J. Basic. Appl. Sci. Res., 4(3)270-279, 2014

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Security and Privacy Issues of Implementing Cloud Computing on NDN

Mahsa Paknezhad[1], Dr. Manijeh Keshtgary[2]

[1]Master Student at the Department of Computer Engineering and IT, Shiraz University of Technology
[2]Assistant Professor at the Department of Computer Engineering and IT, Shiraz University of Technology

Received: December 15 2013
Accepted: February 26 2013

ABSTRACT

The significant deficiencies of current internet architecture in providing users' present and future Quality of Service requirements have driven scientists to suggest a different network architecture for the internet. This new architecture is called Named Data Networking (NDN) and works on the basis of distributing named content. Recent studies indicate that the future internet will be in fact content centric. Consequently, many researchers are trying to improve this newly proposed architecture, and offer solutions for the implementation of different essential applications on this content centric network. One of the most important applications currently being utilized by many businesses is Cloud Computing. Cloud Service Providers offer organizations considerable IT services eliminating the need for them to take care of their IT infrastructure. However, current implementations of cloud computing seem to have security and privacy issues making people hesitant about moving to the cloud. These issues stem from vulnerabilities of the cloud architecture itself and the network on which it is implemented. Many studies have been conducted to investigate these issues on the current internet architecture. In this paper, however, we tried to investigate the network-layer related threats of a cloud implemented on NDN. We mentioned previously known security and privacy issues of cloud services and analyzed their effect on this new cloud system to help readers gain an insight into the changing nature of threats and security concerns on this new network architecture. Our study revealed that the influence of some of these issues alleviates on NDN.
KEYWORDS: Named Data Networking; Content Centric Networks; Cloud Computing; Security; Privacy.

INTRODUCTION

Developed in 1960's for resource sharing, the Internet architecture was based on establishing connections between two remote hosts. The main reason for choosing this architecture was due to the very few number of communication devices at that time. However, currently there are lots of inexpensive computers and other communication devices with huge amounts of content on the internet. Today, people value the content while the internet is still working based on where the content is. This translation of what to where has made the internet architecture inefficient [1] and makes it unable to serve the increasing needs and foreseen requirements of companies and individuals[2]. As a result; new architectures have been proposed to access content by a name rather thanthe IP address of the device having the content. These architectures include DONA, PSIRP, SAIL, GENI, FIND and FIA all circling around a network in which content plays the most important role[3]. Content Centric Networking also called Named Data Networking is a recently proposed internet architecture based on named data. This new architecture utilizes network bandwidth efficiently and decreases communication costs for popular contents. It also decreases network congestion and provides a much better security, delivery efficiency and disruption tolerance [1]. The significance of this new architecture is because of scientists' belief that future internet will be content-centric to efficiently handle the content and lead to a new generation of internet [2].

On the other hand, cloud computing has gained great popularity in recent years. It is a growing trend in information technology which provides dynamic, low cost computing solutions [4]. By cloud computing, users can be provided with their software or hardware requirements. They will pay for what they need when they need it, while being supplied with easier monitoring of data and resources for security issues. Moreover, it provides useful utilities such as scanning for vulnerabilities or password assurance tests which frees companies from considering security software engineering when designing their applications [5]. However, outsourcing data and applications on third-party cloud computing platforms has resulted in new security issues which contribute to less trust in moving to this new environment. As a consequence; many enhancements have been proposed recently to improve security and privacy in cloud computing hoping to eliminate these issues that are preventing users from taking advantage of significant benefits of cloud computing.

***Corresponding Author:** Mahsa Paknezhad, Master Student at the Department of Computer Engineering and IT, Shiraz University of Technology. Phone: +98-917-7208909. m.paknezhad@sutech.ac.ir

In this paper, our goal is to analyze the possible security and privacy issues of implementing cloud computing on Content Centric Networks. In the next section, we will give some background about the previous studies on Named Data networking and Cloud computing. Then, we will investigate known security and privacy issues of cloud computing that are related to the underlying network architecture to see their effect on the implementation of cloud on NDN. Finally, a conclusion is provided in section IV.

Background

In contrast to the current internet architecture in which security is an afterthought, NDN provides built-in security mechanism by signing all named data. In this mechanism, data is bound to its name by signing together both the name and the data. This is a mandatory process which results in decoupling trust from the source from which the data was received and the way it was sent to the receiver. Providing a basic security level, it also helps customers to judge whether a public key owner is a good source for a particular piece of data [6]. However, this security mechanism is not supposed to be sufficient for hostile activities that are currently being done on the internet. For instance, it does not provide confidentiality and privacy for users, contributing to lack of trust to the architecture. As a consequence, a number of other solutions have been suggested in this regard. As an example:

