ლინა მღებრიშვილი

ინფორმაციის დაცვა კომპიუტერულ ქსელებში

წარმოდგენილია დოქტორის აკადემიური ხარისხის
მოსაპოვებლად

საქართველოს ტექნიკური უნივერსიტეტი
თბილისი, 0175, საქართველო

საქართველოს ტექნიკური

უნივერსიტეტი

ინფორმატიკისა და მართვის სისტემების ფაკულტეტი

ჩვენ, ქვემორე ხელისმომწერნი ვადასტურებთ, რომ გავეცანით მღებრიშვილი ლინას მიერ შესრულებულ სადისერტაციო ნაშრომს დასახელებით: "ინფორმაციის დაცვა კომპიუტერულ ქსელებში" და ვაძლევთ რეკომენდაციას საქართველოს ტექნიკური უნივერსიტეტის ინფორმატიკისა და მართვის სისტემების ფაკულტეტის სადისერტაციო საბჭოში მის განხილვას დოქტორის აკადემიური ხარისხის მოსაპოვებლად.

თარიღი

ხელმძღვანელი: _____

ხელმძღვანელი: _____

რეცენზენტი: _____

რეცენზენტი: _____

საქართველოს ტექნიკური უნივერსიტეტი

2009

ავტორი:	მღებრიშვილი ლინა
დასახელება:	ინფორმაციის დაცვა კომპიუტერულ ქსელებში
ფაკულტეტი :	ინფორმატიკისა და მართვის სისტემების ფაკულტეტი
ხარისხი:	დოქტორი
სხდომა ჩატარდა:	

ავტორის ხელმოწერა

რეზიუმე

უსადენო ქსელებში უსაფრთხოება და მომსახურების ხარისხი უკანასკნელ დროს უაღრესად მნიშვნელოვანი და აქტიური კვლევის საგანი გახდა, რის მიზეზს აუდიოსა და ვიდეოს ცოცხალი გადაცემის მხარდაჭერის მზარდი მოთხოვნა წარმოადგენს, როგორც სამოქალაქო, ასევე სამხედრო სფეროში. ადექვატური უსაფრთხოების გარეშე საწარმოები თავს აარიდებენ უსადენო ქსელების გამოყენებას, თავდაცვის ორგანიზაციებმა შესაძლოა ვერ უზრუნველყონ პერსონალის უსაფრთხოების გარანტიები ბრძოლის ველზე, მომხმარებლები კი პასუხისმგებელნი გახდებიან ქმედებებისთვის, რომლებიც მათ არასდროს ჩაუდენიათ. უსაფრთხოების საკითხები უსადენო ქსელებში მნიშვნელოვან დაბრკოლებას წარმოადგენს ასეთი ქსელების ფართოდ ადაპტირებისთვის. შესაბამისად, მსგავსი უსადენო ქსელების უსაფრთხოება მნიშვნელოვანი სფეროა, რაც რეაგირებას მოითხოვს, თუკი ასეთი ქსელები ფართოდ იქნება გამოყენებული. აუცილებელია, რომ აღნიშნული სფეროს მკვლევარებმა მოახდინონ ღია პრობლემების იდენტიფიცირება და უზრუნველყონ შესაბამისი გადაწყვეტილებები ამ პრობლემებისთვის. თითოეული ასეთი მცდელობა უსადენო ქსელს ოდნავ უფრო უსაფრთხოს ხდის.

წინამდებარე კვლევის მიზანს ის წარმოადგენს, რომ შემუშავდეს რიგი ღონისძიებებისა, რომლებიც აამაღლებს უსადენო ქსელების უსაფრთხოებას. ყოველივე ზემოთქმულიდან გამომდინარე, სადისერტაციო ნაშრომში დასმულია შემდეგი ამოცანა: უსადენო ქსელებში მარშრუტიზაციის უსაფრთხოების ამაღლება.

ამასთან დაკავშირებით წინამდებარე ნაშრომში მოყვანილია უსაფრთხოების ფუნდამენტური პრინციპები, ისევე, როგორც ღია პრობლემები. მოცემულია უსადენო ქსელების უსაფრთხოების საკითხების ყოვლისმომცველი მიმოხილვა და ამასთან ერთად განიხილება დღემდე შემოთავაზებული სხვადასხვა სქემების უპირატესობები და ნაკლოვანებები.

