake the evidence from one of its neighbors or an intelligent adversary performing the pollution attack and sending the block with wrong evidence. In both cases, the reporting node will send a report $[\Phi(e)|e[ID_A]$ where evidence $\Phi(e)$ does not match with the block $e$ along with the ID of suspected adversary $ID_A$.

1) **Adversary node trying to deceive controller by sending a wrong report**: Fig. 4 shows a scenario when the adversary node does not pollute any packet but tries to disparage a genuine node by accusing it of sending a polluted packet. However, the adversary can not create valid evidence for any block which will entitle the authenticity of the block to another node since the secret key of each node is only known to that particular node and the controller. The controller can not verify the evidence with the reported block and ID and then both the reporting and reported nodes will be added to the list of suspicious nodes. If the same node repeatedly sends such fake reports, then the controller will identify the malicious activity and move it from the suspicious node list to the list of malicious nodes and the other node can be removed from the list of suspicious nodes. Further, if it is a genuine node and it genuinely received a corrupted packet, it will not accept any more packets from that sender. Thus any node repeatedly sending reports disparaging a particular node should be considered as a malicious node without waiting for further reports.

2) **An intelligent adversary injecting polluted packet with bogus evidence**: This is the scenario where an adversary node injecting polluted packets along with non-verifiable evidence. In this case, again the central unit will consider both reporting and reported node as suspicious and wait for further reports to decide the malicious node. Fig. 5 shows the scenario where malicious node A sends a corrupted packet to node B. Node B will drop the packet as soon as pollution is detected and also send the report $[\Phi(e)|e[ID_A]$ to the controller. Since the controller can not verify the block-evidence pair, it will add both node A and node B to the suspicious node list. Node B will not accept any further packet from node A since it already detected pollution. However, the adversary node may try to send corrupted packets to any other node will again be reported to the central controller by that node. Thus there will be multiple reports against node A and the central controller can identify the malicious node A and move it into the list of malicious nodes. Also, the other suspicious node can be removed from the list as well. However, in this case, a minimum number of reports to decide whether a node is malicious or not should be defined depending on the probabilities of colluding attackers. If the network is susceptible to multiple attackers, then the number of reports to identify a malicious user needs to be more than two. Otherwise, it will lead to a situation where multiple adversaries can collude to report a genuine node and isolate it in the network.
Identifying pollution attacks and preventing further occurrences of the attack is a foremost concern to ensure high throughput in a network coding enabled environment. While adapting network coding to a dense small cell environment identifying the adversaries and publicizing their presence to other nodes as soon as they perform malicious activities, can improve the performance of the network considerably. This paper presents a malicious user identification scheme for network coding enabled mobile small cells. However, it requires a trusted central unit to make the final decision regarding the adversary and inform the network regarding the presence of malicious users in the network. A non-repudiation protocol for a dense network of mobile nodes is presented to ensure that no adversary can disparage a benign node. This non-repudiation protocol is a standalone protocol which makes it adaptable to different integrity schemes to enhance the security of the network. Different attack scenarios are analyzed to verify the accuracy of the proposed scheme. The analysis shows that the malicious user identification scheme can successfully identify different types of attackers effectively in the small cell environment.

