MSc Research Review Report

Named Data Networking (NDN) tunneling over IP

Unit Name: Engineering Research Skills
Unit Code: EENGM0004
Student Name: Jiadi Li
UoB Account: eb21997
Date: 1st May 2022
Supervisor: Dr. Rasheed Hussain

1 Introduction

TCP/IP is a communication model that solves conversation between exactly two hosts, in current internet, which is coming in all fields as a foundational network infrastructure. In 1973, Bob Kahn and Vint Cerf joined forces to bring the worldÂ TCP/IP(Transmission Conversion Protocol and Internet protocol) and it already turns to 40 years old now. A typical widely used architecture based on TCP/IP is known as client-server model in which one machine intends to request the resources in another machine by send request packets and receive response packets. An IP packet contains two assigned IP addresses in the header which are the source address and the destination address. The unparalleled success of TCP/IP is due to a wide variety of factors. Primarily, these include its technical features such as routing-friendly design, scalability and reliability, also, its commercial features such its open standard and development process. One important reason why TCP/IP is so successful is World Wide Web, HTTP protocol mainly relied by World Wide Web uses TCP as a transport protocol to ensure reliability. TCP defines a communication standard which is designed to send packets across the internet and ensure the successful delivery of data and messages over networks. It perform a connection operation by method called âthree-way-handshakeâ before transmission. Another reason why is widely used is because its architecture called âthin waistâ that allows the minimal implementing of the functionality necessary for globally scalable, it allows the lower and upper layer technologies to innovate without unnecessary constraints. But when the founders designing a IP network, security issues were not considered originally. The security strategies of IP were modified later which is implemented by securing the communication channel but not the data itself.

When exabytes of new content are produced periodically, content and connected devices exponentially increase, IP, which is designed for conversation between communication endpoints, is overwhelmingly used for content distribution(IP can only name communication endpoints). Named Data Networking was brought forward by Lixia Zhang and her teams in [1] which is one of the most successful instances of Information Centric Networking(ICN/CCN). Named Data Network is designed as an evolutionary project that will replace the IP architecture that generalizes all data transmissions from previously using address-to-address to become "fetching" the data identified by a name. The current Internet hourglass architecture (thin waist architecture) shown in figure 1 allows the top and bottom layers of the "Thin Waist' to innovate independently. NDN specifically replaces network communications from packet delivery previously based on IP as the basis of delivery, becoming content as the basis for delivery. In contrast to IP-based communication, NDN is data-oriented rather than establishing a session between two endpoints. At the same time, NDN have a build-in security itself. Each data is signed with its name and bind with a mandatory Data signature.

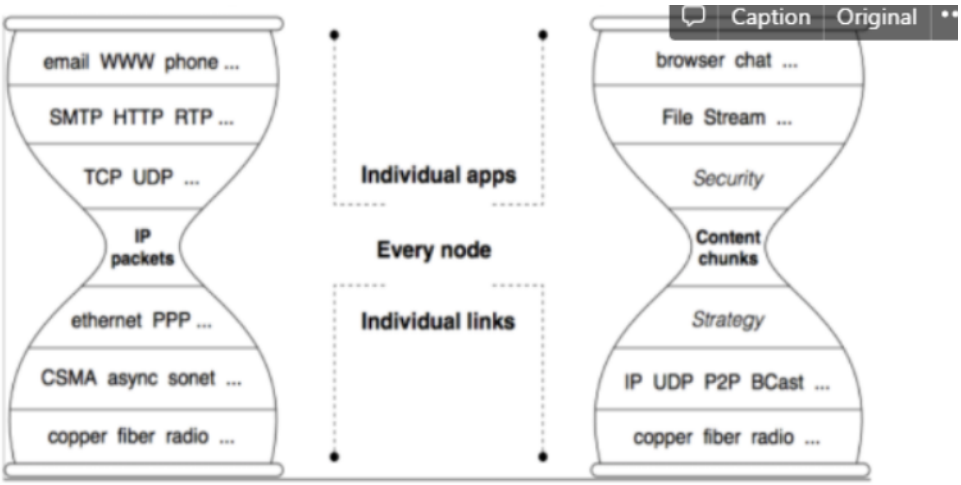


Figure 1. Internet and NDN Hourglass Architecture

NDN is a receiver-driven content-oriented communication model. NDN uses a human-readable and hierarchical naming scheme to identify resources, known as Data Name Prefixes. NDN uses three main types of entities to deliver the information; consumer, producer and router. NDN uses the routing protocol to reach every Data Name Prefixes, instead of carrying source and destination IP address in each packet, NDN puts a data name in each packet. There are two kinds of packets: Interest packets and Data packets. Data Consumer generates and sends an interest packet containing the specified content name to the router. Each NDN node forwards the Internet Packets based on the specified name, records the interface information from which the Interest Packet is received, and is stored in the Pending Interest Table. After the Interest Packet finds the appropriate Data Packet, the copy of the Data Packet is also stored in the nodeâs Content Store(CS). If there is the same Interest Packet request from another client, the node will respond with the Data Packet in its Content Store.

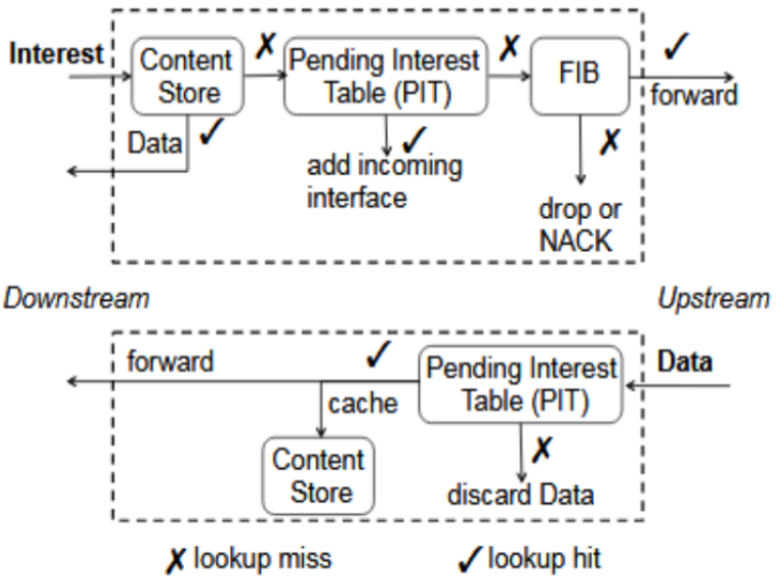
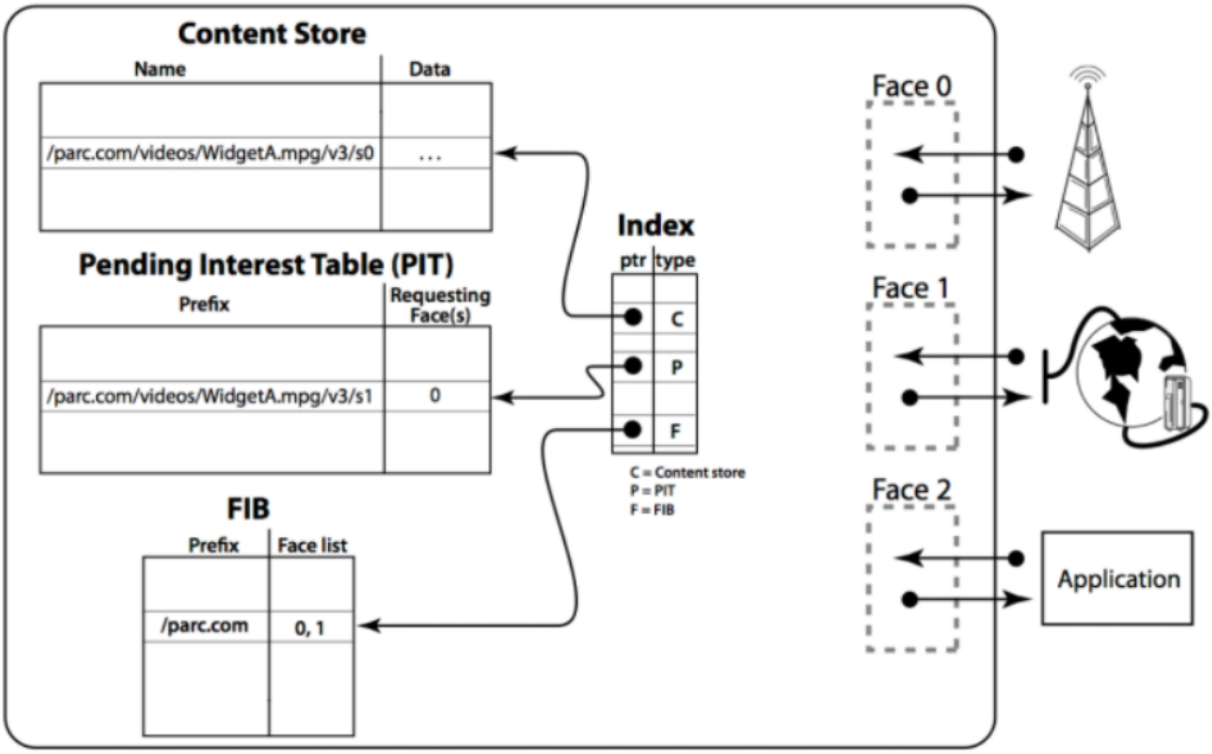


Figure 2. Forwarding Process in NDN Router [1]



The NDN packet forwarding engine includes three key components to achieve the above functions: Pending Interest Table(PIT), Forwarding Information Base(FIB) and Content Store(CS). PIT maps an identification of interest with its incoming interface in order to send it back the corresponding data. The next-hop forwarding face is determined by looking up name prefixes in FIB, which is empowered by name-based routing protocol. CS is employed to cache data packets for an identical request to improve performance. Below is a work flow: when a NDN router receives interest packet from down-stream routers or consumers, it firstly checks the interestâs name in the CS. If found, the router returns this data through the incoming interface, otherwise it searches for the packet name in PIT. If a match is obtained in PIT, the router will insert interestâs incoming face into this matching entry. If not, PIT will create a new entry to identify the interest. Meanwhile, the interest will be forwarded to the outgoing face based on FIB and forwarding strategy. When NDN receives a data packet, if the Dataâs name can be found in PIT, the NDN router will send the Data to all faces that are recorded in the PIT entry and cache a copy in CS. Otherwise, the data packet will be dropped. Data becomes independent from location, application, storage, and means of transportation, enabling in-network caching and replication. The expected benefits of NDN are improved efficiency, better scalability with respect to information/bandwidth demand and better robustness in challenging communication scenarios.