უნდა აღინიშნოს, რომ უსადენო ქსელების მარშრუტიზაციის პროტოკოლები სპეციფიკაციებში არ განსაზღვრავენ რაიმე სახის პრევენციულ ღონისძიებებს ან უსაფრთხოების მექანიზმებს. ამდენად, უსადენო ქსელების მარშრუტიზაციის

პროტოკოლების უსაფრთხოება გადაუდებელ აუცილებლობად იქცა ქსელის გაშვების სტიმულირებისა და გამოყენების სფეროს გაფართოებისთვის.

შესაბამისად, წინამდებარე ნაშრომში შემოთავაზებულ და განსაზღვრულ იქნა განსხვავებული გადაწყვეტილებები და კონცეფციები უსაფრთხოების მიმართულებით. ძირითადი ყურადღება თავდაპირველად გამახვილებულია საწყის ნაბიჯზე – უსადენო მარშრუტიზაციის პროტოკოლების ხარვეზების შესწავლასა და ანალიზზე. თავდასხმათა შედეგების შეფასებისთვის სამიზნე მარშრუტიზაციის უსადენო პროტოკოლად თავდასხმისთვის არჩეულია OLSR პროტოკოლი. განხილულია უსადენო ქსელებში OLSR პროტოკოლის მიერ გენერირებული მაკონტროლებელი ტრაფიკის უსაფრთხოება. მოცემულია თავდასხმებისა და ნაკლოვანებების ტაქსონომია, წარმოდგენილია არსებული უსაფრთხოების გადაწყვეტილება, რომელიც კვანძებს ავჯილდოვებს მარშრუტიზაციის ინფორმაციის გაცვლის ოპერაციების შესაბამისად. ნაჩვენებია რომ შემოთავაზებული სქემა, რომელიც წარმატებით მიწოდებული პაკეტებიდან მიღებული მარშრუტების ინფორმაციას უკავშირებს გადაცემებზე უშუალო დაკვირვებას, ამსუბუქებს უსაფრთხოების საკითხებს. მაგრამ ამასთან ერთად მისთვის დამახასიათებელია გარკვეული ნაკლოვანებები, როგორიც არის: დაკვირვების მექანიზმის გამოყენება, რომელსაც ახასიათებს გარკვეული ცდომილება; გამაფრთხილებელი შეტყობინებების გავრცელება ქსელში, რაც საშუალებას იძლევა დადანაშაულდეს კარგად მომუშავე კვანძები და ამის შესახებ ინფორმაცია გავრცელდეს ქსელში, და ა.შ. ამ ნაკლოვანებებიდან გამომდინარე, წარმოდგენილ სადისერტაციო ნაშრომში შემოთავაზებულია უსაფრთხიების მოდიფიცირებული ალგორითმი (მექანიზმი), რომელიც იყენებს რეიტინგების სისტემას, ეყრდნობა ახალ დავჯილდოვებისა და დასჯის მეთოდს და თავის მუშაობაში კვანძების ქცევის დადგენის მიზნით აკონტროლებს OLSR-პროტოკოლისათვის დამახასიათებელ შეტყობინებებს, როგორიც არის TC და HELLO. წარმოდგენილი მოდიფიკაცია საშუალებას იძლევა უფრი ზუსტად მოხდეს მცდარად მომუშავე კვანძების აღმოჩენა და მათი დასჯა, ხოლო ადრე მცდარად მომუშავე კვანძებისათვის მათი გამოსწორების შემთხვევაში გათვალისწინებულია რეპუტაციის აღდგენის პროცედურა. გარდა ამისა აღკვეთილია არსებული უსაფრთხოების ალგორითმისათვის ისეთი დამახასიათებელი ნაკლოვანებები როგორიც არის გამაფრთხილებელი შეტყობინების გავრცელება, კვანძების მოძრაობასთან დაკავშირებული შეცდომები და სხვა.

სადისერტაციო ნაშრომში წარმოდგენილია შემუშავებული მოდიფიცირებული ალგორითმის მოდელირების შედეგები. მოდელირება ჩატარებულია ქსელის სიმულატორის ns2 ვერსია 2.29.2- გამოყენებით. წარმოდგენილია გრაფიკული ნახაზები, რომლებზედაც ასახულია კვანძების ქცევა და მათი რეიტინგები სხვადასხვა სიტუაციებში. მოდელირების შედეგები ამტკიცებს შემუშავებული მოდიფიცირებული ალგორითმის ფუნქციონირების ეფექტურობას.

სადისერტაციო ნაშრომი შედგება შესავლის, ოთხი თავისაგან, დასკვნების, გამოყენებული ლიტერატურის სიისაგან და ორი დანართისგან.

შესავალში ზოგადად დახასიათებულია სადისერტაციო ნაშრომის პრობლემატიკა.

