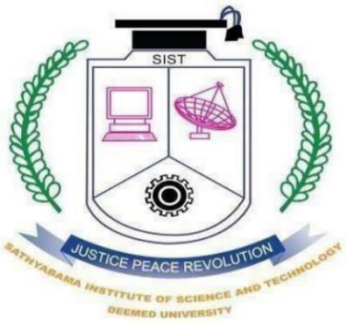
PRIVACY PRESERVING DATA AGGREGATION IN WIRELESS SENSOR NETWORK

Submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering

By

P ARUN KUMAR

(39110768)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHISALAI,

CHENNAI - 600119

NOVEMBER, 2022

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR,RAJIV GANDHI SALAI, CHENNAI - 600 119
www.sathyabama.ac.in

SCHOOL OF COMPUTING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the BONAFIDE work of **P ARUN KUMAR (39110768)** who carried out the Project Phase-1 entitled **"PRIVACY PRESERVING DATA AGGREGATION IN WIRELESS SENSOR NETWORK "** under my supervision from November 2022 to April 2023.

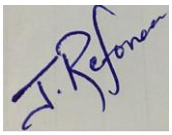
Internal Guide

GRACELIN SHEENA B

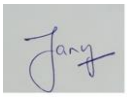
Head of the Department

Dr. L. LAKSHMANAN, M.E., PH.D.

Submitted for Viva voce Examination held on _20.4.2023



Internal Examiner



External Examiner

DECLARATION

I, **P ARUN KUMAR(39110768),** hereby declare that the Project Phase-1 Report entitled **" PRIVACY PRESERVING DATA AGGREGATION IN WIRELESS SENSOR NETWORK "** done by me under the guidance of **GRACELIN SHEENA B** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE: 20.4.2023

PLACE: CHENNAI **SIGNATURE OF THE CANDIDATE**

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.SASIKALA M.E., PH. D**, **Dean**, School of Computing, **Dr. L. LAKSHMANAN M.E., PH.D.,** Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **GRACELIN SHEENA B ,** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-1 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Privacy preservation is an important issue in today's context of extreme penetration of Internet and mobile technologies. It is more important in the case of Wireless Sensor Networks (WSN) where collected data often requires in-network processing and collaborative computing. Researches in this area are mostly concentrated in applying data mining techniques to preserve the privacy content of the data. These techniques are mostly computationally expensive and not suitable for resource limited WSN nodes. In this paper, a scheme is developed to provide privacy preservation in a much simpler way with the help of a secure key management scheme and randomized data perturbation technique. We consider a scenario in which two or more parties owning confidential data need to share only for aggregation purpose to a third party, without revealing the content of the data. Through simulation results the efficacy of our scheme is shown by comparing the results with one of the established schemes.Wireless sensor network produces a huge amount of applications specific data.These data need to be processed and transmitted to base station.Which is costly affair.Since WSN nodes are resource-constrained , efficient data processing and conversing energy are prime challenges.We have discussed data aggregation taxonomy ,challenges and critical analyzed aggregation techniques.

TABLE OF CONTENTS

List of Figures

1. INTRODUCTION

1.1 GENERAL INTRODUCTION

Wireless sensor nodes are small-sized, resource-constrained, inexpensive devices deployed in large number to sense data from its field of deployment and send it to base station. WSN finds its application in different areas including healthcare monitoring, border surveillance, habitat monitoring. Due to small size, sensor nodes have small battery source which powers all other components (transmission, reception, sensing and processing). In certain scenarios, it becomes either difficult, costly or almost impossible to recharge or replace their battery. Minimizing energy dissipation in WSN is a primary challenge for researchers

To monitor actual environment or current status of equipment, Wireless sensor network which is an network consisting of a huge amount of sensor nodes are used. WSN is deployed increasingly in military, industrial and civilian applications. Sensor nodes organize themselves into a tree topology post deployment with base station as root of operations. Along path of tree topology Sensor nodes gathers different types of information within a fixed time interval sends them to base station (BS) for further processing. Consumption of energy during transmission will effects lifetime of sensor nodes in a network. Therefore, as an influential technique to minimize amount of data transfer methodology of data aggregation has been extensively applied in WSN. To decrease communication overflow and to extend lifetime of network TAG arranges every node into an aggregation tree. But, there is lack of data security with respect to privacy in TAG as malicious node can easily decrypt and access important information of other nodes. Slice-mix-aggregate (SMART) algorithm was put forth for addressing and resolving this problems. So as to conceal raw information of nodes every node in SMART slices its data into a predefined fixed number of packets and transmits data packets to its neighbor nodes in a network. Moreover, large amount of energy is consumed with slicing technique that leads to a huge number of exchanged messages.