[7]Indicates that in order for users to be able to confirm if they have received their requested data, they should be able to assess three properties of the received content: 1) if the received data is a complete, uncorrupted copy of what the publisher sent.2) If the publisher of the data is a person whom the receiver trusts for that specific type of data. 3) If the received data is a reply to the request of the receiver. To allow assessment of the first property the writers suggest naming the content by its cryptographic digest. However, this method of naming provides no way to investigate the relevance of data. Another solution which provides some control over naming is that publishers need to select a unique, user-friendly label for each piece of content and include this name as well as the publisher's public key in the content's name. As a result, for the content C the name may be $Digest(Pubkey_p||Label_C)$ or $Digest\ (Pubkey_p)||Label_C$. Just the same, this solution does not allow assessment of relevance either. In order to enable assessment of relevance, it is suggested that publishers sign the mapping from name N to Content C. Consequently, the content will be available on the network in the form of $M_{(N,P,C)} = (N,C, Sign_P\ (N,C))$.Also, [8] offers an overlay anonymization network so as to provide efficient privacy for users on Content Centric Networks. This anonymizing technique utilizes onion routing that deploys layers of concentric encryption which are peeled off as packets travel through the network. By routing packets through at least two onion routers, this anonymizer named ANDANA decouples requested content from the user. Moreover, providing secure audio conferencing on CCNs is investigated in [9]. Although decentralized, this security mechanism offers source authentication, participant control and private conferencing. In this mechanism, securing communications is achieved by public key cryptography. As a result, each party owns one or multiple keys. Furthermore, adding or removing participants and enforcing participant control policies are carried out only by an entity named Organizer. More importantly, a number of studies have been conducted to find out what security and privacy issues Content Centric Networks are confronted with. Some solutions are also offered in these studies which; although do not eliminate those issues; can mitigate their effects[9, 10, 11].

On the other hand, there are a wide range of security and privacy issues revolving around cloud computing making companies less confident about moving to the cloud. As a result, scientists have offered many suggestions to improve security of cloud computing. For instance, [12]suggests a protection system named Advanced Cloud Protection System to monitor security of the cloud and take effective steps in case of attacks. This protection system remains transparent to the cloud users so as to avoid becoming the target of attackers. Also, [13] discusses the importance of offering a new service on the cloud in the name of Security-as-a-Service. Moreover, a number of cloud service providers and research groups such as Alert Logic, Cloud Security Alliance, and security lab of the University of Arizona publish some periodicals to give cloud-based companies and users some advice about what measures they should take to improve the security and privacy of their cloud infrastructure. Besides, many studies have been carried out to help users gain a deep insight into the threats they are faced with on the cloud[4, 5, 14, 15].

Consequently, there are a large number of security and privacy issues on both cloud computing and NDN. In the next section, we will analyze what the security and privacy problems can be when implementing cloud on NDN, and how they can be different from security and privacy issues of cloud on TCP/IP architecture. We hope to be able to give a comprehensive image of these issues on both the infrastructures.

Security and Privacy Issues of Implementing Cloud Computing on Content Centric Networks

There are a wide range of security and privacy issues on cloud computing. According to a study conducted by Alert Logic, attacks on cloud services can be categorized into 6 incident classes, namely Application Attacks, Brute Force, Malware/Botnet Activity, Reconnaissance, Vulnerability Scan, and Web Application Attacks[16]. The existence of these attacks on the cloud is obviously due to the exposure of cloud services to the internet enabling

hackers to access the cloud and conduct various attacks on it. In this section, we will discuss security and privacy issues of cloud computing related to the underlying network architecture. We will investigate these issues onContent Centric Networks and the solutions (if any) offered by scientists for these attacks. In the following, the threats are classified based on the layer they act on.

1- Physical & Data Link Layer

1-1 Network Sniffing

If the data that is being transferred on the network is not encrypted, applications that capture packets traveling on the network can read sensitive information resulting in Network Sniffing. There are also sniffer programs that can record data sent or received from the network through the NIC [14]. To handle this problem, encryption methods are used for securing the data [15]. Furthermore, there is a sniffing detection platform based on address resolution protocol and round trip time to detect sniffing activities [14]. Other solutions for this problem are using Secure Socket Layer (SSL), Transport Layer Security (TLS), or Secure Internet Protocol (IPsec) [5]. Network sniffing attacks in NDN can be avoided by utilizing appropriate encryption mechanisms as in the TCP/IP architecture, except that not only the content but also some part of the name needs to be encrypted to provide privacy for users similar to what is done in ANDANA.

2- Network Layer

2-1 BGP Prefix Hijacking

In this type of attack wrong announcements for IP addresses in an Autonomous System are made. These announcements enable malicious parties to have access to untraceable IP addresses. As a result, data may not be routed to the intended destination. Instead, they are delivered to some other IP contributing to information leakage. Cryptographic authentication of routing updates seems to be inefficient on TCP/IP architecture due to the absence of a global PKI infrastructure, and high computational overhead of signature verification [17]. However, a distributed anomaly detection and response system is described in [18] for providing security for BGP Algorithm. This approach named Pretty Good BGP does not eliminate BGP Hijacking attacks, but proves to be more effective than cryptographic solutions.