პირველ თავში აღწერილია უსადენო ქსელები და მათი თავისებურებანი. დახასიათებულია შესაძლო საფრთხეები. ამავე თავში აღწერილია ის თავდასხმები, რომლებსაც შეიძლება ადგილი ჰქონდეს უსადენო ქსელებში. გარდა ამისა განხილულია კრიპტოგრაფიის მეთოდები.

მეორე თავში განხილულია უსადენო ქსელებში მარშრუტიზაციის რეალიზაციის საკითხები. ამასთან დაკავშირებით აღნიშნულია ის მომატებული საფრთხეები, და ის მიზეზები, რომლებიც განაპირობებენ მომატებულ საშიშროებას უსადენო ქსელებში. დახასიათებულია არსებული მარშრუტიზაციის პროტოკოლები და მოყვანილია მათზე შესაძლო თავდასხმები.

მესამე თავში განხილულია უსადენო ქსელებისათვის შემუშავებული OLSR მარშრუტიზაციის პროტოკოლი. აღწერილია მისი ფუნქციები და შეტყობინებები, მოცემულია მათი ფორმატი.

მეოთხე თავში მოყვანილია OLSR პროტოკოლის არსებული გაფართოვება შემუშავებული მისი საიმედობის ამაღლების მიზნით. წარმოდგენილია აღნიშნული გაფართოვების ჩვენს მიერ შემუშავებული უსაფრთხოების მოდიფიცირებული ალგორითმი.

დასკვნებში აღნიშნულია, რომ:

- რეპუტაციის კონცეფციაზე დაფუძნებული უსაფრთხოების უზრუნველყოფა არის უფრო ეფექტური, ვიდრე მხოლოდ კრიპტოგრაფიული მეთოდებით უზრუნველყოფილი;
- შემოთავაზებულია კვანძების მუშაობის რეპუტაციის ფუნქციონალური დამოკიდებულება რეიტინგების სისტემაზე;

- შემუშავებულ მოდიფიცირებულ OLSR პროტოკოლს ემატება მხოლოდ ორი ელემენტი, არ საჭიროებს გამაფრთხილებელი შეტყობინების გავრცელებას, რაც იცავს სწორად მომუშავე კვანძების უსაფუძვლოდ დადანაშაულებისაგან;
- მოდიფიცირებული ალგორითმი ამოწმებს OLSR-სთვის დამახასიათებელ შეტყობინებებს და მათი შემოწმება წარმოებს ისეთი მეთოდით, რომელიც უფრო სანდოს ხდის თავად შემოწმებას და თავიდან გვაცილებს გადაადგილების შეცდომების გენერირებას;
- გააჩნია მოქნილი დასჯისა და დავიწყების მექანიზმები, რომელიც უზრუნველყოფს კვანძების უფრო ეფექტურ მუშაობას;
- შემოთავაზებული რეიტინგების სისტემა უფრო ეფექტურს ხდის კვანძების მუშაობას (რეიტინგების საწყისი მნიშვნელობების ცვლა, დასჯისა და დავიწყების მექანიზმი, ყალბი მდგომარეობიდან აღდგენის მართვა);
- შემოთავაზებული მოდიფიცირებული მეთოდი ეფექტურია სხვადასხვა თავდასხმების წინააღმდეგ და
- დამუშავებული მეთოდი ეკონომიურია ჭარბი ინფორმაციის გადაცემის თვალსაზრისით.

სადისერტაციო ნაშრომს აგრეთვე ახლავს დანართები: პირველ დანართში მოცემულია მოდელირების დროს გამოყენებული კოდი, ხოლო მეორე დანართში მოცემულია კვანძების გადაადგილების სცენარი.

Abstract

Security and quality of service in wireless networks have recently become very important and actively researched topics because of a growing demand to support live streaming audio and video in civilian as well as military applications. Without adequate security, enterprises will shy away from the use of wireless networks, governmental, defense organizations might be unable to guarantee the safety of their personnel in battlefield scenarios and users will be liable for actions that they never committed. The security concerns in wireless networking remains a serious impediment to widespread adoption of wireless networks. Thus, the security of such wireless networks is an important area that needs to be addressed if such networks are to be widely used. Very important is for the researchers in this field to identify open problems and provide solutions to the identified open problems. Each such effort makes these wireless networks a little bit more secure.

The objective of this research is to identify a set of measures that will increase the security of wireless networks. Therefore, the following topic is analyzed in this thesis: increasing the routing security in wireless networks.

Based on this, the thesis makes aware of the fundamentals of the area of security of wireless networks as well as the open problems. This will hopefully spur much more activity in this area. This thesis provides a comprehensive overview of the security of wireless networks and discusses the advantages and disadvantages of the various schemes that have been proposed so far.