1.2 PROJECT OBJECTIVES

- To capture the effects of aggregation over different scales.
- To efficiently and accurately obtain the sensor data from the nodes.
- To reduce the energy consumption and thus increase the lifetime of a network.
- To reduce the traffic
- To reduce the duplication data.

1.3 PROBLEM STATEMENT

- In Wireless Sensor Networks,accurate data extraction and aggregation is difficult is due to
- Significant variations in sensor readings
- Frequent link and node failure

2. SYSTEM PROPOSAL

2.1 **EXISTING SYSTEM**

Data aggregation is a very important technique, which is considered to significantly decrease the communication overhead and energy expenditure of sensor node during the process of data collection in WSN. However, privacy-preservation is additional challenging issue in data aggregation, where the aggregators need to perform some aggregation operations on sensing data it received.

2.1.1 ADVANTAGES

- Very high throughput
- Consumes very less energy
- Can maintain both spectral and energy efficiency
- Optimizing technique helps to harvest the energy

2.1.2 DISADVANTAGES

In the existing system, there as not considered the fact of cost constraints, only focused the power constraints.

2.2 PROPOSED SYSTEM

Privacy-preservation is additional challenging issue in data aggregation, where the aggregators need to perform some aggregation operations on sensing data it received. We classify the presented privacy-preserving data aggregation methods into different types by the core privacy-preserving techniques used in each scheme. And then compare and contrast different algorithms on the base of performance measures such as the privacy protection ability, communication consumption, power consumption, data accuracy and aggregation function

2.3 LITERATURE SURVEY

Title 1: Privacy preserving data aggregation in Wireless Sensor Networks

Author:XU JIAN ET AL

Year: 2017

Data aggregation is a very important technique, which is considered to significantly decrease the communication overhead and energy expenditure of sensor node during the process of data collection in WSN.

However, privacy-preservation is additional challenging issue in data aggregation, where the aggregators need to perform some aggregation operations on sensing data it received. We classify the presented privacy-preserving data aggregation methods into different types by the core privacy-preserving techniques used in each scheme. Then compare and contrast different algorithms on the base of performance measures such as the privacy protection ability, communication consumption, power consumption, data accuracy and aggregation function.

Based on the comparison result, we presented an overview of the solutions; adopt which type of privacy-preserving technique, and whether the solution is actually applicable for data aggregation in WSN.

Advantages:

- Improved energy efficiency.
- Low energy ,increased lifetime of network.

Disadvantages:

- It is not used for large network region.
- More power consumption

Title 2: Privacy-Preserving Access Control method for sensor networks.

Author:RUI ZHANG ET AL

Year:2012

The owner and users of a sensor network may be different, which necessitates privacy-preserving access control. The network owner need enforce strict access control so that the sensed data are only accessible to users willing to pay. Users wish to protect their respective data access patterns whose disclosure may be used against their interests. Users purchase tokens from the network owner whereby to query data from sensor nodes which will respond only after validating the tokens. The use of blind signatures in token generation guarantees that tokens are publicly provable until unpleasant to user individuality, so privacy-preserving access control is achieved. A central component is to prevent cruel users from reusing tokens, for which we recommend a suite of distributed token reuse detection schemes without concerning the base station. These schemes share the necessary scheme that a sensor node checks with some other nodes whether a token has been used, but they differ in how the witnesses are selected. We meticulously evaluate their performance with observe to TRD capability, communication overhead, storage overhead, and attack resilience. The efficacy and efficiency of AC are confirmed by detailed performance evaluations.

