Communication Security in Internet of Thing: Preventive Measure and Avoid DDoS Attack Over IoT Network

Congyingzi Zhang
Computer Science Department
Bowling Green State University
Bowling Green, OH, US
czhang@bgsu.edu

Robert Green
Computer Science Department
Bowling Green State University
Bowling Green, OH, US
greenr@bgsu.edu

ABSTRACT

The idea of Internet of Things (IoT) is implanting networked heterogeneous detectors into our daily life. It opens extra channels for information submission and remote control to our physical world. A significant feature of an IoT network is that it collects data from network edges. Moreover, human involvement for network and devices maintenance is greatly reduced, which suggests an IoT network need to be highly self-managed and self-secured. For the reason that the use of IoT is growing in many important fields, the security issues of IoT need to be properly addressed. Among all, Distributed Denial of Service (DDoS) is one of the most notorious attacking behaviors over network which interrupt and block genuine user requests by flooding the host server with huge number of requests using a group of zombie computers via geographically distributed internet connections. DDoS disrupts service by creating network congestion and disabling normal functions of network components, which is even more disruptive for IoT. In this paper, a lightweight defensive algorithm for DDoS attack over IoT network environment is proposed and tested against several scenarios to dissect the interactive communication among different types of network nodes.

Author Keywords

IoT, DDoS attack defensive mechanism, network communication simulation.

ACM Classification Keywords

I.6.4 MODEL VALIDATION AND ANALYSIS

INTRODUCTION

The Internet of Things (IoT) is networked interconnections among various objects which interact and communicate with each other in real-time [23]. Those objects are mostly monitoring and sensory devices which are implanted into target physical environments intended for a seamless presentation of the physical world to the digital world in the form of data flow. From bottom up, IoT network consists of uniquely addressable data communicating and collecting objects, data transmission network, computing platform, and customized user applications [22].

SpringSim 2015, April 12 - 15, 2015, Alexandria, VA, USA

Thanks to the declining price and maturity in technology of data collection sensors, wireless mobile communication, embedded system, and cloud computing, almost all electronic devices could be included into the IoT network. As a smart architecture facilitating information exchange, IoT is used to supports global services and goods supply chain networks in many area such as industry, logistics, academe, medical system, military, government and so on. It has been considered as the third wave of information technology after Internet and mobile communication network [10, 19, 25].

The rapid growth of IoT application in multiple areas also brings up the research challenges tightly related to the nature of IoT technology. Compared with the internet which majorly connects personal computers and mobile communication device, large number of heterogeneous devices are included in the IoT network. The number of IoT sensory devices is growing rapidly over the world, which exceeded the number of human beings on this planet since 2008. The tremendous amount of data collected by those devices overwhelms the system with great challenges in information modeling and reasoning for further understanding of the data [17]. To seamlessly report the physical world to the digital world, those distributed heterogeneous devices need to communicate with each other at real time, which requires solution to ensure efficient and reliable end-to-end communication over the IoT network [6]. In some area of application, the collected data is sensitive, which also brings the security concerns of client privacy, data protection, authentication, access control, and so on [1, 14, 22].

Distributed Denial of Service (DDoS) is a type of network attack featuring disrupting service for legitimate requests, which is often done by flooding the targeted host server with bad requests to temporarily reduce legitimate users' bandwidth [9]. In the attack on February 9, 2000, it caused huge financial loss for big companies who largely relies on electronic business such as Amazon, Yahoo, and eBay. Later in 2006, over 1500 IP addresses were attacked at a high rate of 10 GB/s.

Many researches have proposed DDoS defense technologies over the internet. Others have done work classifying types of DDoS attacks and defense mechanisms [2, 11, 21]. However, not much has been done for addressing and solving DDoS problem specifically over IoT network even though DDoS poses more threats to IoT network because of its open nature. To specify, the communication between two devices over IoT network are machine-to-machine instead of human-to-machine as the case of the Internet. Less human involvement requires a more responsive system for error detection and correction. Moreover, communication is cascaded end-to-end in the IoT network, which means congested network or malfunctioning devices might impair a subset of the network. As a result, not only the two devices at communication are delayed in such circumstance. The information security of IoT network is a big concern in the near future for the reason that it collaborates everyday objects and thus enables further impact on our everyday life.