Name-based internet means an obvious opportunity but with some challenges. Then current network TCP/P is predicted to be replaced by ICN/NDN in next generation. However, it is a unrealistic to replace IP routers to ICN/NDN at once. A transition periods and some co-existecne is unavoidable. Achieve NDN extensive usage in the future requires a connecting gateway between the Internet Protocol Version 4 (IPV4) which is widely used in world infrastructure, with a NDN-based network edge. According to the survey, [] ,The proposed approached can be mainly divided into three main categories, stack modification, translation and tunneling. Stack modification is implement dual-protocol-stack (ICN/NDN and TCP/IP) at the same hardware such as routers and switches. Author in [], proposed to use a dual switch which have independent dual channel to forward TCP/IP and NDN. A particular dual-stack router is mandatory

to implement this approach which should both distinguish the IP header and the NDN header. The second main approaches is called protocol Translation. Translation is the mechanism of converting from TCP/IP to ICN and vice versa. It is considered the most economical migration approach to be implemented without losing the merit of ICN/NDN benefits. This type of approach demands a gateway used as an interpreter between the two network protocol, as shown in Figure 6. In OSI model, the translation approaches can be implemented in three levels of translation in network layer, transport layer and application layer.

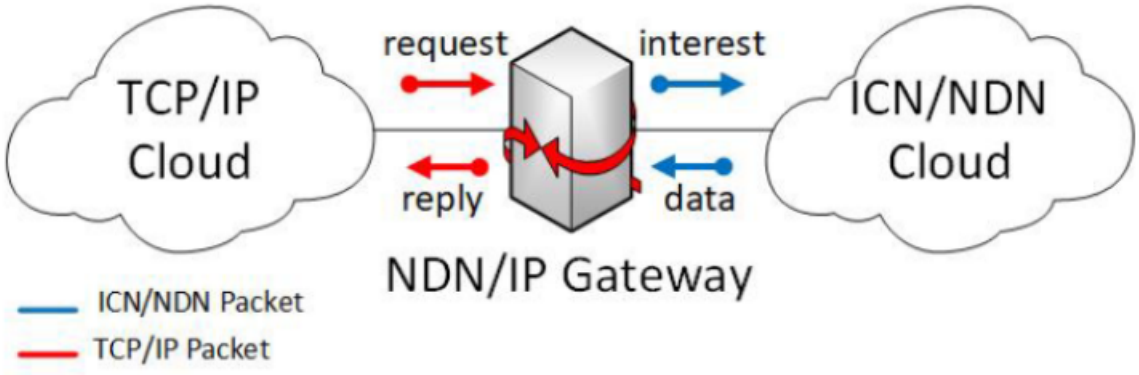


Fig. 6. Translation gateway

This project will mainly focus on the approaches called tunneling, which is the most simplified approach. In computer networks , a tunneling protocol is a communications protocol that allows for the movement of data from one network to another. It involves allowing private network communications to be sent across a public network (such as the Internet) through a process called encapsulation. The tunneling protocol works by using the data portion of a packet (the payload) to carry the packets that actually provide the service. It allows a foreign protocol to run over a network that does not support that particular protocol, such as running IPv6 over IPv4. In designing an NDN-TCP/IP tunneling, the encapsulation uses TCP/IP as the base layer and the NDN as the upper layer. So it doesnât require router upgrade. There are two main different encapsulation strategies: Up-layer encapsulation and underlayer encapsulation. Up-layer encapsulation uses an IP payload to forward the whole NDN packet, such as a virtual network built upon the underlay. It means we build a virtual ndn network upon the physical IP network. The other way is under -layer encapsulation in which it implements NDN as a base layer. This approach is required to add new physical NDN infrastructure which ensures a runnable NDN local network. An underlying deployment approach often involves the introduction of proxies or protocol conversion gateways. Compared with the other two approaches above, tunneling doesnât require changing the existing IP backbone to a large extent(No upgrading demand in upper layer; Gateway and proxies introduced in the Underlay deployment but upgrading in the existing protocols). By considering the feasibilities, this project will only focus on the Tunneling Approaches.

In the paper[], the authors provide a good analysis on the difference between IP network and the NDN network, which is the key challenge to be solved in design. IP network completes packet delivery in two phases. On the routing plane, routers exchange routing information and choose the best routers exchange routing information and choose the best routes to build the forwarding table. At the forwarding plane, routers forward packets strictly in accordance with the forwarding table. Therefore, IP routing is stateful while forwarding is stateless. The robust data transmission depends entirely on the routing system. The routing in an NDN network is similar

to on the IP routing. The NDN routing computes routing tables which are used in forwarding interest packets. However, the forwarding plane of NDN includes two parts: the Consumer first sends interest packets, then data packets are sent back along the reverse route in the reverse direction. Routers save the state of pending Interests to guide Data packets back to the Consumer who required the content before [15]. The forwarding plane of NDN is stateful, which enables each NDN router to measure packet delivery performance and make corresponding adjustments, giving NDN many special advantages including built-in network caching and multicast data delivery. The forwarding plane of NDN is stateful, while the forwarding plane of IP is stateless. The biggest challenge is to complete the conversion between the stateful and the stateless, because the forwarding state in NDN edge network will be lost in IP backbone transmission if there is no processing, which may heavily interfere the benefits of NDN. Therefore, in IP backbone transmission, NDN-IP gateway should preserve the state of NDN protocol. In addition, a conversion overhead is inevitable. The gateway should help to reduce the overhead. in [], it proposed that it is possible to use some important tables and entries in the NDN-IP Gateway to reduce the overhead. In their deployment scenario, they implement pure NDN network, and the edge networks using pure NDN protocol are connected in IP backbone. In comparison, the common overlay approach, NDN edges, can not be network independent of IP, which is the opposite of our intention. It gives a Gateway to assemble NDN edges into IP world which can be used to translate protocols between stateless NDN and stateless IP to coexist in the same network layer. When the protocol is translated, the state and the security of data should be reserved in the forwarding plane.

The aim for this project is to analyze this gateway solution, find the vulnerabilities and the security risks, and find solutions to improve the tunneling mechanism and provide some suggestions to mitigate security risks.

Aim:

- To analyse the Gateway design for tunnelling NDN network edges over an IP backbone, including understanding the motivation, mechanism and evaluating its performance, security and also tradeoff.
- To find the vulnerabilities and security risks and propose solutions to mitigate the security risks.
- To propose a modified design suggestion by combing the survey and experiments

Objectives:

- To collect related works on tunnelling technology of NDN-IP to establish the fundamental knowledge of the latest state-of-art.
- To synthesise different approaches, emphasise the key challenges demanded to solve.
- To construct the evaluation metric and compare an NDN-IP design with the existing IP network to provide an evaluation outcome.
- To establish a simulation or simulation program for testing the feasibility and availability of a real tunnelling process
- To understand the security requirements and model the possible attack aiming to the vulnerabilities in this process.
- To find a security protection add-on mechanism on the original design. To provide an evaluation summary at the end, possibly with a run-able simulation program.

In Section II, there will be a survey on related work and discussion works of the state-of-art, then emphasize that the main approaches will be selected and analyzed. In Section III, highlighted paper that will critical compared to help us identify the main address. A general summary will be given for discussing useful approaches and potential challenges.

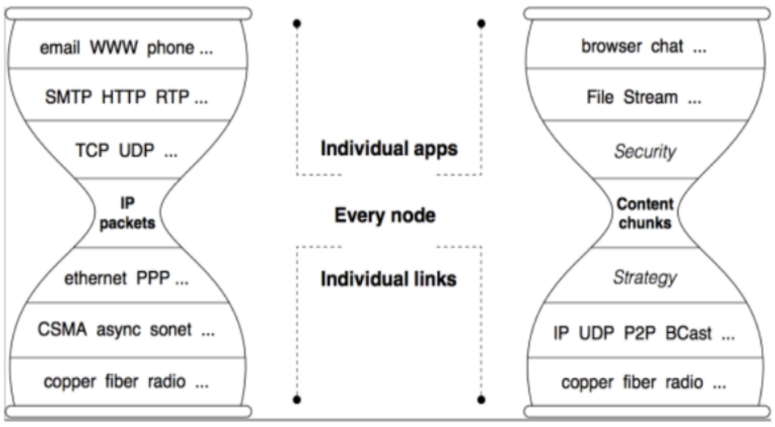
2 Literature Review

The organisation in this section of those materials will follow a general-to-specific order. It will begin with a very general topic including the basic component of a named data network, mechanism of message conversation in named data network and then evaluate the benefits and challenges in the development of such a network. One of the main challenges that should be solved in this project is the coexistence problem between NDN and TCP/IP. Several related works will be summarised and classified to illustrate a big picture of the state of the art in this field. Then, research with the highest relevance to project objectives will be illustrated in detail. In this sub-section, a detailed graph and system behaviour will be presented and evaluated in detail, and used as reference in later research. In the discussion part, the methods mentioned above will be put in comparison. Trade-offs for each approach will be discussed. The experiment tool will be discussed in the last part.

2.1 Overview of Named Data Networking Architecture

The Report NDN-001[1] from Lixia Zhang published in 2010 stands as a project proposal for the Named-data networking project. It proposed project background, basic definition, design and overall architecture. By reviewing this paper, fundamental concepts can be well defined and linked.

Named Data Networking(NDN), it is a new network architecture that has been projected as the future of internet architecture[2]. Unlike a traditional client-server communication model, NDN relies on data as an entity. The Design of NDN is based on the reflection of the strengths and limitations of current Internet practices. TCP/IP was developed in 80s, the problem that TCP/IP solved is a point-to-point conversation between two entities. But the situation in the world has changed, with the rapid growth of e-commerce, digital media and social networking. Internet communication has become increasingly dominated by content distribution. IP can only name communication endpoints. So the content is forced to couple data with a conversation. This IP network design for point-to-point communication is overwhelmingly used for content distribution. One example is that the IP addresses are exponentially reduced in naming space. Named Data network changes the glasshour model, allowing the thin waist to use data names instead of IP addresses for data delivery[2]. For this reason, it is a type of information-centric network in which there is no need for an IP address when delivering messages and packets. In this model, the core role of conversation has been replaced by the data chunk, which is friendly to a network with high demands for content distribution.