REFERENCES

[1] G. Bianchi, E. Biton, N. Blefari-Melazzi, I. Borges, L. Chiaraviglio, P. Cruz Ramos, P. Eardley, F. Fontes, M. J. McGrath, L. Natarianni et al., “Superfluidity: a flexible functional architecture for 5G networks,” Transactions on Emerging Telecommunications Technologies, vol. 27, no. 9, pp. 1178–1186, 2016.
[2] L. Bolivar, C. Tselios, D. Mellado, and G. Tsolis, “On the deployment of an open-source, 5G-aware evaluation testbed,” in IEEE International Conference on Mobile Cloud Computing, Services and Engineering (Mobile Cloud), March 2018, pp. 1–6.
[3] P. Georgakopoulos, T. Akhtar, I. Politis, C. Tselios, E. Markakis, and S. Kotsopoulos, “Coordination multipoint enabled small cells for coalition-game-based radio resource management,” IEEE Network, vol. 33, no. 4, pp. 63–69, July 2019.
[4] J. Rodriguez, A. Radwan, C. Barbosa, F. H. P. Fitezk, R. A. Abd-Alhameed, J. M. Noras, S. M. R. Jones, I. Politis, P. Galiotos, G. Schulte, A. Rayit, M. Sousa, R. Alheiro, X. Gelabert, and G. P. Koudouridis, “SECRET - Secure network coding for reduced energy next generation mobile small cells: A European Training Network in wireless communications and networking for 5G,” in 2017 Internet Technologies and Applications (ITA), Sept 2017, pp. 329–333.
[5] R. Ahlswede, N. Cai, S.-Y. Li, and R. W. Yeung, “Network information flow,” IEEE Transactions on information theory, vol. 46, no. 4, pp. 1204–1216, 2000.
[6] N. Cai and R. W. Yeung, “Secure network coding on a wiretap network,” IEEE Transactions on Information Theory, vol. 57, no. 1, pp. 424–435, 2010.
[7] V. N. Talouki, R. Bassoli, D. E. Lucani, J. Rodriguez, F. H. Fitzek, H. Marques, and R. Tafazolli, “Security concerns and countermeasures in network coding based communication systems: A survey,” Computer Networks, vol. 83, pp. 422–445, 2015.
[8] R. Parsamehr, G. Mantas, A. Radwan, J. Rodriguez, and J.-F. Martínez, “Security threats in network coding-enabled mobile small cells,” in International Conference on Broadband Communications, Networks and Systems. Springer, 2018, pp. 337–346.
[9] A. Antonopoulos and C. Verikoukis, “Cops: Cooperative statistical misbehavior mitigation in network-coding-aided wireless networks,” IEEE Transactions on Industrial Informatics, vol. 14, no. 4, pp. 1436–1446, April 2018.
[10] V. Adat, C. Tselios, and I. Politis, “On security against pollution attacks in network coding enabled 5g networks,” IEEE Access, pp. 1–22, 2020.
[11] P. Zhang, Y. Jiang, C. Lin, H. Yao, A. Wasef, and X. Shen, “Padding for orthogonality: Efficient subspace authentication for network coding,” in 2011 Proceedings IEEE INFOCOM, April 2011, pp. 1026–1034.
[12] A. Esfahani, G. Mantas, J. Rodriguez, and J. C. Neves, “An efficient homomorphic mac-based scheme against data and tag pollution attacks in network coding-enabled wireless networks,” International Journal of Information Security, vol. 16, no. 6, pp. 627–639, 2017.
[13] V. Adat, I. Politis, C. Tselios, and S. Kotsopoulos, “Secure network coding for sdn-based mobile small cells,” in Broadband Communications, Networks, and Systems, V. Sucasas, G. Mantas, and S. Althubabat, Eds. Cham: Springer International Publishing, 2019, pp. 347–356.
[14] V. Adat, I. Politis, C. Tselios, P. Galiotos, and S. Kotsopoulos, “On Blockchain Enhanced Secure Network Coding for 5G Deployments,” in 2018 IEEE Global Communications Conference (GLOBECOM), Dec 2018, pp. 1–7.
[15] R. Parsamehr, A. Esfahani, G. Mantas, A. Radwan, S. Mumtaz, J. Rodriguez, and J.-F. Martínez-Ortega, “A novel intrusion detection and prevention scheme for network coding-enabled mobile small cells,” IEEE Transactions on Computational Social Systems, vol. 6, no. 6, pp. 1467–1477, 2019.
[16] M. J. Siavoshani, C. Fragouli, and S. Diggavi, “On locating byzantine attackers,” in 2008 Fourth workshop on Network Coding, Theory and Applications. IEEE, 2008, pp. 1–6.
[17] Q. Wang, L. Yu, K. Nahrstedt, and H. Khurana, “Identifying malicious nodes in network-coding-based peer-to-peer streaming networks,” Tech. Rep., 2009.
[18] A. Le and A. Markopoulou, “Cooperative defense against pollution attacks in network coding using spaceman,” IEEE Journal on Selected Areas in Communications, vol. 30, no. 2, pp. 442–449, 2012.
[19] R. Parsamehr, A. Esfahani, G. Mantas, J. Rodriguez, and J.-F. Martínez-Ortega, “A location-aware idps scheme for network coding-enabled mobile small cells,” in 2019 IEEE 2nd 5G World Forum (5GWF). IEEE, pp. 91–96.
[20] M. Kim, M. Médard, and J. Barroso, “Algebraic watchdog: mitigating misbehavior in wireless network coding,” IEEE Journal on Selected Areas in Communications, vol. 29, no. 10, pp. 1916–1925, 2011.
[21] A. Antonopoulos and C. Verikoukis, “Misbehavior detection in the internet of things: A network-coding-aware statistical approach,” in 2016 IEEE 14th International Conference on Industrial Informatics (INDIN). IEEE, 2016, pp. 1024–1027.
[22] F. Adelantado and C. Verikoukis, “Detection of malicious users in cognitive radio ad hoc networks: A non-parametric statistical approach,” Ad Hoc Networks, vol. 11, no. 8, pp. 2367–2380, 2013.
[23] S. Agrawal and D. Boneh, “Homomorphic MACs: MAC-Based Integrity for Network Coding,” in Applied Cryptography and Network Security, M. Abdalla, D. Pointcheval, P.-A. Fouque, and D. Vergnaud, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 292–305.
[24] V. Adat, I. Politis, C. Tselios, and S. Kotsopoulos, “Blockchain enhanced secret small cells for the 5g environment,” in 2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Sep. 2019, pp. 1–6.
[25] C. Tselios, I. Politis, and S. Kotsopoulos, “Enhancing SDN security for IoT-related deployments through blockchain,” in 2017 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Nov 2017, pp. 303–308.
[26] X. Liang and X. Qiu, “A software defined security architecture for sdn-based 5g network,” in 2016 IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC). IEEE, 2016, pp. 17–21.