The literature review section covers the nature of IoT and discusses the reason why IoT network is potentially vulnerable to DDoS attack. Then, the related technologies of DDoS attack are explained. The preventive and defensive approaches for DDoS attack over the Internet are discussed as the standpoint for the conducted research. Furthermore, two research cases proposing a DDoS attack defensive mechanism are reviewed as to see the designing trend and their limitation. In the methodology section, a defending algorithm is proposed for an IoT end network. The related simulation technology and tools are introduced to demonstrate their fitness for the study. In the fourth section, three experiments were conducted to demonstrate the effectiveness of the algorithm and to show the interactive communication in an IoT end network. In the last section, some potential extensions from this preliminary work were pointed out as future study.

LITERATURE REVIEW

Security Concerns Rising From the Nature of IOT

IOT merges the physical world and the digital world by handling over control of real-life objects to the massive and ubiquitous autonomous network, which weaken the physical boundary between the two worlds and will inevitably cause great changes in our life in the near future. Since the application of IoT technique would be blooming in crucial areas of human activities including economy, e-health, and industrial supply chain, we should consider many of its natures listed below raise high demand for addressing the according security concerns.

- IoT network is not integrated with traffic policing mechanism.
- IoT network is designed to be open to new devices.
- IoT connected heterogeneous objects diverse largely in power supply and computing capability.
- IoT enables automated communication among interconnected nodes without human interaction.
- IoT could be powered by cloud and grid computing and scale up fast.

How does it possible for DDoS pose threats to IoT network? This question could be considered from IoT's inherited features from the Internet as a network of connected devices and also from its distinct differences from the Internet.

First, since the Internet is not designed to police intermediate traffic. Its end-to-end design paradigm make the intermediate network simple and optimized to ensure the fastest packet forwarding service while leave the complexity of packet processing to the hosts on the two ends of the communication. When proper detecting and preventive mechanism are missing on the receiver, the system becomes venerable to malicious packets streamed from the sender. In the IoT network, end devices are usually not equipped with high computational resources for implementing complex security algorithm and usually limited in power supply, which makes them not intelligent enough to detect and avoid network attack.

Second, the service available on one IoT network component is limited, which means only certain number of requests could be served at one time. When malicious packets taking a large portion of the total requests, chances that legitimate requests being temporarily blocked becomes larger.

Third, for the reason that IoT devices are connected via end network with relatively low bandwidth capacity compared with the intermediate network, it becomes easier to flood a target end network by dumping huge amount of packets from the faster intermediate network [11].

Compared with the public Internet, IoT network is exclusively designed to be opened for many types of devices. Its loose control over the connected simple devices increases the risk of including malicious devices into the network. Moreover, for IoT, the work flow is highly dependent on the communication between the chained devices over the network. Single point failure would lead to cascade effect over an area of end network. For example, once DDoS attack brings down the serving device on a IoT network, the other IoT devices whose functions rely on the this blocked device will be also blocked from serving their client devices, which causes impairment of a local network [12].

DDoS Attack

To start with, Denial of Service (DoS) attack is defined as denying and disrupted legitimate access to the service or resources on target server. Even worse, Distributed Denial of Service (DDoS) attack typically engages more computers and internet connections to such attacking behavior to engender real threats that seriously blocks or suspends other users' accesses to the host server, which leads to huge business loss and client inconvenience.

The targeted service could be disrupted by the attack crashing the host server with some carefully designed packets whose content causes certain operating system to freeze or reboot. Other than that, the malicious packets occupy all the resources on the host server with massive volumes of bad requests, which is also called bandwidth attack in related researches. Prevented by patching the host operating system against the identified attack, the first form of attack could be stopped at some point. However, the massive volume-based attack is quite hard to defense.

A volume-based attack is usually initiated with installing "bot" onto vulnerable systems. Bot technology was used in industry for automating process. In such way, hackers can easily populate their attacking army with zero cost. Zombies' or bots' behavior could be manipulated through secured channels in order to launch further attacks to the targeted IP or a local network.

To specify the difficulties in finding solutions, first, the aggregated large traffic volume exceeds throughput of many network security devices and capacity of corporate internet link. Second, controlled zombie systems are geographically distributed, which is hard to locate source IP addresses. Third, when separately examined, single attack from one source is not powerful enough to be discriminate from a legitimate request, which makes it look similar to a flash crowd created by legitimate requests at a website peak time [8, 16].

Current DDoS Defense Strategies

Many DDoS defense strategies were proposed, implemented, and tested to be effective against DDoS attack over the Internet. In this section, the most common defense designs are to be reviewed for potential solution to the DDoS attack over an IoT network. Defensive strategies could be categorized by the sequence of the attacking event.