Advantages:

- Reduces the risk of unauthorized login
- Reduces the malware

Disadvantages:

- Time consuming
- Authorization need to change

3 SYSTEM DIAGRAMS

3.1 SYSTEM ARCHITECTURE

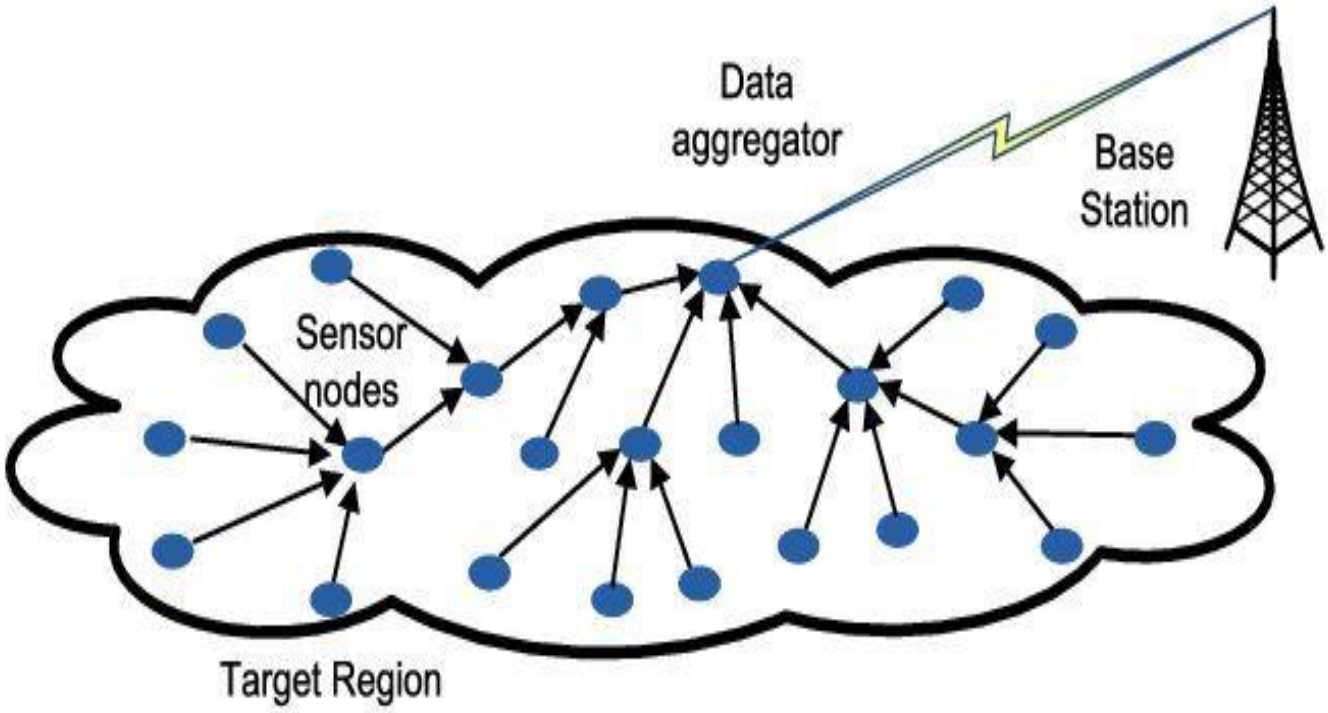


FIG 1: Privacy preserving data aggregation

3.2 FLOW DIAGRA

FIG 2:MODEL DIAGRAM

4 IMPLEMENTATION

4.1 MODULES:

- SENSOR NODES
- CLUSTERING
- DATA AGGREGATOR
- BASE STATION

4.2 MODULE DESCRIPTION

SENSOR NODE is a node in a sensor network that is capable of performing some process

CLUSTERING is one of the most popular techniques for Wireless Sensor Network management

DATA AGGREGATION is an organization that collects the data from the sensor and form the clusters

BASE STATION provides the communication link between the sensor network and end user

5 SYSTEM REQUIREMENTS

5.1 Hardware Requirements

- **System : Pentium IV 2.4 GHz**
- **Hard Disk : 1000 GB**
- **Monitor : 15 VGA color**

- **Mouse : LOGITECH**
- **Keyboard : 110 keys enhanced**
- **Ram : 4GB**

4.2 Software Requirements

- O/S : Windows 7
- Language : TCL.
- IDE : NS 2 STIMULATOR

4.3 SOFTWARE DESCRIPTION

Python

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van ROSSUM during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on Python programming language.

Python is a popular programming language. It was created in 1991 by Guido van ROSSUM.

It is used for:

- web development (server-side),
- software development,
- mathematics,
- System scripting.