BGP is also suggested to be used as the Inter-AS routing protocol on NDN after bearing some minor modifications in order to become adaptable to Content Centric Networks [6]. Similar to the efforts made to provide security for BGP on TCP/IP architecture, securing this routing protocol on NDN is also a key issue investigated by scientists. The same mechanism utilized for securing named content is used to prevent BGP prefix hijacking on NDN. Each NDN router whose name follows the network management hierarchy will hold a public key that is certified by an anchor key configured by the network operator. Each interface on the router will also have an interface key which is certifiable by the router key. In this case, router updates will be signed by the interface key of the interface from which they originate allowing receivers to check the authenticity of the signed update information [19]. As this mechanism is the main security strategy on NDN and thanks to the Robust Trust Model [20] and Key Management System [21] offered for this network architecture, it will have no problems in terms of the existence of a global PKI infrastructure.

2-2 DHCP Attack

Overall, there are three different DHCP attacks. DHCP Server Spoofing attack is an attack in which the DHCP server is a compromised system. In this attack, the server may give client machines DHCP based information and use it to sniff connections. This can be done by setting the DNS addresses and default gateway values with the IP address of a machine with a sniffer software installed [22]. Another attack called DHCP consumption attack consumes all the available IP addresses in the DHCP server pool to prevent client machines from being assigned IP addresses and therefore implement a DoS attack [23]. Finally, in IP Address Hijacking attack the attacker tries to release an authorized IP address assigned to a machine by the DHCP server by sending a DHCPRELEASE packet with the same IP address for the server to release that IP address and then lease that same IP address. In this way, the attacker can cause disruption in network communications [22].

Providing DHCP-like mechanism on NDN is a challenge that scientists are working on at the present time. NDN Auto Configuration (NAC) [24] is a proposed solution offered recently to provide this mechanism. In this solution, current DHCP servers are utilized to provide NDN configuration parameters such as NDN NAC server address, NDN gateway address, and usable NDN namespace for the client. For this purpose, the client uses different options on DHCP to request NDN parameters. In case the DHCP server supports NDN, it will reply to the client with NDN options. However, if the DHCP server does not support NDN, the client broadcasts DHCPINFORM to locate NDN servers. In this situation, only NDN servers reply with a DHCPACK packet

followed by providing the client with NDN parameters. If there are no NDN servers in the local subnet, the client queries NDN server address from the DNS server and then unicasts DHCPINFORM to that NDN server. Implementing an attack similar to DHCP Spoofing seems to be less possible on NDN mainly because the configuration parameters offered by NDN server to the client are not sufficient to enable the hacker to conduct sniffing on the client's traffic. Clients receive encryption keys from a server other than the NAC server, and since each and every packet is encrypted by the client's key, sniffing will be impossible unless the hacker succeeds in acquiring the encryption key in some other way.

As routing is based on named data in NDN, content name is hierarchically structured to make routing scale to the existing hardware limitations. For instance, the name of a video produced by PARC may be /parc/videos/WidgetA.mpg which, as can be seen, consists of a number of name components [6]. Although there may be limitations on the top level name components of the NDN namespace in a subnet, yet a vast variety of different names can be constructed and assigned to different clients. As a result, determining an upper bound for the number of usable names on a subnet is impossible contributing to the inability of implementing DHCP Consumption attacks on NDN. Also, this way there will be no need to have a name release mechanism (although we can have one) when a client leaves the network avoiding the hijacking attack described above.

2-3 Outage due to Congestion

If congestion happens on the network, there is always the problem of outage from a cloud environment. To provide security against outage, it is significant to ensure load balance service, in addition to the requirement for the data to be encrypted before being placed on the cloud [14]. Congestion control mechanism on TCP/IP alleviates this problem up to a point. However, this mechanism is different on NDN. On NDN, routers can prevent congestion by controlling the PIT size. This is done by each and every router, eliminating the need for end hosts to do that. Moreover, caching mitigates congestion by helping data retransmission. In case a piece of data is dropped on the way to the destination due to congestion, retransmitting the interest packet for that piece of data will make it to travel the path from where it was dropped to the destination, since possibly a copy of that data has been cached in the existing caches on the way before being dropped. Consequently, avoiding repeated retransmission of data from the original producer helps NDN to prevent congestion collapse that may occur on today's internet decreasing the probability of outage in the network [6], and preventing attackers from abusing this vulnerability.

3- Transport Layer

3-1 Man-in-the-Middle Attack

This attack usually happens on TCP/IP if secure socket layer is not correctly configured[15].There are also other strong encryption technologies offered such as Dsniff, Cain, Ettercap, Wsniff, and Airjack[14]to prevent this type of attack on TCP/IP architecture.