Before attack, preventive approaches have been added to eliminate the attack traffic. Attack detecting and identifying mechanism is implemented to monitor the coming traffic. Three parameters are often examined in this link including resource IP address, traffic increasing degree, and similarity among the traffic. However, traffic degree monitoring sometimes could cause false alarm because sudden traffic increase can also be the result of a flash crowd which consists of legitimate requests [8]. Using the other two parameters, one would more confidently distinguish between malicious traffic and flash crowd. The similarity among the traffic of a DDoS attack is usually higher than that of flash crowd for two reasons. First, attacking traffic is usually generated by bots from one botnet, which indicates high similarity in source IP. Second, in the cases that the attacking IP addresses are distributed from slave machines all over the world, because all bots execute same or similar source code, the similarity in packet content could also be higher than those from a flash crowd [20]. Some counter actions are taken to limit malicious traffic. The simplest one is filtering out the packets from identified spoofed IP addresses and dropping them using unicast reverse path forwarding at routers. Attacks from valid IP addresses cannot be avoided in such. Firewall is a common options which be used to stop the traffic upon identified attackers' IP. There are also indirect methods to solve the DDoS problem, for example, using congestion control to cut down the attacking traffic flow and increasing the resource production at host server. However, this method is not quite effective when the target flow is small and similar to legitimate request and attacking machine is highly distributed. Some other method such as reconstructing the attacking path to limit amount of packets going through, however this method needs large storage and computing resources for path mapping function. Similarly, mining old attacker data and using their features for packet sampling is also suggested in some researches. Even more, by tracking back the attacking root, server can actively block the attack traffic, which proved to be effective defensive response mechanism [21].

Conventional DDoS preventive measures and defenses too heavily rely on power supply, computing resources, and longtime processing. Considering the characteristics of IoT environment, all such preconditions should be avoided in the design of IoT defense system. One needs to keep it in mind that IoT hardware components are highly heterogeneous and very limited in power supply and computing capability when comparing to traditional nodes over the internet such as personal computers, smart phone, and tablets. Other than that, maintaining real-time communication in IoT network is fairly important, longtime processing will cause delay and target miss during the task of identifying malicious traffic.

Considering all the device and environment constraints of IoT network, implementing light weight defending mechanism for node devices is the first key for the design. Additionally, distributing defending mechanism across the multi-layer architecture of IoT is also applicable as the second key to the solution. Third, adding extra security devices in a small subnet as a checking center is also feasible. Such device would be responsible for examining packets, keeping records of old attacking information, and tracking back the root of attacks to proactively reject threat in the future. Since the security mechanism relies on a small group of nodes whose computing resources are separated from the general IoT data collecting nodes, it would be cost-efficient to enable such mechanism on a small percentage of hardware instead of all devices over the IoT network.

Constructing Preventive Systems against DDoS Attack in IoT Network

From the many researched DDoS attacks and defense mechanisms, similar approaches could also be applied to the IoT domain. Since the implementation of IoT has just comes into our sight, currently, the amount of researches specifically for solving DDoS attack problem in the IoT network is still quite limited.

Learning Automata (LA) is proposed as a strategy to prevent DDoS attack by intelligently determining the packet sampling rate from the environment [13]. The adaptive learning component was proposed in their previous work for the solution to DDoS attack in wireless mesh networks (WMN) [21]. The defensive mechanism has been designed cross the layers in the SOA architecture of IoT, which enables effective prevention schemes on all layer of the network model. Contained by their proposed defensive component, the LA mechanism takes set of sampling rates from the random environment as the input and responses a best suited action according to the given actions as output. In the detecting phase, the DDoS prevention component in each device monitors the number of requests that each layer receives. A preset maximum servicing capacity for each layer will be used as a threshold value for issuing a DDoS alert (DALERT) among the neighboring nodes once it is exceeded. Once the devices are notified of a potential attacking, they start sampling the IP addresses, among which the host sending most requests would be identified as the attacking device. Then, the attacker's information is dispersed in the Attacker Information Packet (AIP) among all the nodes. They start sampling from the coming traffic based on the content from the AIP and drop the malicious packets. To minimize the latency and energy consumption during the sampling phase, the LA component is used to establish the optimum sampling rate. The process will continue until the coming traffic flow drop below the preset threshold. The communication process among the nodes relies on the functions performed by the peering SOA layers.