Python can be used on a server to create web applications. Python can be used alongside software to create workflows. Python can connect to database systems. It

can also read and modify files. Python can be used to handle big data and perform complex mathematics. Python can be used for rapid prototyping, or for production-ready software development.

Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc). Python has a simple syntax similar to the English language. Python has syntax that allows developers to write programs with fewer lines than some other programming languages.Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick. Python can be treated in a procedural way, an object-orientated way or a functional way.

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

Python was developed by Guido van ROSSUM in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, Small Talk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).

Python is now maintained by a core development team at the institute, although Guido van ROSSUM still holds a vital role in directing its progress.

Python's features include −

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.
- **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.

- **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** − you can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** − Python provides interfaces to all major commercial databases.
- **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Saleable** − Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below –

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, Active X, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X.

Python Syntax compared to other programming languages

- Python was designed to for readability, and has some similarities to the English language with influence from mathematics.
- Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
- Python relies on indentation, using white space, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

Ns2 Simulator

- This network consists of 4 nodes (n0, n1, n2, n3).
- The duplex links between n0 and n2, and n1 and n2 have 2 MBPS of bandwidth and 10 ms of delay. The duplex link between n2 and n3 has 1.7 MBPS of bandwidth and 20 ms of delay.
- Each node uses a Drop Tail queue that has a maximum size of 10
- NS are a discrete event simulator targeted at networking research.
- NS-2 is built using C++ and Python with scripting capability

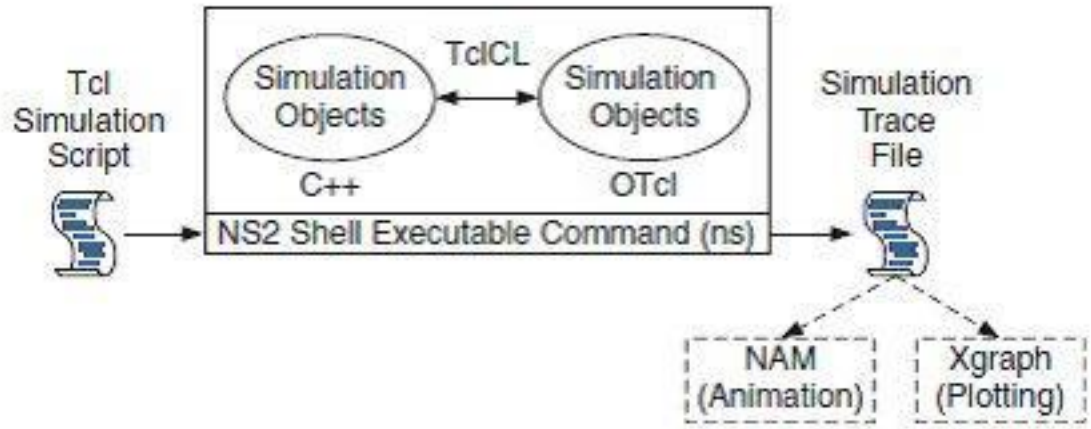
Network Simulator (NS) is simply a discrete event-driven network simulation tool for studying the dynamic nature of communication networks. Network Simulator 2 (NS2) provides substantial support for simulation of different protocols over wired and wireless networks. It provides a highly modular

platform for wired and wireless simulations supporting different network elements, protocols, traffic, and routing types.

NS2 is a simulation package that supports several network protocols including TCP, UDP, HTTP, and DHCP and these can be modeled using this package. In addition, several kinds of network traffic types such as constant bit rate (CBR), available bit rate (ABR), and variable bit rate (VBR) can be generated easily using this package. It is a very popular simulation package in academic environments.

Basic Architecture

NS2 consists of two key languages: C++ and Object-oriented Tool Command Language (OTCL). While the C++ defines the internal mechanism (i.e., a back-end) of the simulation objects, the OTCL sets up simulation by assembling and configuring the objects as well as scheduling discrete events. The C++ and the OTCL are linked together using TCLCL.



Basic architecture of NS.

NS2 stands for Network Simulator Version 2. It is an open-source event-driven simulator designed specifically for research in computer communication networks.

NS2 uses OTCL to create and configure a network, and uses C++ to run simulation. All C++ codes need to be compiled and linked to create an executable file.