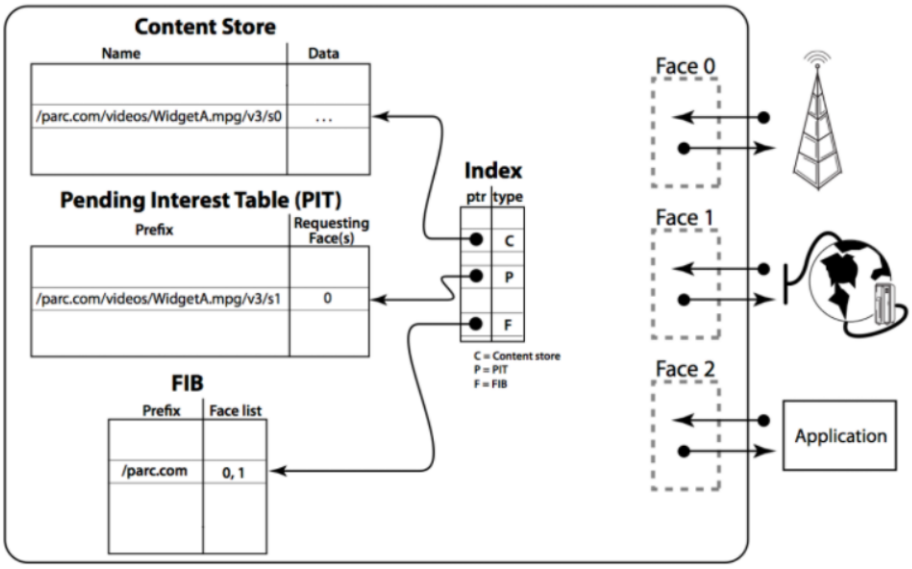


As the project report proposed, NDN uses a âwhat modelâ to directly integrate with a application with network. Current applications are generally written in terms of what information they want rather than where it is located. In the current application, a middleware is used to do this mapping between the location and data. With NDNâs âwhat modelâ, applications can be implemented directly, removing all the middleware and associated configuration and inefficiency.

Another important feature of NDN compared with the current network is end-to-end security. The current security approach is securing the channel between two IP addresses. The main reason is because in the original design of TCP/IP, security problems were being considered. The TCP/IP model has inefficient security features itself. An end-to-end security is hard to ensure even through encrypted channel. While on NDN, all data is secured end-to-end as NDN signs the data with Digital Signature to secure each data's authenticity. In conclusion,NDN has advantages in content distribution, application-friendly communication and native end-to-end data security.

NDN communication uses three entities to deliver the information, namely consumer, producer and router. Communication in NDN is driven by the data-consumer side, or it can be understood as a âfetchâ driven system[3]. It uses routing protocols to reach every Data Name prefixes, instead of relying on source and destination Ip address. Resources are assigned with a unique data name(mostly it is with a hierarchical naming prefix). (Resources include data, including media, files, website files and even the network management configuration).Each NDN node forwards the interest packet based on the specified name, records the interface information from which the Interest Packet is received, and store in Pending Interest Table(PIT). After the Interest Packet finds the appropriate Data Packet, the copy of the Data Packet is also stored in the Nodeâs content Store(CS). If there are identical interest packet requests again, the node will respond with the Data Packet in its content store. That is how caching worked in NDN.

There are three main components used for the NDN routing system. Pending Interest Table(PIT), Forwarding Information Base(FIB) and Content Store(CS).



Pending Interest Table(PIT), where each entry contains the name of the interest and a set of interface from which the matching interests have been received. When the Data packet arrives, the router finds the matching PIT entry and forwards the data to all the interface listed in the

PIT entry. To maximize the usage of the PIT, PIT entries need to be timed out pretty quickly. The PIT state at each router has several functions: . Since it includes the set of interfaces over which interests have arrived, it provides natural support for multicast functionality. .Second, the router can control the rate of incoming data packets by controlling its PIT size. FIB will be a role in the forwarding strategy of NDN.

FIB plays a role in the forwarding and routing process in a NDN system. NDN routers forward packets by looking up a Forwarding Information Base (FIB), each entry of which has a name prefix and a set of output faces. The simplest strategy is to send an Interest to each of the interfaces in a FIB entry in sequence â if there is no response to the Interest, then try the next interface. A more flexible design is for each FIB entry to contain a program specialized in making interest-forwarding decisions. With FIB, routing can be done in a similar fashion to todayâs IP routing. A NDN router will announce name prefixes that cover data that the router is willing to serve. The announcement is propagated through the network via routing protocols, and every router builds its FIB based on a received routing announcement. Traditional routing protocols, such as OSPF and BGP, can be adapted to route on name prefix.

Content Store, where NDN first checks when receiving an interest. If the name in the incoming interest falls in the records of the content store, the data will be sent back as a response. The basic use of content store is just the buffer memory in todayâs router. Both Ip and ndn can buffer data packets. The difference is that IP routers cannot reuse the data after forwarding it, while NDN routers are able to reuse the data since they are identified by a persistent name. Reusable caches provide NDN with Â optimal data delivery. Cache management and replacement are mentioned Â as research topics need to be solved, as claimed by the author in the research agenda.

Today's content distribution network can also solve problems on content distribution. But it is different from the NDN. CDNs are a heavy-weight overlay infrastructure with a large number of servers for caching and serving contracted resources. Because the service is expensive, only contracted applications can be modified to use it. CDNs are isolated from each other, their server coverage and performance depend on their server limitations. NDN is an overlay stack on the same type of IP that is able to be an overlay on top of different transmissions developed today. NDN doesnât require heavy resources, it can run simple packet transmission in best-effort quality. A NDN overly can run over any layer-2 technology or above.

2.2 Coexistence Challenges and tough points analysis

Being a network architecture with a content- oriented feature, NDN is well positioned for the demand and has recently emerged with many advantages in edge networks,â including privacy protection, mobile content access, network traffic balance and adaptive routing [3].Regarding the huge number of uses on IP-based network, and the minimum number of NDN-based network implementation, how to evolve NDN in real world is still a major challenge.

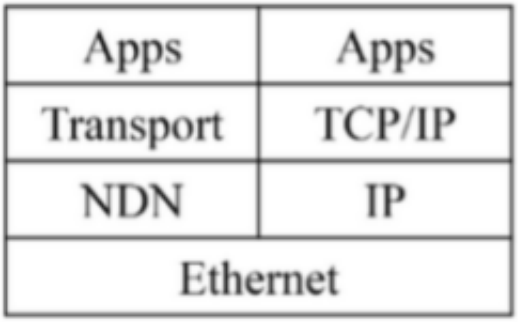
In paper[3], a solution is proposed to use a customised border gateway as a bridge to assemble the NDN edges and IP backbone network. However, NDN is data-centric but IP networks are host-centric. Due to the different design principles, coexistence has become complicated. TCP/IP backbone is an essential role in todayâs global network. NDN is necessary to coexist with TCP/IP. The project proposal report[] claimed major roadblocks to make NDN nodes infeasibly communicate with each other. The most important one is the lack of access to low-level network API. NDN node cannot directly communicate using layer-2 frames (through ethernet) in a NDN-IP mixed network domain. So, NDN nodes must use IP connectivity for interconnectivity. A native TCP/IP backbone network don't support two separate NDN nodes communicating directly in an efficient way, so a specified component is necessary to provide functionality for that.

Most related research choose to design a new form of NDN gateway for connecting two NDN edges and IP backbone, at the same time, as an agent to manage the conversation between two NDN edges. There are two main reasons behind this. First, it is cheaper to upgrade the key components in a network rather than upgrade most related facilities. The second is, NDN give an end-to-end anonymity because it doesnât use IP address. A gateway can be used to weaken the exposure of the IP address under the coexistence concession. To be specific,when an eavesdropper captured the packet at the intermediate of the transmission. If it upack and check the source IP and destination IP, only the IP addresses of the gateways will be exposed. Through this way, the consumer and provider can keep anonymous.

To design such a gateway, it has to address the conflict between two standards . The author of [4] discussed and defines to essential challenges in two main tips. The first is the conversion between stateful and stateless communication. The forwarding plane of NDN is stateful, but that of IP is stateless. The difference may increase the difficulty of protocol conversion. it is a key question to preserve the state of NDN even during a long-stand transmission over TCP/IP backbone network. The second is preserving the unique features as much as possible of NDN in a coexistence solution. The unique advantages of NDN like caching may be interfered in the coexistence scenario.In NDN, duplicate requests for the same data are merged in Pending Interest Table (PIT), which means a returned Data packet can satisfy many requests according to interface ID in PIT. However, for NDN- IP gateway, how to satisfy duplicate requests from the large IP backbone network will be an issue, which may weaken the effect of in-network caching. Thus, the preservation of NDN speciality is critically important for NDN-IP gateway.

In section 2.3, the state of the art in this field will be discussed. There are several migration and transition approaches that have been proposed recently by many researchers. Furthermore, these methods can be categorized into three types. After a summary of the main content of each finding, discussion and some recommendations will be given at the end as a remark for future works. Totally, there are three main directions to design a NDN-IP gateway, stack-modification, translation and tunneling[3].

Stack reprogramming or reframing in the network protocol is one of the candidates for the migration approach. It is generally implemented through modified the network protocol stack on the hardware or extend the existing protocol stack for adding new features. A particular dual-stack router is mandatory to implement this approach. The router must be able to distinguish both the IP header and the NDN header.



(a) The dual protocol stacks of NDN-enabled hosts

Encapsulation(Tunneling) is the simplified approach to migrate from a TCP/IP to an NDN protocol. A tunneling protocol is a communications protocol that allows for the movement of data from one network to another. The tunneling protocol works by using the data portion of a packet (the payload) to carry the packets that actually provide the service. By using a tunneling approach, it allows a foreign protocol to run over a network that does not support that particular protocol, such as running IPv6 over IPv4. An encapsulation method for a NDN-IP-NDN topology is the easiest way to achieve coexistence. Compared with other methods, it modifies fewer on the current network system, therefore fewer problems it will possibly bring.

Translation is the mechanism of converting from TCP/IP to ICN/NDN and vice versa. It is considered as the most economical migration approach to be implemented without losing the merit of the ICN/NDN protocol. Translation approaches intends to perform a conversion between NDN header and TCP header when packets through the border gateway. In this process, it is worth mentioning that most papers design a mapping mechanism from the data name prefix to IP address.