Another approach by a different group is backing up the sink node with a new sensor node. The newly added node is referred as a redundant channel to hold a portion of the responsibility of the sink node. In such simple way, the chance that the sink node down is reduced and also prolong the life of the whole sub-network. Moreover, backing up only the sink node is considering being cost-efficient since usually there is just one of it in a sub-network [24].

The ideas behind these two research works cover part of designing rules of thumb for a defending system in IoT network including adding resources, filtering by ID address, proactive block, and peer layering communication. However, more structured, efficient, and acute approach needs to be added into this scope.

PROPOSED METHODOLOGY

In this section, an IoT DDoS defense algorithm for an IoT end network is proposed for preventive measuring and avoiding DDoS attack. The design of the defense algorithm is guided by several research motivations including how to make working nodes which are mostly data collecting nodes in an IoT network intelligently detect and avoid DoS-like attack and remain functioning? How to make such "intelligence" lightweight and inexpensive? How to make a local IoT end network sensitive to certain attacker for a long time after the first detection of its malicious behavior? Following these questions, major types of network elements and their behavior are designed to meet the above demands in a modeled IoT end sub-network.

Modeling IoT End Network

In a typical IoT end network involved with DDoS attack scenario, four different types of nodes including working node, monitoring node, legitimate user node, and the attacker node are constructed to be present in a simulation environment.

Working node

A working node is the device collecting information and executing simple tasks in an IoT network. In one hand, they are characterized by limitation in memory, storage, and power supply. On the other hand, they are usually of the most number in a functioning IoT local network. So, it is necessary to ensure each of them is equipped with certain attack detecting mechanism which also has to be lightweight and inexpensive to implement.

A major behavior of a proposed working node is serving request and defending itself from attacks. During the request serving stage, the service of a node should be blocked by a previously validated request and not be available to server other request when it is busy. The node will notify the requesting entity whether its request has been served. Additionally, the node will not enable queuing function for the rejected requests, which corresponds to the simplicity of the device. As a result, the competition over a limited service is always won by the user who requests most frequently, which in the case of a typical DDoS attack, this role is played by the attacker.

To defend itself from DDoS attack, a node should be able to distinguish malicious requests from legitimate ones. As for the reason that DDoS requests usually contain the same meaningless content, the proposed defending algorithm determines a sender is malicious according to the consistency of the content in the packets it sends. If a sender repeatedly send request with same content, it will be flagged as an attacker. Upon the reception of request from this exact address, the working node will refute its request and remain bandwidth for service providing.

To implement the above features, a list of records of served request is maintained. Each record contains the information including sender address, the most recent request content, and a flag to mark whether a sender has been determined as an attacker. Upon the detection of repeated request content or a true flag for being malicious, service will not be provided. Furthermore, considering the limitation of the working node devices, the length of record list is maintained short.

Legitimate User node

A legitimate user is distinguished from an attacker by sending request for service with a lower frequency and reasonable content. To implement this feature, a legitimate user node is designed to unicast its request with a frequency of 10 seconds after initiation to one of the working node in an IoT end network. It will wait and print the response from the working node.

Attacking node

An attacker's behavior could be differed from that of a legitimate user by its high frequency of sending requests and the same content in those sent packets. To implement this feature in a simulation, an attacking node is designed to always send same request with certain higher frequency compare to that of legitimate user node. To detail, a timer to be expired in random seconds between 1 to 3 second is set, after the initiation of the attacking node, whenever the timer is expired, it broadcast and send same junk packets to the nearby working nodes to ask for service. Hypothetically, in this proposed sub-network, for each attacking node, it only has one chance to be served and block the serving node. Once it has been detected as an attacker, its packets are to be rejected and dropped. So, with the implementation of the defending mechanism in the working nodes and monitoring node, the effect of a wave of DDoS attack will be relived within one service cycle.