Use OTCL

- For configuration, setup, or one time simulation, or

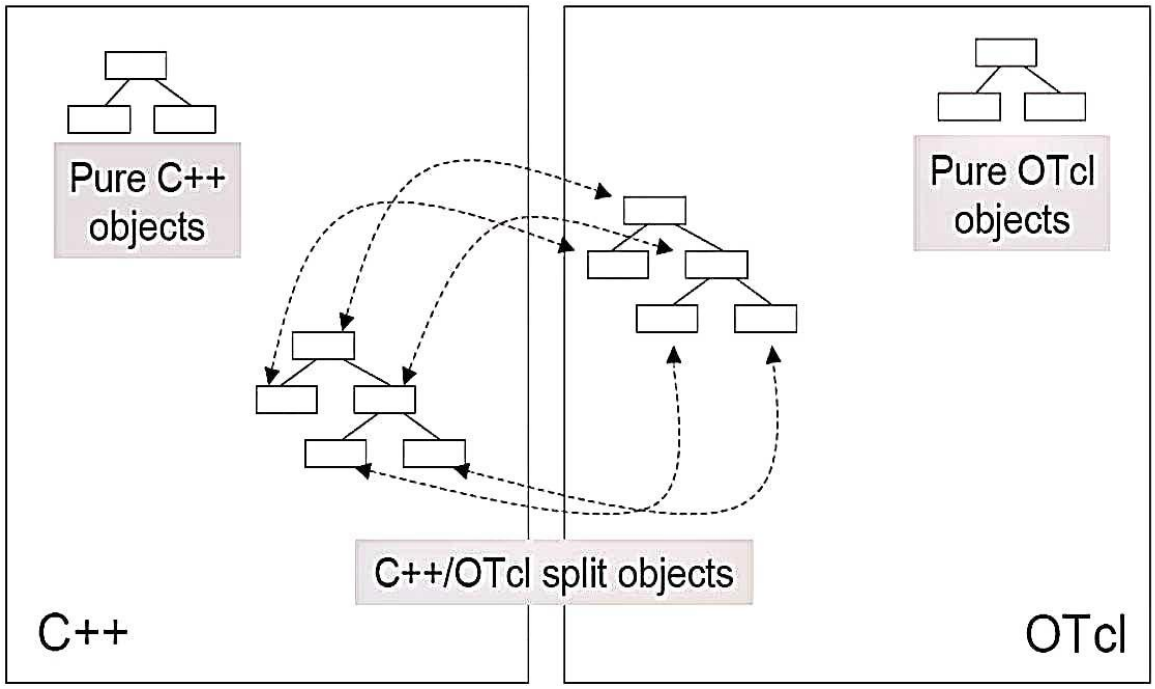
- To run simulation with existing NS2 modules.

This option is preferable for most beginners, since it does not involve complicated internal mechanism of NS2. Unfortunately, existing NS2 modules are fairly limited. This option is perhaps not sufficient for most researchers.

Use C++

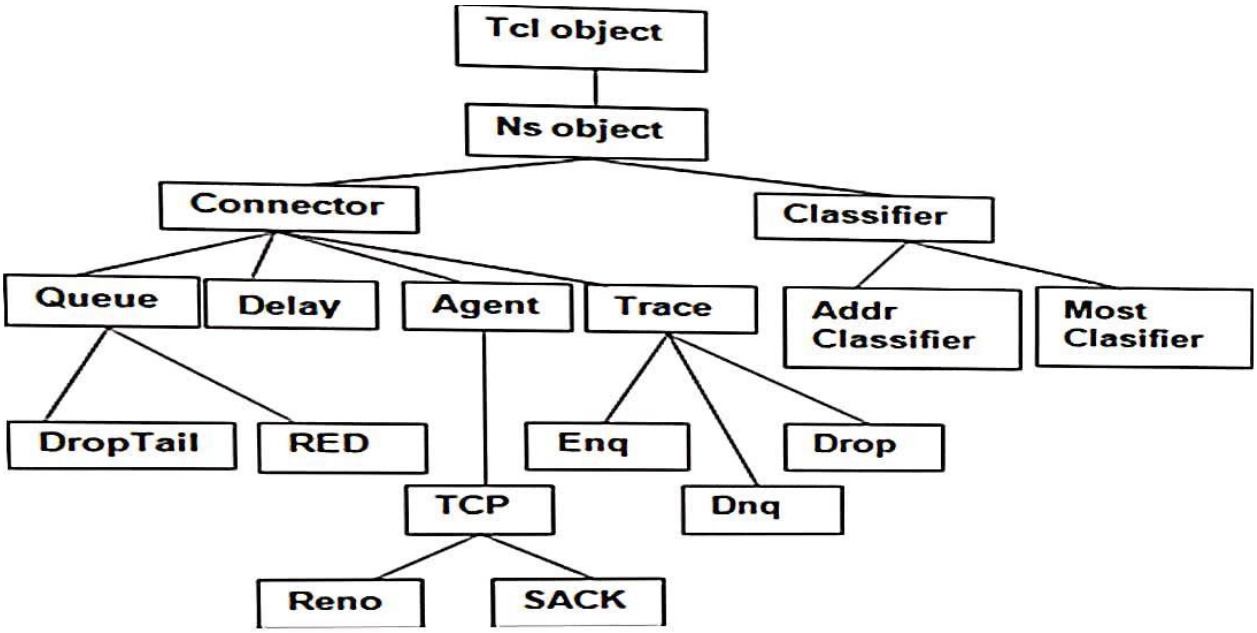
- When you are dealing with a packet, or - when you need to modify existing NS2 modules.