Although, removing source and destination addresses in data packets improves privacy, CCN architecture results in some privacy issues mainly due to semantic richness of names. As names are semantically related to the content, they provide more information than IP addresses. Also, one can easily find out when two requests seek for the same content even if they are encrypted. Moreover, on CCN encryption is not applied on public content. Consequently, users can not count on encryption to hide whatever public data they access. Furthermore, hosts connecting to the same access router can determine accesses by their neighbors taking advantage of timing information. Finally, as each and every piece of content should be signed by its publisher in NDN, the publisher's identity can leak sensitive information about the content. In order to address these issues, [8]offers an anonymizer called ANDANA briefly introduced in the previous section. It proves to handle these privacy challenges efficiently in NDN. However, there is still some vulnerability in this proposed solution. Since ANDANA is an open network, attackers can deploy compromised routers or compromise existing routers to interrupt the normal procedure of ANDANA by showing bad behavior such as injection, delay, or dropping packets. The attackers can also deploy compromised caches or compromise existing caches to monitor cache requests and reply with corrupted data. Moreover, a link carrying anonymizer traffic can be sniffed by the attacker to gather useful information such as traffic patterns or replay them by a compromised router. Even so, the time required to compromise each element of this system is more than the time it takes to send an interest and receive the corresponding data. Consequently, the information an attacker gains by compromising a router only refers to packets received by the router after the compromising decision on that router is made. Finally, there may be the possibility to relate an outgoing packet to a consumer due to the low latency of some onion routing networks, but as ANDANA uses caches on routers and ephemeral circuits which carry only one or a few number of interests and their corresponding data, implementing this attack is much harder. In brief, ANDANA proves to be a good solution for the privacy issues mentioned above.

Once again the global PKI infrastructure of NDN benefits this network architecture to prevent another common type of attack on today's internet and that is the Man-in-the-Middle attack. Preventing this attack on TCP/IP architecture was carried out by encrypting transmitting content in a communication with the other side's public key or encrypting the content with an optional symmetric key and then encrypting the symmetric key with the other side's public key[25]. The problem with this mechanism was that one was not sure if the public key he/she encrypted the data with was in fact the public key of the person he/she wanted to communicate with. In other words, there was no global key management infrastructure to verify that a public key was owned by the person who claimed that. However, the global PKI infrastructure in CCNs which is the main mechanism to provide security, privacy, integrity, provenance, and access control in NDN helps to solve this problem.

There are also some other privacy attacks taking advantage of the caching mechanism of NDN. These attacks are generally called cache snooping attacks and are briefly discussed in the following.

- Cache Monitoring Attack: In this attack the adversary needs to be a neighbor of the victim so that he is attached to the same access router. He also needs to have some prior information about the victim to know the name of a sensitive object which he wants to find out if, how often, and when the victim has accessed it. This can be done by the adversary by periodically sending requests for that object to see if it is cached in the access router. Determining whether an object is cached in the router or not can be done by measuring the time it takes to receive the requested object. To prevent this type of attack, onetime names can be used to make object names unpredictable. Also, tunneling mechanisms such as ANDANA can be utilized to prevent caching in the routers. Moreover, we can disable scope field in the interest packets to prevent detection of cache hits. Cached data can also be sent with some artificial delay to make cache hit detection by using timing information more difficult. However, these solutions only mitigate this attack and will reduce the functionality of NDN. Other solutions can be placing caches only at the higher aggregation levels to prevent attributing a specific request to a specific user or using tunneling only for sensitive information[26].
- Object Discovery Attack: In this attack the adversary gets to know the objects that are cached in the access router. For this purpose, he utilizes two features of interest packets in NDN, namely prefix matching and exclusion patterns. By prefix matching the adversary can find out the name of an object he didn't know previously. This way he can retrieve objects stored in the cache whose name is not known to him. Exclusion patterns can also help him to ask for cached objects he has not seen yet. He can put the name of visited objects in the exclusion list of the next requests to discover other cached objects in the router. However, routers can detect such attacks easily since they encompass requests with long exclusion lists[26]. Also, a kind of prevention strategy can be to change the protocol specification to let data objects determine which parts of their name the exclusion mechanism can work on, or to completely disable exclusion mechanism and give the responsibility of object discovery to another layer in which better security can be provided. Also, unsuccessful number of prefix matches can be restricted [10].
- Flow Cloning Attack: This attack helps the adversary to receive a replicate of an entire data flow to the victim's machine. By using Object Discovery attack, the adversary discovers the prefix of an ongoing data flow. Then he can predict the names of the future packets and request them in the same way as the victim. Although these packets may be encrypted, the size or time of the dataflow can still give some useful information to the attacker. However, if multiple data flows with same prefixes are traveling in the network, the adversary may not be able to make a distinction between them. Solutions for this attack are encrypting as much of the names of the packets in the dataflow as possible and the suggestions provided for preventing Object Discovery attack[26].In case of conversations in which sequence numbers are used to name the next pieces of data, the attacker will not be able to guess the name of the following pieces of data if the numbering scheme is not known by the attacker, or that part of names is encrypted with a secret key. If encryption is not utilized for the numbering part of names, the naming scheme should be difficult to guess for an outsider. Finally, for dynamic sessions it is proposed to cache content for a very short period of time since it is unlikely to be shared. Also, it makes attacks much difficult, since adversaries will have no choice but to carry out the attack in real time [10].