2.3 Sate of Art

2.3.1 On Incremental Deployment of Named Data Networking in Local Area Networks[5]

This essay introduces Dual-Stack switches which can run both NDN protocol stacks and IP protocol stacks. These dual-switches can recognise the identifier in the packet headers. It is a typical stack modification approach. The dual-stack approach means that the router can receive and forward the incoming packet independently for both NDN and IP packets. The authors show an implementation of the dual-stack switches to forward NDN packets in the Local Area Network (LAN). Using the MAC addresses, the authors managed to build the FIB, PIT, and CS tables. By using the dual-stack approach, each packet running in the network uses its header independently. The dual-stack router processes the packets individually regarding their type of NDN or IP, as shown in Figure 2.

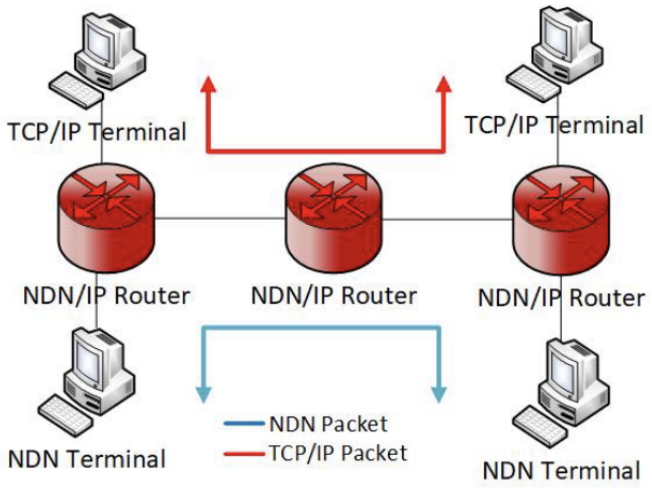


Fig. 2. Dual stack

It is type of stack modification approaches. The good point of this is the dual stack switch will bring no overhead occurred in the packet header. The bad point is that is could be the most expensive approaches because it replaces all the current hardware in the network.

2.3.2 Hybrid information-centric networking[6]

This paper[6] aims to insert ICN into the Internet Protocol. Named Data network is one of the successful instances of an Information-centric Network. This paper discussed a more general

concept. It is proposed to modify the current internet protocol to make it both obtain benefits of IP content networking and Information centric Networking. The author introduces hybrid ICN or hICN. In hICN, the ICN prefix name is forced to be fitted and encoded in the IP address header. A regular IP router can recognize an hICN packet as a standard IP packet in the network. In this way, ICN edges can communicate through an IP network because the packet will be delivered as a regular IP packet. On another hand, the hICN router manages to obtain extra information from the source/destination IP header as ICN prefix name.

Hybird ICN to encodes the prefix name into 128bits of the IPV.6 address. As a supplementary, hICN uses TCP or UDP header as a suffix name. The concatenation of prefix and suffix creates an encoded ICN prefix name in an interest or data packet. The hICN can only use a fixed length of the prefix name instead of a variable length. By putting the ICN/NDN name into IP address, hICN enables regular IP routers to propagate a name-centric packet in its network. Thus, partial router upgrading is sufficient for implementing hICN, which leads to migration cost reduction.

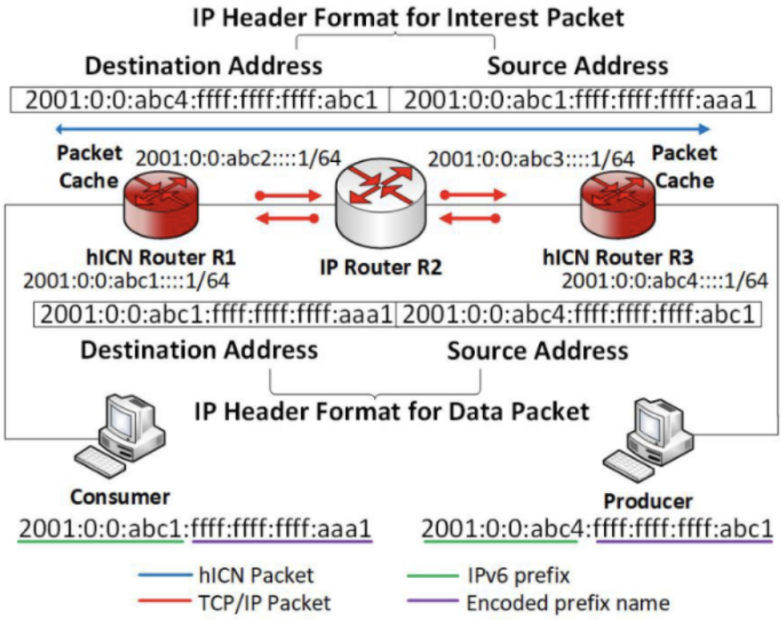


Fig. 3. Hybrid ICN

This paper [6]is another type of implementation of stack modification approach. It modified the packet header to make sure the existing network routers can automatically distinguish and deliver both an ICN packet and a IP packet. In contrast, the dual switch modified the router to distinguish the headers. It is worthy to noticed, this mapping scheme between IPV6/IPV4 and Name prefixed are refer in many paper which mainly use a translation gateway.The hybrid ICN was proposed by Cisco and in the development process. It âmay possibly be another competitor against named data networking in the future.

2.3.3 The Dual-Channel IP-to-NDN Translation Gateway[7]

This essay presents a privacy-preserving translation method between IP and NDN called the dual-channel translation gateway. The gateway provides two different channels dedicated to the interest and the data packet to translate the IP to the NDN protocol and vice versa. A

name resolution table is provided at the gateway that binds an IP packet securely with a prefix name.The main features can be summarised as:

1. packet in the network layer by implementing these channels. We introduce the asymmetric name resolution table to provide a naming service for binding an IP address with a variable length of prefix name set statically in advance.
2. The name resolution table consists of two independent tables, namely the producer table and the consumer table.

The dual-channel translation gateway focuses on building a translation gateway that strict to the driven-pull semantics in the NDN family protocol. The objective is to bring NDN semantics into IP protocol so the translation process can be done effectively. The gateway can recognize the IP packet identified as an interest packet or a data packet in the network layer by implementing these channels.It includes following components:

- The IP data channel also consists of a static IP address devoted to sending or receiving an IP data-like packet between the IP node and gateway.
- The name resolution table gives information about the prefix name associated with the combination of IP address and port number.

The data deliver process example in this design can be described as below two scenarios. As for scenario IP-to-NDN, IP node as a producer and an NDN node as a consumer.

1. the NDN node emits an interest packet with a particular prefix name, /abc, to the gateway.
2. Next, the gateway translates the NDN interest packet to the IP interest packet The by retrieving the associated prefix name in the producer table.
3. The gateway constructs an IP packet with a destination address header of 10.10.10.17:1001 and a source address header of 10.10.10.1:7777.
4. Finally, the gateway sends it through the interest channel.
5. IP node replies directly by sending the content with IP packet through data channel with destination address header 10.10.10.2:9000.

For scenario NDN-to-IP,The IP consumer sends an interest-liked IP packet to NDN producer.

1. The IP node constructs an IP header packet with 10.10.10.1:8000 in the destination part and 10.10.10.17:1002 in the source part with an empty data payload.
2. Receiving the packet, the gateway knows that the packet is an interest packet since it uses IP 10.10.10.1 as the destination address.
3. The gateway converts this packet into an NDN packet with a particular prefix name mapped from the source IP address and port number.
4. The gateway converts this packet into an NDN packet with a particular prefix name mapped from the source IP address and port number.
5. The NDN producer receives the interest packet with prefix name /eee and replies directly by sending the data packet to the gateway.
6. The gateway translates the packet by constructing the IP packet with 10.10.10.17:1002.

In this paper[7], it presents a typical translation methods which allows IP node and TCP node can communicate with each other through a gateway as an Interpreter.

The good point of this is the dual-channel reduces the investment cost network because it only needs to deploy at one router for each network without modifying each endpoint. While the disadvantages are it relies on an unreliable mapping process. The anonymity of NDN will be lost.

2.3.4 Automated Tunneling Over IP LAN: RUN NDN Anywhere[8]

This paper proposed a neighbour discovery mechanism called NDND. In many cases,Â NDN nodes need to communicate over IP connectivity, for example over UDP/IP tunnels. However, establishment of these tunnels requires support for automatic discovery of the NDN nodesâ IP addresses and their data name prefixes. The author describes a rendezvous service for NDN nodes on the same IP subnet to automatically discover each otherâs IP addresses and data name prefixes so that they can establish NDN connectivity among themselves by setting up UDP/IP tunnels.

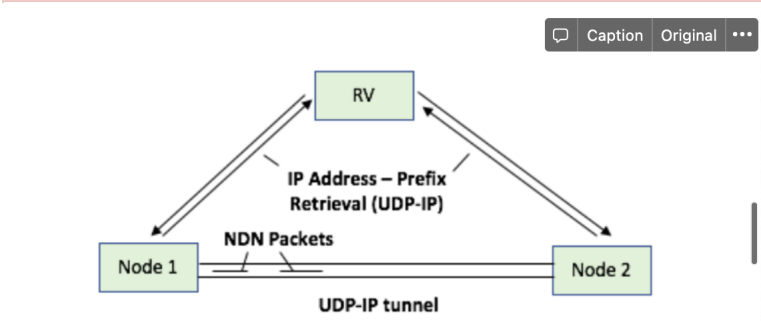


Figure 1: NDN Neighbor Discovery Protocol

The author presents a rendezous service for NDN nodes using the RV () server to automatically discover each otherâs IP address and data neme prefix. The author named it the NDND-NDN Neighbour Discovery. Components of this Neighbour Discovery Protocol include: 1. Rendezvous Server RV; 2. a Neighbour Discovery Application (nd-app) running at each NDN node: app. 3.a Protocol for Neighbor Discovery local nodeâs name prefixes, use the neighbour discovery protocol to communicate with the rendezvous server RV to report its own IP address and the prefixes it serves, and retrieve information about other NDN nodes.

Process:

1. Each application running on aÂ NDN node registers the prefixes with the nd-app by sending a interest with the name "/ndn/servicedsicovery/prefixregistration". For example, a camera app may register the prefix "/username/camera." The nd-app at the node collects all the prefixes in preparation for sending them to RV.