This option perhaps discourages most of the beginners from using NS2. This book particularly aims at helping the readers understand the structure of NS2 and feel more comfortable in modifying NS2 modules.



Hierarchy of an Object in NS-2

TCL Object is the base class for most of the other classes in the interpreted and compiled hierarchies. Every object in the class TCL Object is created by the user from within the interpreter.



Work efficiently in a multi-language editor with a function/class browser, code analysis tools, automatic code completion, horizontal/vertical splitting, and go-to-definition.

I Python Console

Harness the power of as many I Python consoles as you like within the flexibility of a full GUI interface; run your code by line, cell, or file; and render plots right inline.

Variable Explorer

Interact with and modify variables on the fly: plot a histogram or time series, edit a date frame or NUMPY array, sort a collection, dig into nested objects, and more!profile

Find and eliminate bottlenecks to unchain your code's performance.

Debugger

Trace each step of your code's execution interactively.

Help

Instantly view any object's docs, and render your own.

FEASIBILITY STUDY

The feasibility study is carried out to test whether the proposed system is worth being implemented. The proposed system will be selected if it is best enough in meeting the performance requirements. The feasibility carried out mainly in three sections namely.

- Economic Feasibility
- Technical Feasibility
- Behaviour Feasibility

Economic Feasibility

Economic analysis is the most frequently used method for evaluating effectiveness of the proposed system. More commonly known as cost benefit analysis. This procedure determines the benefits and saving that are expected from the system of the proposed system.

Technical Feasibility :

This study centre around the system's department hardware, software and to what extend it can support the proposed system department is having the required hardware and software there is no question of increasing the cost of implementing the proposed system. The criteria, the proposed system is technically feasible and the proposed system can be developed with the existing facility.

Behaviour Feasibility

People are inherently resistant to change and need sufficient amount of training, which would result in lot of expenditure for the organization. The proposed system can generate reports with day-to-day information immediately at the user's request, instead of getting a report, which doesn't contain much detail.

5.4 TESTING OF PRODUCT

Testing of Product

System testing is the stage of implementation, which aimed at ensuring that system works accurately and efficiently before the live operation commence. Testing is the process of executing a program with the intent of finding an error. A good test case is one that has a high probability of finding an error. A successful test is one that answers a yet undiscovered error.

Testing is vital to the success of the system. System testing makes a logical assumption that if all parts of the system are correct, the goal will be successfully achieved. The candidate system is subject to variety of tests-on-line response, Volume Street, recovery and security and usability test. A series of tests are performed before the system is ready for the user acceptance testing. Any engineered product can be tested in one of the following ways. Knowing the specified function that a product has been designed to from, test can be conducted to demonstrate each function is fully operational. Knowing the internal working of a product, tests can be conducted to ensure that “al gears mesh”, that is the internal operation of the product performs according to the specification and all internal components have been adequately exercised.

UNIT TESTING

Unit testing is the testing of each module and the integration of the overall system is done. Unit testing becomes verification efforts on the smallest unit of software design in the module. This is also known as ‘module testing’. The modules of the system are tested separately. This testing is carried out during the programming itself. In this testing step, each model is found to be working satisfactorily as regard to the expected output from the module. There are some validation checks for the fields.