3-2 Scanning Attacks

- Unauthorized Nessus Scan

  Nessus Vulnerability Scanner is a free of charge scanner developed by Tenable Network Security. It is utilized to discover security problems of a software product from the internet with no initial information, from the internet with some initial knowledge, from a trusted network, or from a trusted network with full privileges. Nessus is regarded to be the most popular vulnerability scanner used by over 75000

organizations [27]. To use this scanner some prior information should be provided including the IP addresses of the systems whose security is under study. However, scanning wrong IP addresses can violate personal freedom [28].

- Port Scan
  Port scanning is the process of sending a message to each and every port on a system and waiting for a reply. The received message includes the port status and some useful information for launching attacks such as the operating system of the machine [29].

As mentioned before, in order to target a specific host on NDN similar to using the IP address of a host on TCP/IP, we need to target the namespace hosted by that machine. As a result, defining the hosts which we would like to execute a Nessus Scan on can be done by determining the namespaces those hosts hold. Moreover, the difficulty of implementing port scanning which is also a part of Nessus Scan [30] depends on our application architecture, whether we are still using ports in NDN or not. Using ports for various applications is optional on NDN. This is mainly because we are allowed to consider different names for our services instead, eliminating the need for using different port numbers. In this case, scanning for all the possible names that could be assigned to these services will be hard. However, if standardized names are considered for services such as email, file transfer, or web, scanning for vulnerabilities on NDN will be as easy as port scanning on TCP/IP architecture. Furthermore, sophisticated mechanisms can be designed to decide whether to respond to an interest as it may be part of a scanning attack.

4- Application Layer

4-1 DNS Attack

DNS attacks can be deployed by rerouting the path between a sender and a receiver and utilizing measures such as Domain Name System Security Extensions (DNSSEC) does not eliminate these kinds of attacks but only reduces their effects. Issue of reused IP addresses can also contribute to a kind of DNS attack, which is mainly because when an IP address is changed in DNS, it takes some time for that change to be applied on DNS caches. To illustrate, consider that an old IP address is assigned to a new user. In this case, some users may still use the old IP address retrieved from an outdated DNS cache and have the chance to access the data belonging to that user violating user privacy[14].

However, these attacks are not possible in NDN. This is mainly because rerouting the path between the sender and the receiver may change the place where the response comes from. A response from a different place will be signed with a key other than the expected key belonging to the original producer, but as the client verifies received packets, he/she will not accept a response with a wrong signature. Also, reusing namespaces does not contribute to any privacy issues since namespaces come with keys, and keys can change when namespaces are re-allocated.

4-2 Denial of Service (DoS)Attack

By flooding the network with useless traffic or overwhelming a server with huge amounts of requests, this attack tries to prevent service delivery to legitimate users. Intrusion Detection Systems (IDS) are the most popular method suggested to be used on TCP/IP networks to prevent DoS attacks on cloud infrastructure. Each cloud will have a separate IDS which works by exchanging information. If a specific cloud is under attack, a co-operative IDS warns the entire system and a decision is made about that cloud by voting without affecting the overall system performance[14].Also, filtering mechanisms may be used to drop packets with similar IP source addresses or server requests[5].

DoS attacks can be implemented on NDN by a vast variety of methods:

- DoS by Forcing Expensive Computations: This attack makes use of the fact that NDN routers verify the signatures of packets they receive, to slow them down or stop them from verifying signatures. The hacker sends requests for data to a malicious content source constantly. In order to make the verification process time consuming, the content source uses different keys for signing each piece of data to make the router require retrieving different keys for verification of each packet. Also, the content source may delay its responses, or use authentication mechanisms that take the worst case time. In this situation, being busy verifying packets, the router may not be able to deliver services to legitimate users. It may also stop verifying signatures to handle other requests for data. Solutions provided for this security threat are: stopping verifying signatures by high loaded routers and giving this responsibility to the end systems, delaying verification until there is processing power available, verifying contents only after they are cached for a specified amount of time, detecting content sources that use different keys for signing and stop verifying their packets or do the verification only when there is processing power available[10].