2. Each nd-app configure with an RV name, which can be resolve to an IP address using DNS.

3. Nd-app communicate with RV by using NDND protocol whose packets are carried in UDP messages

4. Node 1's nd-app aggregates its prefixes and IP address into message of type IP-PREFIX-MAPPING, sign with its key, and sends the message to RV.

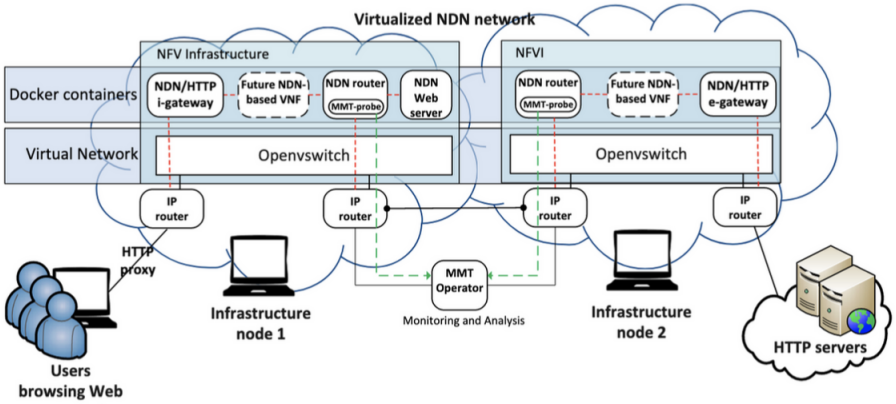
5. When RV receives the Node 1âs "IP-PREFIX-MAPPING" message, it valid and autenticates the message. then add the mapping to its local storage of all IP-Prefix mappings it has received.Alternatively, if an entry for the IP address exist, RV replace the prefix list with the newly received list. Otherwise, RV creates a new mapping.
6. RV the responds to node 1 with a message of type(âIP-PREFIX-MAPPING-LISTâ) which contains all knowing mappings.Alternatively, if Node 1 does not receive a response within some expected time period, it will retransmit the âIP-PREFIX-MAPPINGâ message.
7. To keep itself up-to-date, nd-app sends an âUPDATE-RERQUEST periodicallyâ, in respinse the RV sends its latest âIP-prefix-mapping listâ
8. Once a node receives IP-prefix mappings for other nodes, it can create UDP-IP tunnels to them and store in FIB(Forwarding Information Base)

To evaluate it, It is not necessary for each node to store every IP-prefix entry it received, it only need to store information of interested node, so a memory waste will be brought to the system. Secondly, it will face a single point failure. The RV need to be robust to avoid the whole system crash. A good points include, it is simple and concrete, helpful to be extend to a wider network. It give clue for Node Discovery for a NDN network.

2.3.5 A virtualised and monitored NDN infrastructure featuring a NDN/HTTP gateway[9]

This paper[9] presents two practical components to allow real deployment: a NDN/HTTP gateway that connects a NDN network to the rest of the World Wide Web. The author points out that network operators are reluctant to deploy globally Named Data Networking (NDN) because of the huge investment costs required and the uncertainty about the security and manageability of such disruptive network protocols when deployed in production, while the return on investment is also uncertain. Meanwhile, NFV provides an extremely cost-saving boost in the deployment of new networks. It allows the use of a large amount of business-level hardware to replace the special hardware. With this background, this project aims to move NDN from a lab-limited solution to a real virtual network function. Since current web clients and servers do not yet implement NDN, we have designed and implemented a dedicated gateway to perform NDN/HTTP translations in order to connect an NDN network to the rest of the World Wide Web.

The gateway is built as a Virtual Network Function (VNF) that may be deployed as needed. The ingress gateway (iGW) (income) is in charge of turning entering HTTP requests from any client into NDN packets and then responding to the servers' responses by converting the received NDN data into HTTP responses. The egress gateway (eGW) (outcome) is in charge of turning NDN packets received from the inside into HTTP requests sent to the appropriate websites, and then transforming the HTTP responses back into NDN data packets for forwarding.



Process:

1. First, when receiving a new request, the iGW creates a NDN name suffix from the requested URL that will be used between the two gateways for this data exchange

2. Then, the iGW sends an Interest to the eGW with a name component that announces a new request for this suffix.

3. The eGW responds with a Data packet that carries a status code,Since Interests should not carry data, ,eGW sends in turn an Interest with a name containing the suffix made by the iGW in order to retrieve the full HTTP request.

4. In the last step, the iGW sends Interests to get the HTTP data retrieved by the eGW on the Web. If the content is already present in caches, Interests are directly answered from the NDN network, following NDN principles, without soliciting the eGW.

This work is noteworthy because it combines NDN and an NFV. In the other part of the research, they also proposed using a virtualized monitor to analyse the state of the whole network. It serves as an example by combining network virtualization and the NDN communication standard. The benefits of this project, as they said in their discussion part, are that it allows a cheap deployment of a new form of network standard by using network virtulisation techniques. This is a common misunderstanding.

2.3.6 iGate: a practical and highly-efficient solution to assemble NDN edges into IP world[4]

In this paper[4], the author proposed an NDN-IP gateway designâiGate, bridging two protocols through a tunnel over the IP world, is proposed to tackle the challenges caused by semantic differences between NDN and IP packets. Moreover, iGate can cooperate with PIT and FIB in NDN to gain less conversion delay.

NDN-IP gateway may offset the unique advan- tages provided by NDN like built-in network caching.Additionally, NDN-IP gateway will inevitably bring extra overhead to the system, which may affect the transmission. This research is a approaches belong to tunnelling classification. To be specific it is an underlay encapsulation approaches. That is to say, it reserve the pure NDN topology instead of implement the NDN feature over the IP protocol stacks. The benefits of it are reserving the nature of NDN in a large extent, a more clear and concise design and less conversion delay in the border gateway. if underlay approach is adopted, IP backbone can not get any information about NDN edges, which motivates us to develop iGate to assemble NDN edges into IP world.

The NDN-IP gateway may offset the unique advantages provided by NDN, like built-in network caching. Additionally, the NDN-IP gateway will inevitably add extra overhead to the system, which may affect the transmission. This research is an approach that belongs to the tunnelling classification. To be specific, it is an underlay encapsulation approach. That is to say, it reserves the pure NDN topology instead of implementing the NDN feature over the IP protocol stacks. The benefits of it are that it preserves the nature of NDN to a large extent, it is more clear and concise and there is less conversion delay in the border gateway. If the underlay approach is adopted, the IP backbone cannot get any information about NDN edges, which motivates us to develop iGate to assemble NDN edges into the IP world.

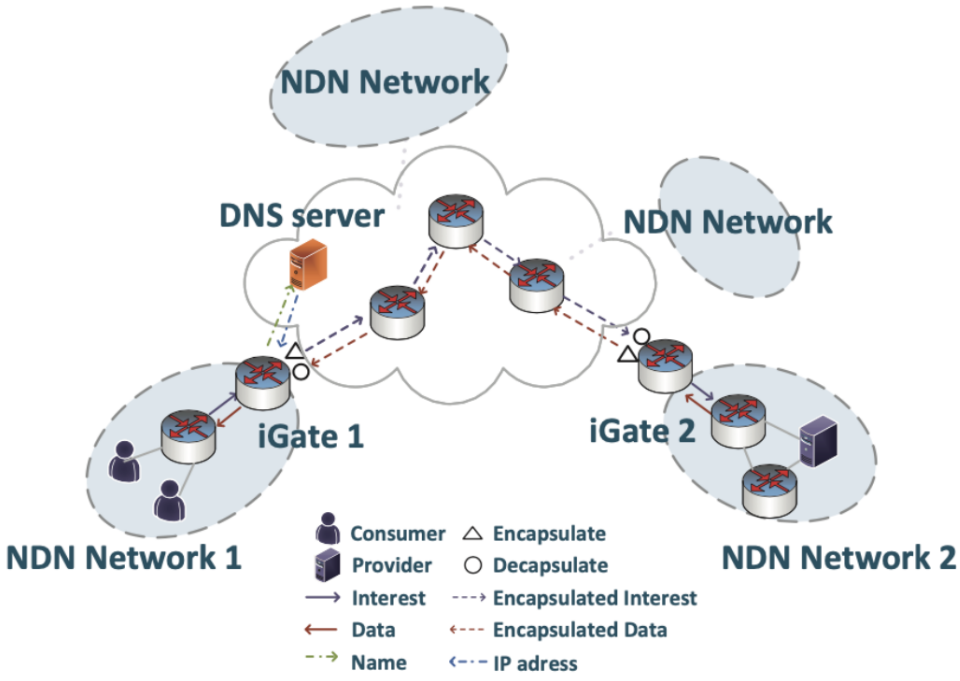


Fig. 2: NDN-IP Interconnection Scenario

It includes the following main components:
Gateway Translation Table (GTT): When receiving the interest packet, GTT complete the conversion between the requested data name and IP addresses. GTT change the content-oriented packet into host-oriented IP packet.

Tunnel State Table(TST): when iGate2 receives the Data Packet returns from NDN, TST will match the corresponding Interest request. TST actually complete the state switch between two kinds of packets in NDN.

There are 2 conversions that will happen.

(1) Make the transition from stateful NDN to stateless IP. iGate 1 receives the interest packet from the consumer, and iGate 2 receives the data packet returned by the provider. When converting from name-identify to IP-identify, the IP address of both iGates needs to be determined.

(2) Make the transition from a stateless IP to a stateful NDN. (It is more complicated to cooperate with PIT to maintain stateful forwarding of NDN). The data packet flows back along the same path in the reserve direction where the interest packet is forwarded. We mainly use the incoming interface ID, which is logically defined in PIT, to maintain the state of pending interests. Therefore, two tables are designed as follows to translate protocols between the stateful NDN and the stateless IP in iGate.

When receiving an interest packet, GTT will complete the conversion between the requested data name and IP address. GTT actually changes content-oriented NDN packets to host-oriented IP packets. For instance, iGate 1 receives an interest packet whose name field is written in "bit/file.txt", showing that the consumer asked for a txt document, and iGate 1âs job is to forward the interest packet. Because iGate needs to send the packet to another NDN node across the IP network, iGate should know the corresponding IP address for encapsulat-

ing interest packets, which will travel through the IP network. For instance, iGate 1 receives an interest packet whose name field is written in "bit/file.txt", showing that the consumer asked for a txt document, and iGate 1âs job is to forward the interest packet. Because iGate needs to send the packet to another NDN node across the IP network, iGate should know the corresponding IP address for encapsulating interest packets, which will travel through the IP network.