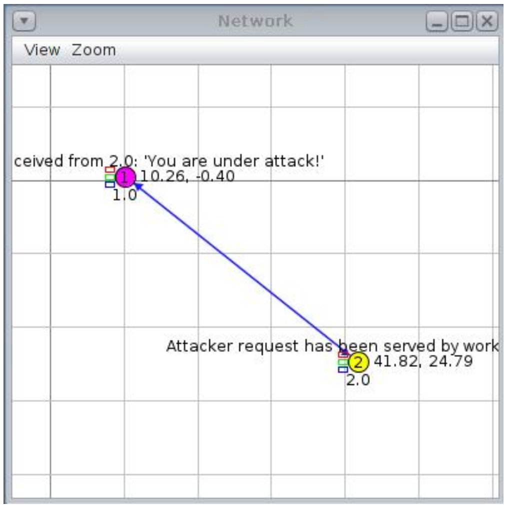
Simulation Platform

Contiki OS

Contiki is an open source operating system for sensor network developed at the Swedish Institute of Computer Science since 2004. Among the available network simulation tools, Contiki operating system holds powerful simulating and communication methodology for the IoT microcontrollers, named 'motes' and mentioned as 'nodes' in this study. Contiki runs as a virtual machine over an operating system handled by VMware player. So, it is highly portable and efficient for code backing up [4]. To keep the memory overhead down in the resource limited devices, event-driven programming is applied in the operating system. Plus, to ensure the event-driven program easy to write and debug, a thread-like programming style, called protothreads, which helps to reduce the lines of code with only two bytes of memory overhead per protothread [5, 7, 18].

COOJA

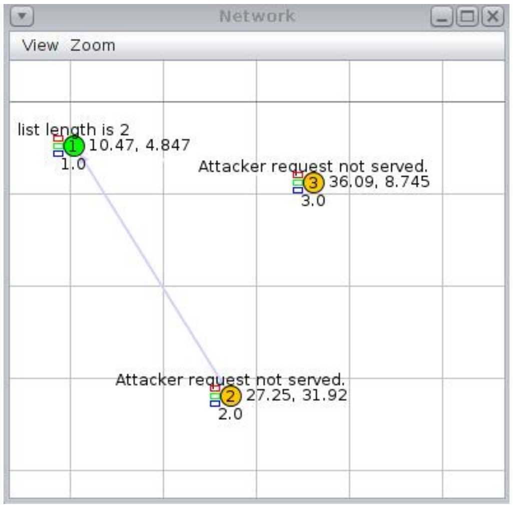
COOJA is a Contiki network simulator. It stands out from other emulators by allowing cross-level simulation in the WSN. It enables simultaneous simulation from low level regarding that for sensor node hardware to high level regarding that for node behavior. With this simulation environment, developers can see their applications run in large-scale networks and also tune the emulated hardware in extreme detail [15].



ceived from 2.0: 'You are under attack!'

Attacker request has been served by work

Figure 1. An attacking node (ID=2.0) and a working node (ID=1.0) interacting without defending algorithm.



list length is 2

Attacker request not served.

Attacker request not served.

Figure 2. Two attacking nodes (ID=2.0, 3.0) and a working node (ID=1.0) interacting with defending algorithm.

Rime stack

As part of Contiki's system core, rime is a lightweight layered communication stack for sensor networks. It was tailored to simplify the implementation of traditional layered communication protocol in sensor network and encourage code reuse. It fully supports operations like broadcasting, unicasting, network flooding, and address-free multi-hop semi-reliable scalable data collection, which makes it a great fundament for building an out-of-tree implantation for the proposed DDoS defending algorithm [3].

EXPERIMENT RESULTS

To test the effectiveness of the proposed algorithm, several IoT network scenarios were constructed with the four types of proposed nodes. To demonstrate and clarifying the effect of the proposed algorithm, interactions between each pair of two different types of nodes are individually tested with and without the defending algorithm. Then, the combined communication of all node types is examined.

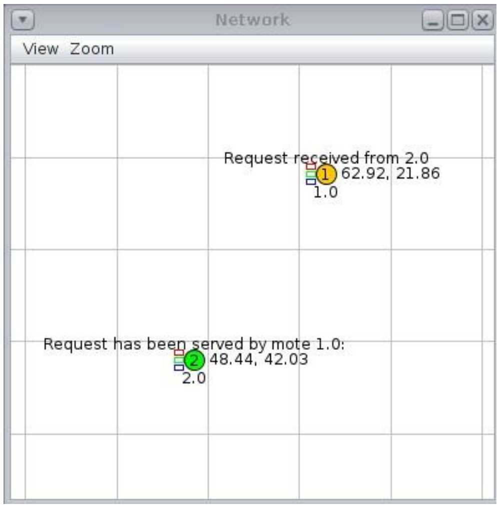

Network
View Zoom
Request received from 2.0
1
62.92, 21.86
1.0
Request has been served by mote 1.0:
2
48.44, 42.03
2.0

Figure 3. A working node (ID=1.0) and a legitimate user mote (ID=2.0) interacting with no attacker.