- DoS with Special Bits: There are special bits in the interest packet that can avoid cache hits in routers. Attackers can benefit from this feature and overwhelm the content source by sending huge amounts of interest packets to the content source with this feature set. To avoid this attack making use of these bits should be limited to a local scope. Only signed interest packets can be allowed to use this option[10].
- DoS by Decreasing the Efficiency of Caching: In this attack, which is also called Cache Pollution attack [11],the attacker sends requests for a specific unpopular content periodically to create fake popularity for that content making that content fill the cache space in routers. Implementing many of these attacks at the same time can increase bandwidth requirements contributing denial-of-service in the network or at the content source. Also, an algorithm for detecting this type of attack is provided in [31]. A more expensive solution can be providing infrastructure for the worst case which means when the cache hit rate of routers is zero[10].
- DoS by Filling Available Memory of a Router: denial of service to legitimate users can happen when the attacker fills up the PIT of a router by asking for a large number of non-existent names. For making the attack indistinguishable for the router, the attacker colludes with a content source to deliver the data just before the entry for the interest packet in the PIT times out. Using only one content source can also help routers detect the attack. However, the attacker can utilize several machines to act as content sources preventing detection of the attack by the router. Using hashing techniques to save memory, dropping interest packets at the head of the PIT, having an algorithm to detect a huge amount of requests for a few numbers of content sources are some of the solutions provided for this type of attack[10].

There are still some other attacks that can contribute to denial of service in NDN such as compromised routers that do not forward interest packets, configuring routers with different timeouts avoiding content from being retrieved, replaying “a content does not exist” response or an old content with the right key by the attacker, and generating fake responses with no or wrong signature hoping that the receiver does no verify it.Finally, it is worth to mention a main disadvantage of NDN in terms of providing security, which is that a CCN host cannot determine the source of an interest packet to identify the origin of an attack [10].

4-3 Distributed Denial of Service (DDoS) Attack

This is a type of DoS attack with multiple compromised systems. These systems are used to implement a DoS attack on a single system. In order to protect against these attacks, a number of actions can be taken. For instance, a swarm-based logic is proposed in [38]that provides a transparent transport layer. Common protocols such as HTTP and SMTP can pass through this layer easily. Also, having intrusion detection systems on all the machines which hold virtual machines can be very useful to protect the cloud against DDoS attacks on TCP/IP architecture.

In NDN, attackers cannot target a host as it is done in TCP/IP architecture using IP addresses. Therefore, implementing a DDoS attack can only be achieved by targeting a specific namespace hosted by the target system by flooding interest packets for unique, unpopular or non-existent pieces of data from that namespace. As a result, this attack changes its name to Interest Flooding Attack on CCNs. To limit the number of interest packets passing through the network, token buckets can be used at each interface of the routers. Also, the Pending Interest Table can be changed to support flagging of interests that cannot be forwarded immediately. However, this method still forwards a considerable number of interests from attackers. As a result, it is suggested to calculate and maintain an up-to-date Interest Satisfaction ratio for each incoming interface. This value determines what percent of the forwarded interests have been satisfied till now. It works as a probability to forward an incoming interest or drop it. Just the same, there is one problem with this solution. Since routers independently decide whether to forward an interest or drop it, the probability of interests reaching their destinations significantly decreases. To address this problem routers announce these limits to their downstream routers to make sure an interest forwarded by the downstream router is permitted to pass. Moreover, separate incoming and outgoing limits are designated for each FIB entry. This can help to mitigate the effect of Interest Flooding Attack on the network[32].

4-4 Session Attack

- Session Riding

  In this type of attack, hackers send emails to users and persuade them to take a specific action such as clicking on a link. In this way, they will be able to send commands to a web application, delete user data, change system or network configurations, open a firewall, or execute online transactions on behalf of the user[33].

- Session Hijacking

  In session attacks, the most important goal is to capture the session identifier (SID) in the application layer or Sequence number and Acknowledgement number in the network layer. There are three different ways to

obtain session identifier, namely Session Prediction, Session Capture, and Session Fixation[34]. Examples of Session Capture attacks are: Session Hijacking by IP Spoofing, by Blind Attack, or by Man-in-the-Middle attack. In IP Spoofing, the hacker uses the IP address of a trusted client to inject his own packets. As the server will expect a sequence number and an acknowledgement number in the packets, the hacker needs to change them since he has no knowledge about the value of these two numbers. For this purpose, the hacker inserts its own packets into the session before the original client can respond making the session desynchronized. This makes the server expect a different sequence number. Consequently, the packets of the original client will become unacceptable. In case a hacker cannot sniff the packets in order to get the correct sequence number expected by the server, he has no choice but to deploy the Blind Attack that is brute forcing 4 billion possible sequence numbers. In the Man-in-The-Middle method, the hacker uses a packet sniffer to read the TCP header of packets and get the value of the expected sequence number, the acknowledgement number, the ports and the protocol numbers. Then, he can forge the packets and send them to the server before the client does. Also, the hacker can change the default gateway of the client machine by ARP spoofing so that packets will pass through the hacker's machine while being sent to the server. Session IDs can also be captured by sniffing Cookies and URLs from unencrypted packets flowing in the network[35].