As a result, GTT is in charge of generating a name-to-IP address mapping. Due to the hierarchical nature of NDN naming, every NDN edge should have the same name prefix, providing further GTT support. Between the name and IP address, GTT employs the longest prefix matching technique. If iGate 1 identifies the longest prefix that matches the IP address according to GTT, the IP address acquired will be the IP address of iGate2, which is the border gateway in the NDN edge that contains the Provider.

This study currently uses a configuration file to set up GTT. Efforts are now being made to automatically establish GTT, which might be achieved using the Domain Name System (DNS). GTT created automatically may follow a similar setup method to DNS.
In "Automated Tunneling Over IP Lan: Run NDN Anywhere," GTT is described as being quite comparable to an RV server. This Discovery Protocol can be used to fill in the gaps in GTT's entries. has a similar setup procedure as DNS.

Tunnel State Table(TST):
TST was created for the purpose that ensuring the maintenance of NDN State is an issue that iGate should prioritise. TST keeps track of a tunnel's quintuple information, as well as two types of sockets. State Keywords, as well as a TimeStamp TST's goal is to discover the relevant Interest request when iGate2 receives a Data Packet (), given that iGate may receive a large number of Interest packets from NDN edges. TST uses two types of sockets to keep track of this correspondence.
One socket in TST is TCP-socket, which stores the socket ID that is created by the connecting or accepting operation. Hence, TCP-socket actually keeps those tunnels beginning at iGate 1 or ending at iGate 2. The other socket is NDN-socket, which is created when iGate 2 receives an encapsulated interest packet from a new tunnel. That is, NDN-socket has a one-to-one relationship with the tunnel established.

TST items have "state" added to them to represent the tunnel state. We use colour to denote the status of a tunnel, and some more components are included, as inspired by Yi et alsimple .'s colour system. The planned state data is depicted in Fig. 5.

The author also provides a Condition table to show the tunnel's current state. The second state of a tunnel is CLOSED when it is added to TST, and it changes to CREATE when the first encapsulated Interest packet arrives at iGate 2. When the first encapsulated Data packet returns to iGate 1, the CREATE tunnel becomes ESTABLISHED. The tunnel state can be changed to GREEN or YELLOW by choosing an acceptable time period. When there is no transmission across the tunnel for an extended period of time, the tunnel state is turned to RED and the tunnel is erased from TST, indicating that the tunnel has been lost.

State	Description
GREEN	the tunnel can transmit data at a reasonable time.
YELLOW	the tunnel transmits data beyond a reasonable time or tunnel transmit Interest NACK.
RED	for a long time, there is no transmission through the tunnel.
CREATE	iGate 2 receives the first encapsulated Interest packet.
ESTABLISHED	iGate 1 receives the first encapsulated Data packet.
CLOSED	an initial state.

Fig. 5: The State items

This research designs two separate mechanisms for TCP and UDP. Under different application requirements, iGate needs to choose a suitable transport layer protocol, so iGate is designed to support the TCP protocol and the UDP protocol. TCP is a connection-oriented protocol, while UDP is a connectionless protocol. The realisation of the two kinds of systems is quite different. If NDN edges deployed on the edge of the IP backbone want to communicate with each other in TCP, a connecting operation will be needed first. iGate should maintain the state of each tunnel and enable the delivery and reception correctly. This is a common misunderstanding.

Then there are encapsulation process suggestions. A packet header including a unique tag and a length field should be used to encapsulate an NDN packet. The NDN packet is identified by the unique tag "!NDNpkt," and the length field displays the packet length so that we may analyse the received data exactly. The information can be organised in the structure depicted in Fig.6. When iGate receives a TCP packet, it compares the first 8 bits of the payload to the tag to determine whether it is an NDN packet. If it isn't, iGate will discard it because it only handles NDN packets. If affirmative, iGate will look for the length field to determine the size of the NDN packet and then receive the packets until all are received.

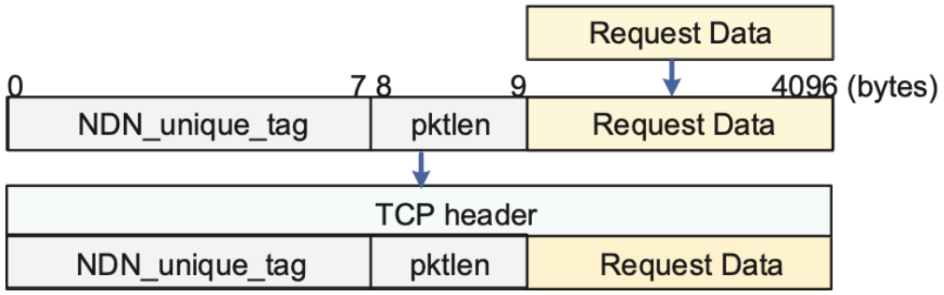


Fig. 6: Self-defined communication rule

Following this research, a use case may be derived to demonstrate how the system functions. This review will demonstrate the data flow along the lifecycle of a data packet, which is slightly different from the data flow in the paper. Assume that each NDN node's Forwarding Interface Base (FIBs) stores the default path to the Gateway in case the incoming Interest Packet doesn't have a mapping prefix name. The following is a description of the process scenario:

1. .Laptop request for a server.log.It generates an Interest Packet. The Content Name is âfile/html/a.htmlâ. The Selector and Nonce is set as default.
2. Forward the interest packet in the edge 1. If the resources can be found in edge 1,

the data packet will be sent back along the reverse path where the interest packet is sent. Alternatively, if the data cannot be found in edge 1, according to assumption 1, the interest data will be sent to the Gateway A, then go to step 3.

3. The Gateway A looks up its GTT, use the âlogging/server.logâ to map the Ip address. If a not exist in the GTT, an resource discovery mechanism should happen, consider it further. Alternatively, if the mapping can be found, (obtain the destination Ip address , for example â192.168.66.135â)

4. The Gateway A looks up its GTT, use the âlogging/server.logâ to map the Ip address. If a not exist in the GTT, an resource discovery mechanism should happen, consider it further. Alternatively, if the mapping can be found, (obtain the destination Ip address , for example â192.168.66.135â)

5. Check the TST and forward. If this is a new tunnel, create for a new tcp sock â3â and stored in TST, as the TST follow. Store the incoming NDN interface, here is â4â.

6. Perform the connect operation with Gateway A and Gateway B, send encapsulated Interest

7. Gateway B listens at designated port and accept via ipsock. The Gateway B receive encapsulated interest packet via ipsock. It apply for a new temp NDN socket for receiving returned data, here â5â. TST State will be updated update to Created.

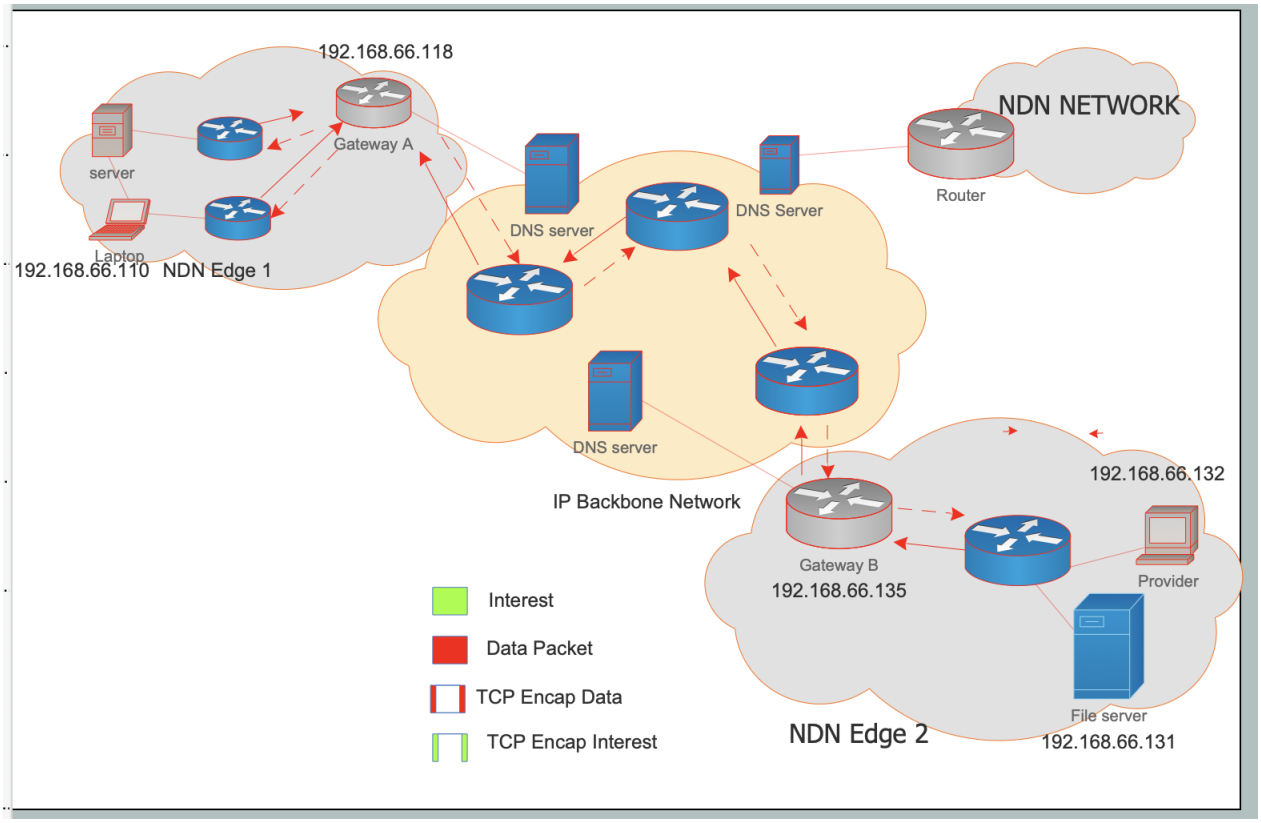
8. The Gateway B decapsulate the packet and forward it to the NDN edge through ndn socket 5. It listens at the socket 5 and waiting for the returned Data Packet until receives the returned data packet.

9. Gateway B received the Data Packet from the edge 2, it will obtain the TCP-socket from the TST through the linked incoming ndn socket ID 5. And then it perform encapsulation.