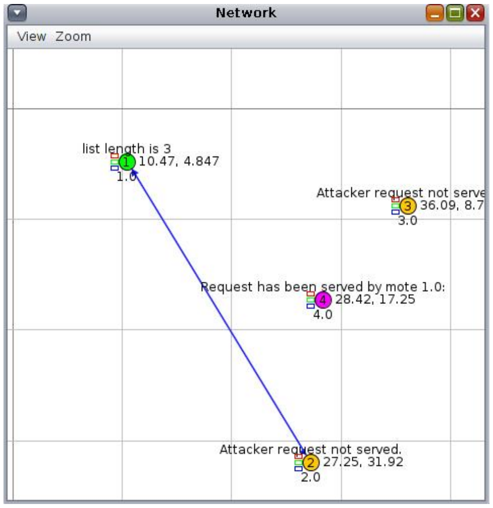


Figure 4. A working node (ID=1.0), two attacking node (ID=2.0, ID=3.0), and a legitimate user node (ID=4.0) interacting with defending algorithm applied.

Interaction between attacking node and working node

In this scenario, one attacker node and one working node are placed in an IoT local network. The attacker node requests for service every 1 to 2 seconds and will not stop until the end of stimulation. The purpose of this scenario is to examine whether the working node is able to distinguish and reject the malicious service request after it being blocked for the first time. The first set of results (Fig.1, Table 1) shows the situation happened without the defending algorithm. However, with the defending algorithm applied (Fig.2, Table 2), the working node is able to distinguish the malicious peers and reject their requests after serving them for the first time. The records of malicious nodes are archived in the record list, which is indicated by the growing length in the record list.

Time (s)	Mote output	
	Mote ID	Message
0.517	2.0	Starting 'Attacker request'
0.663	1.0	Starting 'Serve request'
1.786	1.0	Request received from 2.0: 'You are under attack!'
1.888	2.0	Attacker request has been served by worker 1.0
3.411	1.0	Request received from 2.0: 'You are under attack!'
3.515	2.0	Attacker request has been served by worker 1.0

Table 1. Interactive communication flow between an attacking node (ID=2.0) and a working node (ID=1.0) without defending algorithm

Time (s)	Mote output	
	Mote ID	Message
0.517	2.0	Starting 'Attacker request'
0.663	1.0	Starting 'Serve request'
1.180	3.0	Starting 'Attacker request'
1.785	1.0	Request received from 2.0: 'You are under attack!' Request served, ID 2.0 has been added to the list
1.889	2.0	Attacker request has been served by worker 1.0
3.283	1.0	Request received from 3.0: 'You are under attack!' Request served, ID 3.0 has been added to the list
3.428	3.0	Attacker request has been served by worker 1.0
3.660	1.0	Request received from 2.0: 'You are under attack!' Found record. Request rejected.
3.762	2.0	Attacker request not served.
5.658	1.0	Request received from 3.0: 'You are under attack!' Found record. Request rejected.

Table 2. Interactive communication flow among a working node (ID=1.0) and two attacking node (ID=2.0, 3.0) with defending algorithm

Interaction between legitimate user node and working node

In this scenario, one legitimate user node and one working node are placed in an IoT local network (Fig.3, Table 3). The user node starts asking for service after the simulation begins for 10 seconds. The working node is expected to service the request and output the job status. If the request is served, the working node returns the "Served" status with an enum in a unicast message to the user node. Then, the user node will print the message about its request has been served by the node ID number of the responder to indicate the completion. Otherwise, it will send a “Rejected” message back to the user node to notify it being unable to fulfill the request.

Time (s)	Mote output	
	Mote ID	Message
0.517	2.0	Starting ‘Legitimate user request’
0.663	1.0	Starting ‘Serve request’
10.534	1.0	Request received from 2.0
10.639	2.0	Legitimate user request has been served by worker 1.0

Table 3. Interactive communication flow between a working node (ID=1.0) and a legitimate user node (ID=2.0) not under attack

Time (s)	Mote output	
	Mote ID	Message
9.660	1.0	Request received from 2.0: ‘You are under attack!’ Found record. Request rejected.
9.763	2.0	Attacker request not served.
9.785	1.0	Request received from 3.0: ‘You are under attack!’ Found record. Request rejected.
9.926	3.0	Attacker request not served.
10.659	1.0	Request received from 4.0: ‘User requesting’
10.750	4.0	Legitimate user request has been served by worker 1.0

Table 4. Interactive communication flow between A working node (ID=1.0), two attacking node (ID=2.0, ID=3.0), and a legitimate user node (ID=4.0) with defending algorithm applied

Interaction between legitimate user node and working node at the presence of multiple attacking nodes

In this scenario, the effectiveness of DDoS attack over an IoT network is demonstrated by adding multiple attackers, one legitimate user, and a working node in to an IoT end network (Fig.4, Table 4). The major purpose of this experiment is to show how differently the working node treats legitimate requests and malicious requests. The result indicates after being detected and added to the record list after the first serving cycle, the attacking nodes were not served afterwards. Instead, when the legitimate user node requests around 10th second of the simulation, it was served and added to the list, which indicated as growth in list length.