One of the significant advantages of NDN, however, is that many applications can be implemented on this internet architecture without using session semantic due to the content-centric property of NDN. As an example, currently proposed architectures for live streaming[36],vehicle-to-vehicle communications [37]and audio conference tools [9]are session-less which eliminates concerns revolving around session attacks. Even so, there are still applications such as ANDANA [8]that are suggested to be implemented on NDN using sessions. Considering these applications, acquiring session ids, sequence numbers and acknowledge numbers will be yet much more difficult on NDN than TCP/IP architecture. This is mainly because in contrast to the TCP/IP architecture in which only the communication channel between source and destination is secured, data packets are secured directly using encryption on NDN. As a result, CCNs appear to be less vulnerable against session attacks.

Conclusion& Future Work

In this paper, we investigated security and privacy issues of cloud computing that are related to the underlying network architecture. We mentioned the attacks that are being carried out on current implementations of cloud on TCP/IP architecture, and analyzed their existence and effect when implemented on Named Data Networks. Our analysis reveals that NDN's main functionality and mechanism helps to decrease the harmful impacts of attacks including BGP Prefix Hijacking, DHCP attacks, Session attacks, and DNS attacks and reduces the probability of Outage due to congestion in the network. However, other attacks such as Snooping attacks, DoS and DDoS attacks, and scanning attacks can be conducted in some other ways on NDN allowing them to still remain as important threats for cloud computing on Content Centric Networks. For our future work, we are planning to provide better solutions for some of these threats on NDN. We hope to be able to offer new strategies to mitigate these attacks by utilizing features of this new architecture.

Acknowledgement

We would like to thank Dr. Christos Papadopoulos for his valuable advice throughout this research.

REFERENCES

[1]. Jacobson, V., D. K. Smetters, J. D. Thornton, M. F. Plass,N. H. Briggs, and R. L. Braynard, 2009. Networking Named Content**,** In Proceedings of the 5th international conference on Emerging networking experiments and technologies pp. 1-12. ACM.

[2]. Future Content Networks Group, 2009. Why do we need a Content-Centric Future Internet? Proposals towards Content-Centric Internet Architectures, European Commission, Information Society and Media. Retrieved December 9, 2012, from http://www.future-internet.eu/.

[3]. Carofiglio, G., M. Galloy,L. Muscarielloy, and D. Perino, 2011. Modeling Data Transfer in Content-Centric Networking.In Proceedings of the 23rd International Teletraffic Congress pp. 111-118. ITCP.

[4]. Schultz, M. J., 2011. A Survey of Cloud Security Issues and Offerings. Retrieved May 3, 2013, from http://www.cse.wustl.edu/~jain/cse571-11/ftp/cloud/index.html.

[5]. Andrei, T., 2009. Cloud Computing Challenges and Related Security Issues. A Survey Paper. DOI= http://www. cse. wustl. edu/~ jain/cse571-09/ftp/cloud. Pdf, retrieved April 29, 2013.

[6]. Zhang, L., D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, D. K. Smetters, B. Zhang, G. Tsudik, K. Claffy, D. Krioukov, D. Massey, C. Papadopoulos, T. Abdelzaher, L. Wang, and P. Crowley, 2010.Named Data Networking (NDN) Project. RelatórioTécnico NDN-0001, Xerox Palo Alto Research Center-PARC.

[7]. Yeh, E., D. Smetters, and V. Jacobson, 2009.Securing Network Content. Palo Alto Research Center Incorporated. RelatórioTécnico TR-2009-1, Xerox Palo Alto Research Center-PARC.

[8]. DiBenedett, S., P. Gasti, G. Tsudik, and E. Uzun, 2011. ANDaNA: Anonymous Named Data Networking Application. arXiv preprint arXiv:1112.2205.

[9]. Zhu, Z., P. Gasti, Y. Lu, J. Burke, V. Jacobson, and L. Zhang, 2011. A New Approach to Securing Audio Conference Tools. In Proceedings of the 7th Asian Internet Engineering Conference. pp. 120-123. ACM.

[10]. Lauinger, T., 2010. Security & Scalability of Content-Centric Networking, Doctoral dissertation, TU Darmstadt.

[11]. Lauinger, T., N. Laoutaris, P. Rodriguez, T. Strufe, E. Biersack, and E. Kirda, 2012. Privacy Risks in Named Data Networking: What is the Cost of Performance? ACM SIGCOMM Computer Communication Review, 42(5), 54-57.

[12]. Lombardi, F., and R. D. Pietro,2010. Secure virtualization for cloud computing.Journal of Network and Computer Applications, 34(4), 1113-1122.

[13]. Alert Logic, 2011. The Inevitability of Security-as-a-Service, Alert Logic Inc. Retrieved May 3, 2013, from http://www.alertlogic.com/wp-content/uploads/2012/01/Inevitability-of-Security-as-a-Service.pdf.

[14]. Bhadauria, R., R. Chaki, N. Chaki, and S. Sanyal, 2011. A Survey on Security Issues in Cloud Computing.arXiv preprint arXiv:1109.5388.