10. The Gateway B forward the TCP datagram through the TCP-socket 3.

11. Gateway A obtains encapsulated Data packet by monitoring TCP-sockets 3 in TST and decapsulates it,

12. The Gateway A listen at the ip socket 3 until receive the UDP datagram. It updates the TST table state into ESTABLISHED. Then it decapsulates the UDP packet and forward the unpacked Data Packet to the NDN edge 1(consumer side) through ndn Socket 4. The NDN nodes will forward the packet in a reverse route back to the originality under the guide of PIT and FIB.



To evaluate this research, it belongs to an environment type. It gives a practical implementation of the gateway with fine-grained details. There are several components that are applied to support tunnelling in an artful way. That is worthy to use as a development basement.

Obviously, this is an incomplete mechanism. For example, there is a lack of an automatic GTT. Also, sychronization problems on the TST table should be considered in future work. But it surely has advantages. First, it provides TCP and UDP alternatives through which it can be suitable for different network situations. Secondly, it holds most of the nature of NDN, especially the anomity. Eavesdroppers can only know the IP addresses of two NDN edges, but the exact IP addresses of the consumer or provider are unknown. Finally, this design is lightweight with less protocol conversion delay and less memory usage.

3 Summary and Conclusions

To sum up, there are several co-existence strategies in this field. In Section II, there are mentioned as three main types: Stack modification, Encapsulation and Translation.

Stack modification approaches intent to enable both network protocol IP and NDN in each router. That is to say, all the routers must be a dual stack router. Obviously, one of the simplest ways is to upgrade the hardware as mentioned in [], Â it introduces Dual-Stack switches that deploy NDN stack and TCP/IP stack protocol over the Ethernet. Due to its technical features, it will bring no overhead cost and traffic loads, and meanwhile, reserve Â benefits of a NDN network. But it is not realistic and only feasible on a small local network. Upgrading all hardware is, no doubt,Â to be extremely expensive. In contrast, hICN presented by Cisco modifies the stack at a software level. A hICN node can map an ndn name prefix to a IPV6 standard address, which enables a node to be a dual router. HICN can be implemented by a stack of protocol up-layer on the current network protocol stacks without a hardware upgrade. The main disadvantage of it is that it will lose some benefits of NDN.

Translation approaches prevent the mixed node being introduced in modifications compared with stack modifications. The pure NDN domain and Ip/TCP domain will be connected through some dorder routers. Protocol translation will happen at the border. The great point is the translation approach shows full benefits from ICN protocol features such as caching. Because only the border router needs to be updated, there will be fewer upgrading costs. The disadvantage is that it has to suffer from a conversion delay.

Another best migrating is using encapsulation where IP protocol used as the base layer and ICN.NDN used as the upper later despite the utlity of caching is underperformed. But by synthesizing the technical cost and economical cost, it should be the easiest and cheapest deployment in the transition period.

In this project, an encapsulation method should be the most efficient way to reseach along. In section II, two emphasized research: iGate and NDND have been given a clear basement. By referring to these two essays, the basic behavior of packet delivery through a NDN-IP-NDN topology can be described. There are obvious challenges to this basic behavior. From the usability side, a resolution service is necessary for establishing routing tables for new added edges, the synchronization problems behind synchronization routing information and also the privacy of drop and updates for the routing records. On the security side, DoS and Spofying will easily influence this system.

My project will begin with this basic design and make contributions to its completeness. The basic design only focuses on its inner components alone. But in the current network, some existing protocols must be reusable in this system. My project will aim to give suggestions on the competence of this system and also link it to existing network protocols in the real world. For a feasibility test, an ndnSim program will be developed on ns-3. The expected outcomes include a topology diagram, key component design, a behavior description, runable simulation and optimization analysis with graphs and reasonable metrics.

References

[1] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. Thorton, D. Smetters, B. Zhang, G. Tsudik, K. Claffy, D. Krioukov *et al.*, "Named data networking tech report 001," *the NDN project team, Technical Report NDN-0001*, 2010.

[2] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, D. K. Smetters, B. Zhang, G. Tsudik, D. Massey, C. Papadopoulos *et al.*, "Named data networking (ndn) project," *Relatório Técnico NDN-0001, Xerox Palo Alto Research Center-PARC*, vol. 157, p. 158, 2010.

[3] F. Fahrianto and N. Kamiyama, "Comparison of migration approaches of icn/ndn on ip networks," in *2020 Fifth International Conference on Informatics and Computing (ICIC)*. IEEE, 2020, pp. 1–7.

[4] R. Zhu, T. Li, and T. Song, "igate: Ndn gateway for tunneling over ip world," in *2021 International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2021, pp. 1–9.

[5] H. Wu, J. Shi, Y. Wang, Y. Wang, G. Zhang, Y. Wang, B. Liu, and B. Zhang, "On incremental deployment of named data networking in local area networks," in *2017 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)*. IEEE, 2017, pp. 82–94.

[6] L. Muscariello, G. Carofiglio, J. Auge, and M. Papalini, “Hybrid information-centric networking,” *Internet Engineering Task Force, Internet-Draft draft-muscariello-intarea-hicn-00*, 2018.

[7] F. Fahrianto and N. Kamiyama, “The dual-channel ip-to-ndn translation gateway,” in *2021 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*. IEEE, 2021, pp. 1–2.

[8] A. Padmanabhan, L. Wang, and L. Zhang, “Automated tunneling over ip land: run ndn anywhere,” in *Proceedings of the 5th ACM Conference on Information-Centric Networking*, 2018, pp. 188–189.

[9] X. Marchal, M. E. Aoun, B. Mathieu, W. Mallouli, T. Cholez, G. Doyen, P. Truong, A. Ploix, and E. M. De Oca, “A virtualized and monitored ndn infrastructure featuring a ndn/http gateway,” in *Proceedings of the 3rd ACM Conference on Information-Centric Networking*, 2016, pp. 225–226.

[10] H. Ben Abraham, A. Afanasyev, Y. Yu, L. Zhang, S. DiBenedetto, J. Thompson, and J. Burke, “Tutorial: Security and synchronization in named data networking (ndn),” in *Proceedings of the 2nd ACM Conference on Information-Centric Networking*, 2015, pp. 3–6.

[11] S. Luo, S. Zhong, and K. Lei, “Ip/ndn: A multi-level translation and migration mechanism,” in *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*. IEEE, 2018, pp. 1–5.

[12] A. Afanasyev, I. Moiseenko, L. Zhang *et al.*, “ndnsim: Ndn simulator for ns-3,” 2012.

[13] D. A. Arji, F. B. Rukmana, and R. F. Sari, “A design of digital signature mechanism in ndn-ip gateway,” in *2019 International Conference on Information and Communications Technology (ICOIACT)*. IEEE, 2019, pp. 255–260.

4 Appendix: MSc Interim Project Plan

4.1 Background

In our current network, the IP based technologies are coming in all field, even the voice service is also changing to ip based network. TCP/IP was developed as a groundbreaking solution which solved a problem on point-to-point conversation(of telephones) in the 1970s. By the support of TCP/IP, the IP network has been scaled as a worldwide infrastructure. Today, the situation of the world changed. When exabytes of new contents are produced periodically, content and connected devices exponentially increased, IP which is designed for conversation between communication endpoints is overwhelmingly used for content distribution(IP can only name communication endpoints). NDN was brought forward by Lixia Zhang with her teams in [1] which is one of the most successful instance of the Information Centric Networking(ICN/CCN). Conceptually, the Named Data Networking (NDN) project proposed an evolution of the IP architecture that generalises the role of this thin waist, such that packets can name objects other than communication endpoints[1].An illustration of NDN is that it names data directly use same data unit at both application layer and network layer and it secure data, not channels so data chunks can be decoupled from conversation. NDN is a revolutionary solution for the current requirements with advantages including privacy and security, mobile content access, network traffic balance and adaptive routing. However, the global TCP/IP is dominating, there will be a challenge in coexistence between NDN edges and IP backbone network for global transmission because IP uses stateless forward in whereas NDN uses stateful forwarding. Three classification migration approaches including stack modification, tunnelling(encapsulation) and translation summarised and compared in [4]. Compared with the other two, tunnelling is the most easier and safe one with no violence to current network deployment and relatively less migration cost. A tunnelling protocol allow a foreign protocol to run over a network that does not support that particular protocol, such as running IPv6 over IPv4. In [5],a solution named iGate designs a gateway to tunnel NDN over the IP world. In this project we will focus the design of such a gateway and the security of the protocol translation.

4.2 Aims and Objectives

Aim:

- To analyse the Gateway design for tunnelling NDN network over a IP backbone including understanding the motivation, mechanism and evaluating its performance, security and also tradeoff.
- To find the vulnerabilities and security risks and propose solution to mitigate the security risks.
- To propose a modified design suggestions by combing the survey and experiments.

Objectives:

- To collect related works on encapsulation technology of NDN-IP for establishing the fundamental knowledge of the latest state-of-art.
- To synthesise different approaches emphasise the key challenges demanded to solve.
- To construct the evaluation metric and compare NDN-IP design with existing IP network for providing an evaluation outcome.
- To establish a simulation or real test program for testing feasibility and availability real tunnelling process.

- To understand the security requirements and model the possible attack aiming to the vulnerabilities in this process.
- To find a security protection add-on mechanism on the original design. To provide an evaluation summary at the end, possibly with a run-able simulation program.

4.3 Summary of Related Works

Current Internet architecture based on TCP/IP was evolved since the 1960s which has connect worldwide computers. With increasing numbers of computer and devices connected, the TCP/IP has been developed form IP version 4 to IP version 6 solving a lot the lack of IP address. The TCP/IP is a host-centric networking with sending and receiving packet to communicate. In a packet frame, source and destination addresses are used as two endpoints location for establish a connection between hots. The packets are forwarded through the Internet Service Provider based on the destination IP address of the receiving end until being delivered to the destination host.[2] [10]

In the report Named Data Networking project[1], the author propose the information-centric networking(ICN), and a powerful branch named data networking(NDN) as a revolutionary internet architecture in 2010 to replace the host centric networking, such as TCP/IP. The ICN/NDN took experience of TCP/IP to refactor a new model with upgraded nature in security and capability of cache in each router. The Information centric networking preserve the hourglass architecture but with replacing âthin waistâ from IP to Data Chunk Name which allows a universal network later implementing the minimal functionality necessary for global scalable[1], namely, allowing lower and upper layer technologies to innovate without unnecessary constraints. As Lixia Zhang mentioned in their proposal[1], two types of packet primitives are introduced namely interest and data packets. In order to make data transactions, the node (data consumer) emit an interest packet in advance with a prefix name to retrieve the content packet. Every node (data producer) that has the content with a designated prefix name will reply to the interest and send the data to the requesting consumer in a reversed route as how the interest packet forwarded. NDN routers have three fundamental components to forwarding and routing[1] which ensure the nature on caching and security. The Content Store(CS) is used to store and cache the data of incoming data packet. The Pending Interest Table(PIT) is used to map between a prefix names of missed interests and corresponding incoming interface if the interest prefix name is not cached in the CS. Forwarding Information Base(FIB) is used to forward the missed interest packet to other interfaces listed in the table. Due to the obvious incompatibility between the NDN and TCP/IP network, an transition and migration are inevitable which is also presented as a main challenge in the very first project proposal[1].