CONCLUSION & FUTURE WORK

According to the results, the proposed defending algorithm could effectively help the working nodes in an IoT network to distinguish malicious requests from legitimate ones and process them differently. For future work, an additional type of node could be used for handling the “running out of list space problem”. A monitoring node could be specifically designed for the extra demand in storage space. Moreover, it could also join policing local traffic and quick responding to an old archived attacker in an IoT end network. Typically, one monitoring node is used with a group of working nodes in a local network. Thus, it is allowed to have higher computing power and power supply to compensate the limited capability in the working nodes. Ideally, old records of attacker will be sent to the monitoring node to be archived. And later, upon the detection of archived attacker activities in the local network, the monitoring node should be able to warning the working nodes about the situation. In such way, most computing budget could be shifted onto one monitoring node while decision execution power is evenly distributed among the entire network to increase the defense sensitivity and lower the total cost on extra hardware.

REFERENCE

1. Aggarwal, Charu C., Naveen Ashish, and Amit Sheth. *The Internet of Things: A Survey from the Data-Centric Perspective, Managing and Mining Sensor Data*, 2013.
2. Douligeris, Christos, and Aikaterini Mitrokotsa. “DDoS Attacks and Defense Mechanisms: Classification and State-of-the-Art.” *Computer Networks* 44, no. 5 (April 5, 2004): 643–66. doi:10.1016/j.comnet.2003.10.003.
3. Dunkels, Adam. “Rime - a Lightweight Layered Communication Stack for Sensor Networks.” Delft, The Netherlands, 2007. http://www.sics.se/~adam/dunkels07rime.pdf.
4. Dunkels, Adam, Oliver Schmidt, Thiemo Voigt, and Muneeb Ali. “Protothreads: Simplifying Event-Driven Programming of Memory-Constrained Embedded Systems.” In *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, 29–42. SenSys ’06. New York, NY, USA: ACM, 2006. doi:10.1145/1182807.1182811.
5. Dunkels, A., B. Gronvall, and T. Voigt. “Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors.” In *29th Annual IEEE International Conference on Local Computer Networks, 2004*, 455–62, 2004. doi:10.1109/LCN.2004.38.
6. Han, Chong, Josep Miquel Jornet, Etimad Fadel, and Ian F. Akyildiz. “A Cross-Layer Communication Module for the Internet of Things.” *Computer Networks* 57, no. 3 (February 26, 2013): 622–33. doi:10.1016/j.comnet.2012.10.003.
7. Heddeghem, Ward Van. “Cross-Layer Link Estimation For Contiki-Based Wireless Sensor Networks.” Vrije Universiteit Brussel, 2009.
8. Jung, Jaeyeon, Balachander Krishnamurthy, and Michael Rabinovich. “Flash Crowds and Denial of Service Attacks: Characterization and Implications for CDNs and Web Sites.” In *Proceedings of the 11th*

International Conference on World Wide Web, 293–304. WWW ’02. New York, NY, USA: ACM, 2002. doi:10.1145/511446.511485.