INTEGRATION TESTING

Data can be lost across an interface, one module can have an adverse effect on the other sub function, when combined, may not produce the desired major function. Integrated testing is systematic testing that can be done with sample data. The need for the integrated test is to find the overall system performance. There are two types of integration testing. They are:

i) Top-down integration testing.
ii) Bottom-up integration testing.

WHITE BOX TESTING

White Box testing is a test case design method that uses the control structure of the procedural design to drive cases. Using the white box testing methods, we derived test cases that guarantee that all independent paths within a module have been exercised at least once.

BLACK BOX TESTING

- ✔ Black box testing is done to find incorrect or missing function
- ✔ Interface error
- ✔ Errors in external database access
- ✔ Performance errors
- ✔ Initialization and termination errors

In ‘functional testing’, is performed to validate an application conforms to its specifications of correctly performs all its required functions.It tests the external behaviour of the system. Here the engineered product can be tested knowing the specified function that a product has been designed to perform, tests can be conducted to demonstrate that each function is fully operational.

VALIDATION TESTING

After the culmination of black box testing, software is completed assembly as a package, interfacing errors have been uncovered and corrected and final series of software validation tests begin validation testing can be defined as many, but a single definition is that validation succeeds when the software functions in a manner that can be reasonably expected by the customer.

USER ACCEPTANCE TESTING

User acceptance of the system is the key factor for the success of the system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system at the time of developing changes whenever required.

OUTPUT TESTING

After performing the validation testing, the next step is output asking the user about the format required testing of the proposed system, since no system could be useful if it does not produce the required output in the specific format. The output displayed or generated by the system under consideration. Here the output format is considered in two ways. One is screen and the other is printed format. The output format on the screen is found to be correct as the format was designed in the system phase according to the user needs. For the hard copy also output comes out as the specified requirements by the user. Hence the output testing does not result in any connection in the system.

Agile Testing

Agile Testing is a type of software testing that accommodates agile software development approach and practices. In an Agile development environment, testing is an integral part of software development and is done along with coding. Agile testing allows incremental and iterative coding and testing.

API Testing

API testing is a type of testing that is similar to unit testing. Each of the Software API are tested as per API specification. API testing is mostly done by testing team unless API to be tested or complex and needs extensive coding. API testing requires understanding both API functionality and possessing good coding skills.

Automated testing

This is a testing approach that makes use of testing tools and/or programming to run the test cases using software or custom developed test utilities. Most of the automated tools provided capture and playback facility, however there are tools that require writing extensive scripting or programming to automate test cases.

End-to-end Testing

End to end testing is performed by testing team, focus of end to end testing is to test end to end flows e.g. right from order creation till reporting or order creation till item return etc. and checking. End to end testing is usually focused mimicking real life scenarios and usage. End to end testing involves testing information flow across applications.

Exploratory Testing

Exploratory testing is an informal type of testing conducted to learn the software at the same time looking for errors or application behaviour that seems non-obvious. Exploratory testing is usually done by testers but can be done by other stake holders as well like Business Analysts, developers, end users etc. who are interested in learning functions of the software and at the same time looking for errors or behaviour is seems non-obvious.

Performance Testing

It is a type of software testing and part of performance engineering that is performed to check some of the quality attributes of software like Stability, reliability, availability. Performance testing is carried out by performance engineering team. Unlike Functional testing, Performance testing is done to check non-functional requirements. Performance testing checks how well software works in anticipated and peak workloads. There are different variations or sub types of performance like load testing, stress testing, volume testing, soak testing and configuration testing.

Penetration Testing

It is a type of security testing, also known as pen test in short. Penetration testing is done to tests how secure software and its environments (Hardware, Operating system and network) are when subject to attack by an external or internal intruder. Intruder can be a human/hacker or malicious programs. Pen test uses methods to forcibly intrude (by brute force attack) or by using a weakness (vulnerability) to gain access to a software or data or hardware with an intent to expose ways to steal, manipulate or corrupt data, software files or configuration. Penetration Testing is a way of ethical hacking

Security Testing

It is a type of software testing carried out by specialized team of software testers. Objective of security testing is to secure the software is to external or internal threats from humans and malicious programs. Security testing basically checks, how good is software's authorization mechanism, how strong is authentication, how software maintains confidentiality of the data, how does the software maintain integrity of the data, what is the availability of the software in an event of an attack on the software by hackers and malicious programs is for Security testing requires good knowledge of application, technology, networking, security testing tools. With increasing number of web applications necessarily of security testing has increased to a greater extent.