[15]. Qaisar, S., and K. F. Khawaja, 2012. Cloud Computing: Network/Security Threats and Countermeasures, Interdisciplinary Journal of Contemporary Research In Business, 3(9), 1323-1329.

[16]. Coty, S., T. Borland, M. Gupta, and P. Snyder, 2013. State of Cloud Security Report, Alert Logic Inc. Retrieved May 22, 2013, from http://www.alertlogic.com/resources/cloud-security-report/.

[17]. Oliveira, R., M. Lad, and L. Zhang, 2009. Understanding the Challenges in Securing Internet Routing. In Applications and the Internet, 2009. SAINT'09. Ninth Annual International Symposium on (pp. 145-148). IEEE.

[18]. Karlin, J., S. Forrest, and J. Rexford, 2008. Autonomous Security for Autonomous Systems. Computer Networks, 52(15), 2908-2923.

[19]. NSF FIA PI Meeting, 2011. Named Data Networking. Retrieved May 22, 2013, from www.named-data.net.

[20]. Pournaghshband, V., and K. Natarajan, 2011. A Robust Trust Model for Named-Data Networks, University of California, Los Angeles.

[21]. Bian, C., Z. Zhu, E. Uzun, and L. Zhang, 2012. Deploying Key Management in NDN Testbed, Technical Report NDN-0009.

[22]. Astorino, J.,2011. Going Deep With DHCP Snooping, Astorino Networks. Retrieved May 22, 2013, from http://astorinonetworks.com/2011/06/28/going-deep-with-dhcp-snooping.

[23]. Lauerman, K., and J. King, 2010. Layer 2 Attacks and Mitigation Techniques for the Cisco Catalyst 6500 Series Switches Running Cisco IOS Software, Cisco Public Information.

[24]. D. Massey, 2012. Information Centric Networks and Named Data Networking. Colorado State University. Retrieved May 22, 2013, from www.cybera.ca/webfm_send/161.

[25]. Kurose, J., K. Ross, and A. Wesley,2007. *Computer Networking: A Top Down Approach 4th edition*, Ch. 8.

[26]. Lauinger, T., N. Laoutaris, P. Rodriguez, T. Strufe, E. Biersack, and E. Kirda, 2012. Privacy Implications of Ubiquitous Caching in Named Data Networking Architectures, Technical Report TR-iSecLab-0812-001.

[27]. LeMay, R., 2005. Nessus security tool closes its source, Cnet. Retrieved May 22, 2013, from http://news.cnet.com/Nessus-security-tool-closes-its-source/2100-7344_3-5890093.html.

[28]. Anderson, H., 2003. Nessus, Part 2: Scanning, Symantec. Retrieved May 22, 2013, from http://www.symantec.com/connect/articles/nessus-part-2-scanning.

[29]. Lee, C. B., C. Roedel, and E. Silenok, 2003. Detection and Characterization of Port Scan Attacks. Univeristy of California, Department of Computer Science and Engineering.

[30]. Schmelzel, P., 2013. Global Information Assurance Certification Paper, SANS Institute. Retrieved May 22, 2013, from http://www.giac.org/paper/gcux/233/global-information-assurance-certification-certified-unix-security-administrator/105841.

[31]. Deng, L., Y. Gao, Y. Chen, and A. Kuzmanovic, 2008. Pollution attacks and defenses for Internet caching systems, *Computer Networks*, vol. 52, no. 5, pp. 935–956.

[32]. Afanasyev, A., P. Mahadevan, I. Moiseenko, E. Uzuny, and L. Zhang, 2013. Interest Flooding Attack and Countermeasures in Named Data Networking, *IFIP Networking*.

[33]. Schreiber, T., 2004. Session Riding: A Widespread Vulnerability in Today's Web Applications, SecureNet GmbH, MünchnerTechnologiezentrum.

[34]. Clay, 2013. Solutions to Session Attacks, Hungred Dot Com. Retrieved May 22, 2013, from http://hungred.com/useful-information/solutions-session-attacks/.

[35]. Kapoor, S., 2006. Session Hijacking Exploiting TCP, UDP and HTTP Sessions. Retrieved May 22, 2013, from http://www.infosecwriters.com/text_resources/pdf/SKapoor_SessionHijacking.pdf.

[36]. Xu, H., Z. Chen, R. Chen, and J. Cao, 2012. Live Streaming with Content Centric Networking. Retrieved May 22, 2013, from www.mit.edu/~caoj/pub/doc/jcao_c_live.pdf.

[37]. Wang, L., R. Wakikawa, R. Kuntz, R. Vuyyuru, and L. Zhang, 2012. Data Naming in Vehicle-to-Vehicle Communications. Retrieved May 22, 2013, from http://named-data.net.

[38]. Lua, R., and K. C. Yow, 2011. Mitigating DDoS Attacks with Transparent and Intelligent Fast-Flux Swarm Network.*IEEE Network*, vol. 25, no. 4, pp. 28-33.