9. Lau, F., S.H. Rubin, M.H. Smith, and L. Trajkovic. “Distributed Denial of Service Attacks.” In *2000 IEEE International Conference on Systems, Man, and Cybernetics*, 3:2275–80 vol.3, 2000. doi:10.1109/ICSMC.2000.886455.
10. Liu, Yuxi, and Guohui Zhou. “Key Technologies and Applications of Internet of Things.” In *2012 Fifth International Conference on Intelligent Computation Technology and Automation (ICICTA)*, 197–200, 2012. doi:10.1109/ICICTA.2012.56.
11. Mirkovic, Jelena, and Peter Reiher. “A Taxonomy of DDoS Attack and DDoS Defense Mechanisms.” *SIGCOMM Comput. Commun. Rev.* 34, no. 2 (April 2004): 39–53. doi:10.1145/997150.997156.
12. Misra, Sudip, P. Venkata Krishna, Harshit Agarwal, Antriksh Saxena, and Mohammad S. Obaidat. “A Learning Automata Based Solution for Preventing Distributed Denial of Service in Internet of Things.” In *International Conferences on Internet of Things, and Cyber, Physical and Social Computing*, 114–22. IEEE, 2011.
13. Misra, Sudip, P. Venkata Krishna, Kiran Isaac Abraham, Navin Sasikumar, and S. Fredun. “An Adaptive Learning Routing Protocol for the Prevention of Distributed Denial of Service Attacks in Wireless Mesh Networks.” *Computers & Mathematics with Applications* 60, no. 2 (July 2010): 294–306. doi:10.1016/j.camwa.2009.12.035.
14. Oleshchuk, V. “Internet of Things and Privacy Preserving Technologies.” In *1st International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace Electronic Systems Technology, 2009. Wireless VITAE 2009*, 336–40, 2009. doi:10.1109/WIRELESSVITAE.2009.5172470.
15. Osterlind, F., A. Dunkels, J. Eriksson, N. Finne, and T. Voigt. “Cross-Level Sensor Network Simulation with COOJA.” In *Proceedings 2006 31st IEEE Conference on Local Computer Networks*, 641–48, 2006. doi:10.1109/LCN.2006.322172.
16. Peng, Tao, Christopher Leckie, and Kotagiri Ramamohanarao. “Survey of Network-Based Defense Mechanisms Countering the DoS and DDoS Problems.” *ACM Comput. Surv.* 39, no. 1 (April 2007). doi:10.1145/1216370.1216373.
17. Perera, C., A. Zaslavsky, P. Christen, and D. Georgakopoulos. “Context Aware Computing for The Internet of Things: A Survey.” *IEEE Communications Surveys Tutorials* 16, no. 1 (First 2014): 414–54. doi:10.1109/SURV.2013.042313.00197.
18. Science, Swedish Institute of Computer, and Oct 2008. “Contiki Programming Course: Hands-On Session Notes.” *TechRepublic*. Accessed May 2, 2014. http://www.techrepublic.com/resource-library/whitepapers/contiki-programming-course-hands-on-session-notes/.
19. Seitz, Ludwig, Goran Selander, and Christian Gehrmann. “Authorization Framework for the Internet-of-Things.” In *2013 IEEE 14th International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM)*, 0:1–6. Los Alamitos, CA, USA: IEEE Computer Society, 2013. doi:10.1109/WoWMoM.2013.6583465.
20. Tao, Yuan, and Shui Yu. “DDoS Attack Detection at Local Area Networks Using Information Theoretical Metrics.” In *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, 0:233–40. Los Alamitos, CA, USA: IEEE Computer Society, 2013. doi:10.1109/TrustCom.2013.32.
21. Tariq, Usman, ManPyo Hong, and Kyung-suk Lhee. “A Comprehensive Categorization of DDoS Attack and DDoS Defense Techniques.” *Advanced Data Mining and Applications* 4093 (n.d.): 1025–36.
22. Weber, Rolf H. “Internet of Things – New Security and Privacy Challenges.” *Computer Law & Security Review* 26, no. 1 (January 2010): 23–30. doi:10.1016/j.clsr.2009.11.008.
23. Zhang, Daqiang, Laurence T. Yang, and Hongyu Huang. “Searching in Internet of Things: Vision and Challenges.” In *International Symposium on Parallel and Distributed Processing with Applications*, 0:201–6. Los Alamitos, CA, USA: IEEE Computer Society, 2011. doi:10.1109/ISPA.2011.53.
24. Zhang, Fangjiao, Wei Guo, Jincui Yang, Fangfang Yuan, and Lidong Zhai. “Research on Redundant Channel Model Based on Spatial Correlation in IOT.” In *Trustworthy Computing and Services*, edited by Yuyu Yuan, Xu Wu, and Yueming Lu, 666–72. Communications in Computer and Information Science 320. Springer Berlin Heidelberg, 2013. http://0-link.springer.com.maurice.bgsu.edu/chapter/10.1007/978-3-642-35795-4_84.
25. Zhu, Qian, Ruicong Wang, Qi Chen, Yan Liu, and Weijun Qin. “IOT Gateway: BridgingWireless Sensor Networks into Internet of Things.” In *2010 IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC)*, 347–52, 2010. doi:10.1109/EUC.2010.58.