Sanity Testing

It is a type of testing that is carried out mostly by testers and in some projects by developers as well. Sanity testing is a quick evaluation of the software, environment, network, external systems are up & running, software environment as a whole is stable enough to proceed with extensive testing. Sanity tests are narrow and most of the time sanity tests are not documented.

Scalability Testing

It is a non-functional test intended to test one of the software quality attributes i.e. "Scalability". Scalability test is not focused on just one or few functionality of the software instead performance of software as a whole. Scalability testing is usually done by performance engineering team. Objective of scalability testing is to test the ability of the software to scale up with increased users, increased transactions, increase in database size etc.

Stability Testing

It is a non-functional test intended to test one of the software quality attributes i.e. “Stability”. Stability testing focuses on testing how stable software is when it is subject to loads at acceptable levels, peak loads, loads generated in spikes, with more volumes of data to be processed. Scalability testing will involve performing different types of performance tests like load testing, stress testing, spike testing, soak testing, spike testing etc.…

Static Testing is a form of testing where in approaches like reviews, walk through are employed to evaluate the correctness of the deliverable. In static testing software code is not executed instead it is reviewed for syntax, commenting, naming convention, size of the functions and methods etc. Static testing usually has check lists against which deliverable are evaluated. Static testing can be applied for requirements, designs, and test cases by using approaches like reviews or walk-through

Stress Testing is a type of performance testing, in which software is subjected to peak loads and even to a break point to observe how the software would behave at break-point. Stress testing also tests the behaviour of the software with insufficient resources like CPU, Memory, Network bandwidth, Disk space etc. Stress testing enables to check some of the quality attributes like robustness and reliability.

6 CONCLUSION AND FUTURE ENHANCEMENT

6.1 CONCLUSION

In this paper we have studied the data communication in sensor networks i.e data aggregation and realized how communication in sensor networks is different from other wireless networks. Wireless sensor networks are energy constrained network. Since most of the energy consumed for transmitting and receiving data, the process of data aggregation becomes an important issue and optimization is needed. Efficient data aggregations not only provide energy conservation but also remove redundancy data and hence provide useful data only. When the data from source node is send to sink through neighbours nodes in a multihop fashion by reducing transmission and receiving power, the energy consumption is low as compared to that of sending data directly to sink that is aggregation reduces the data transmission then the without aggregation. In this paper, we have proposed an energy-efficient techniques for data aggregation in wireless sensor networks. Our scheme integrates energy-efficient and data storage mechanisms. This survey paper shows that these techniques not only reduces power consumption but also prolongs the lifetime of a network.

3. REFERENCES

1 Gupta, P and M. CHAWALA, 2012. "Privacy preservation for WSN: A survey," Int. J. COMPUT. APPL, 48(3): 11-16.

2 W. He, X. LIU, H. Nguyen, K. NAHRSTEDT, and T. ABDELZAHER (2002), “Pda: 508 Privacy-preserving data aggregation in wireless sensor networks,” in 509 IEEE INFOCOM 2007-26th IEEE International Conference on Com- 510 puter Communications. IEEE, pp. 2045–2053.

3 ESMAEIL Rezaei and Safieh Ghasemi (2018), “Energy-Aware Data Aggregation in Wireless Sensor Networks Using Particle Swarm Optimization Algorithm”, American Journal of Information Science and Computer Engineering, vol. 4, no. 1, pp. 1 - 6, 2018.

4 YADAV S, Yadav RS (2019) Redundancy elimination during data aggregation in wireless sensor networks . In: Recent trends in communication, computing, and electronics. Springer, Berlin, pp 195-205

5 Yadav S, Yadav RS (2019) Redundancy elimination during data aggregation in wireless sensor networks for iot systems. In: Recent trends in communication, computing, and electronics. Springer, Berlin, pp 195-205

6 Roy NR, Chandra P (2018) A note on optimum cluster estimation in leach protocol. IEEE Access 6:65690-65696

7 T. Wang, X. Qin, Y. Ding, L. Liu, and Y. Luo (2018), “Privacy-preserving 512 and energy-efficient continuous data aggregation algorithm in wireless 513 sensor networks,” Wireless Personal Communications, vol. 98, no. 1, pp. 514 665–684.