The survey 'Comparison of Migration Approaches of ICN/NDN'[3] summarised updated state-of-arts for the co-existence approaches. Conceptually, they are divided into three main direction: stack modification, encapsulation(tunnelling) and translation. Stack modification is one of the candidates of migration approach by reprogramming and reframing in the protocol stacks. One of clear design example[5] published in 2017 proposed to apply a dual protocol stacks with two co-existence protocol stack for NDN and TCP/IP over an Ethernet layer. Even though it is ideal for preserving complete advantages of NDN networking, there would be the most expensive migration cost behind it to upgrading all uncountable routers in the network. Another approach for implementation is recommended by Cisco, which is named hybird Information Centric Networking, namely, hICN[6]. In hICH, the ICN prefix name is encoded in the IP address header. regular IP router can process an hICN packet as IP packet in the network. Meanwhile, an ICN node will have to access an extra IP address information which have violence on the ICN design principle then negatively influence the security nature of ICN/NDN. From a more economical perspective, some teams suggested a different tact. In [11] ,the author

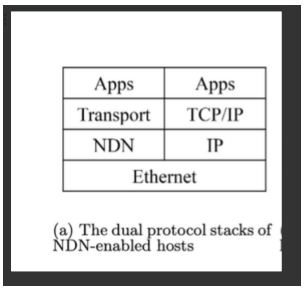


Figure 1: Dual protocol stack over Ethernet

suggested a multi-level translation mechanism. They present to use a TUN device to capture IP datagrams from the senders and apply a translation program to convert IP datagrams to NDN Interest or Data packets by add-on naming scheme, vice versa. The translation device will be responsible construct prefix name for a pure IP datagram and map a destination Ip address on an received NDN packet. The same principle but a branched approach, Fahrianto and their team introduced a general purpose IP-to-NDN gateway with dual channel[7] in 2021. In their research, the gateway they suggested provides two different channels dedicated to the interest and the data packet to translate the IP to the NDN protocol and vice versa. In their gateway, a resolution table consisting producer table and consumer table are used for binding an IP packet securely with a prefix name. For example, with the resolution table, and NDN interest packet with a prefix name can be translate to a IP interest packet by retrieving the associated prefix name in producer table. Modifying a Gateway is an advisable approach because it only needs to deploy at one router for each network without modifying each endpoints[10]. But in their performance evaluation, the results shows the encapsulation is slightly outperformed than dual-channel gateway(translation approach) in throughout and latency.

A solution named iGate[4] designs a gateway to tunnel NDN over the IP backbones using an encapsulation approach in Gateway. Moreover, it also designed to cooperate with PIT and FIB in NDN to gain less conversion delay. It is an underlay encapsulation approach, by this kind of deployment, IP backbone cannot get any information about NDN edges, compared with hICN[8], it preserved the nature of NDN to some extend. Main Components in iGate includes a Gateway Translation Table(GTT) which store a mapping list between Name Prefix and Ip address, a Tunnel State Table(TST), which match data packet corresponding interest request. One of the an artistry of this design is that it apply some concept of a Bind system such as how socket working in current TCP/IP networks. The incoming interface ID [6] can be implemented as a NDN Socket which make it possible to highly preserve NDN behaviour as the original design. A TST table realising the preservation of NDN states which helps Data packet flow back along the same path in the direction opposite to the flow direction. and listen the state of tunnel for flow control. The solution provides a TCP/UDP conversion choice and give a basic security consideration because itâs a underlay deployment with only the Ip address of the two iGate are known rather than the specific Ip address of consumer and provider. But for the disadvantage, the author presented in their paper[5], a DoS attack may happen because the NDN domain is highly relying on the Gateway, for example, when millions of fake interest packet attack the iGate, a crash may happen. Also, the GTT in their very starting design is manual configured which is expected to be assigned automatically by a DNS-like service.

4.3.1 Methodology

In this project, we are going focus on the tunnelling technology of NDN-IP. Our goal is to analyse the design of such a gateway and security of the the protocol translation. In detail, the

authenticity an integrity of data must be be intact. We are requested to analyse the gateway solution, find vulnerabilities and security risks, and propose solution to mitigate the security risks. The tool will be used in this project is python and NS3 simulator whichls.

Due to previous work has been highly collected and organised in both the NDN research community and the NDN official organisations, volume of research results, possible design and runnable programs are accessible in some official sites. It is good for us to utilise the existing library, for example, ndnSIM is developed as a NDN simulator for NS-3[12]. So a technical choice is necessary in our progress. For another hand, basic knowledge of the theoretical preparation is also needed. By widely reading the research articles, we hope to establish a basic knowledge base about the architecture choice, design principle and suggested solutions for the future revise. Then a experiments environments should be established for verification and justify our finding, for this, and ndnSim may be used as a simulation environment and also the official recommended environment named NDNTestbed should be worthy to research.

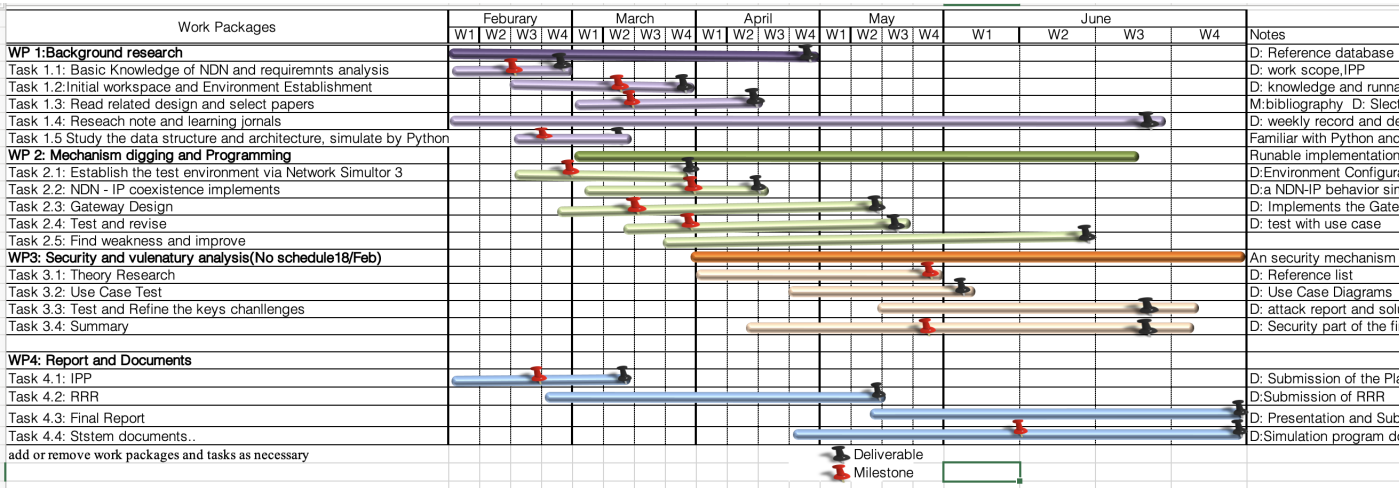
The first steps for initial period is to reading relative papers and prepare for the later test. For the first several week, it is needed to established a list for the gateway architecture ,the mechanism including conversion, forwarding and routing under NDN behaviour, and build the basic knowledgebase to manage the reference list. Then we need a technical choice, with the mature NDN open source community, there should be a trade-off selection on the possible test environment and simulator. Otherwise, topics about security and privacy should be covered for a further evaluating the security problems happened under the behaviour of NDN and also the vulnerable point existing after the integration between IP/TCP and NDN. Theoretical knowledge like new evaluation metric on NDN networking is suggested by the project proposer. The research on evaluation should be aslo including in the preparation period.

Before we get our hands dirty, we should first build up a runnable tools environment, An initial NDN simulation environment should be established which should be able to simulate the co-existence of IP networking and NDN networking including simulating necessary use case and protocols in a network.

When we finished collect the related research evidence and building on the test environments. Two or three associated solution should selected and being focused and getting deep in. A clear approach should start from those selected material. A simulation of the suggested solution should be simulated in our simulator for further analyse. By applying our evaluation metric and test with proper use case, the data (metric) should be captured and classified by using some analysis methods. Python can empower data analysis by library like SciPy, NumPy Sk-learn and Pandas. A ideal outputs for this period should be a runnable network model and data analysis outcomes.

With outcomes above, we need find the weakness of existing solution and improve it by add-on mechanism. In this period, some analogy design of tunnelling protocol may be referred, such as VPN(IP-in-IP), SSH(TCP-IN-TCP) or SSTP(TCP port 443). Exception improve the mechanism, from the security points, we plan to model some possible attack and give an algorithm or mechanism to solve it. A related work is [13] which indicate usage of the digital signature of NDN in security protection.
By testing our runnable simulation program and keep improve it, we can the conduct a summary of the new Design and security suggestion which may be contributive to the progress of this field.

4.4 Project Workplan



4.5 Risk Analysis and Mitigation

In this section, the potential risk and the alternative path will be given in a listing table.

Table 1: Risk analysis and mitigation plans.

No.	Description	WP	Impact Level	Mitigation Measures
1	Project Overdue	WP1/2/3/4	High	Arrange weekly workload
2	Test Environment	WP2/3	Medium	Establish the NS3 envirnment
3	Lack of basic material	WP1/4	Medium	UoB Library plus google

4.6 Resource Requirements

Table 2: Resource list.

No.	Resource	WP
1	NS3 Simulator	WP2/3
2	Python	WP2/3
3	NS3 NDN tookit	WP2/3