Routing protocols for wireless networks haven't defined any prevention measures, or security mechanisms in their specifications. Securing wireless routing protocols had then appeared as an urgent need in order to promote the network deployment and to widen its application domains.

Consequently, in this thesis different solutions and concepts were proposed and defined. The primary focus in a preliminary step is on the study and the analyses of the wireless routing protocols vulnerabilities before proceeding to the conception of a security solution. In order to evaluate attacks consequences, the OLSR protocol is chosen as the targeted wireless routing protocol to attack. Focusing on the Optimized Link State Routing (OLSR) protocol, taxonomy of attacks and vulnerabilities is provided based on which a security solution that rewards nodes depending on their cooperation in the exchange of routing information is proposed. The proposed

scheme, which correlates direct observation of transmissions with path information from successfully delivered packets, is shown to mitigate a relevant set of security issues. However it is characterized with some disadvantages as well such as: usage of observation mechanism, which have some level of inaccuracy; spreading of warning messages in the network, which gives the possibility to punish the correct working nodes and spread this information in the network, etc. Based on these disadvantages the modified security algorithm (mechanism) is proposed in this thesis which uses the system of rating focusing on new method of awarding and punishment. It also controls the OLSR messages such as TC and HELLO in order to know the behavior of nodes. The introduced modification gives the possibility to identify and punish more accurately the fault nodes. It also considers the recover procedure for these nodes. Apart from this, the following disadvantages that are characterizing the security algorithm are mitigated: spreading of warning messages, faults related to movement of nodes, etc.

The simulation results of introduced algorithm are given in the research. The simulation is done with network simulator NS2 ver.2.29.2. The graphical schemes are reflecting the behavior of nodes and their rating in different situations. The simulation results prove the effectiveness of the modified algorithm.

The thesis consists of introduction, four chapters, conclusion, references and two Appendices.

The main issues of the research are generally discussed in introduction.

Chapter 1 focuses on wireless networks and their characteristics. Those attacks are considered that might affect such networks. Possible vulnerabilities are discussed together with the cryptography methods.

Chapter 2 considers the routing schemes in wireless networks and the increased vulnerabilities related to routing issues in such networks. Different routing protocols are analyzed together with possible attacks.

Chapter 3 focuses on OLSR routing protocol which is designed for wireless networks. Its functionality, messages and format are analyzed.

Chapter 4 discusses already existing improvements of OLSR protocol from the point of view of security. The development of modified algorithm is also presented in this chapter.

And finally, the conclusion states that:

- Security based on reputation mechanism is more effective than security based only on cryptographic methods
- Functional dependency on rating system of functionality of nodes is introduced.
- in the developed modified algorithm of OLSR protocol only two elements are added, there is no need for spreading the warning messages which secures exactly the correctly behaving nodes from faulty punishment
- The modified algorithm checks the messages characterizing the OLSR protocol and their checking is done via the method which makes more secure the checking itself and avoids the generation of movements faults.
- Has more flexible punishment and awarding mechanisms which ensure more effective functionality of nodes.
- The introduced rating system makes more effective the functionality of nodes (the change of initial rating states, punishment and awarding mechanisms, recovery from wrong states)
- Is more effective against various attacks
- Is more efficient from the point of view of excessive transmission of information.

The thesis also includes appendices: Code used for modeling is given in Appendix 1. Scenario for movement of nodes is given in Appendix 2.

შინაარსი

ცხრილების ნუსხა

სურათების ნუსხა

დისერტაციაში გამოყენებული აბრევიატურები

AODV	Ad hoc On-demand Distance Vector routing; მიზნობრივი მოთხოვნით დისტანციურ-ვექტორული მარშრუტიზაცია
ADV	Adaptive Distance Vector routing; ადაპტური დისტანციურ-ვექტორული მარშრუტიზაცია
ADVSIG	ADVanced SIGnature message; გაუმჯობესებული ხელმოწერის შეტყობინება
AES	Advanced Encryption Key; კოდირების თანამედროვე სტანდარტი
Bluetooth	უსადენო ქსელის პროტოკოლი
CA	Certification Authority; სერტიფიცირების ორგანო
CBRP	Cluster Based Routing Protocol; კლასტერზე დაფუძნებული მარშრუტიზაციის პროტოკოლი
DES	Data Encryption Standard; მონაცემთა კოდირების სტანდარტი
DH	Diffie–Hellman protocol; დიფი-ჰელმანის პროტოკოლი
DoS	Denial of Service; სერვისის მტყუნება
DSDV	Destination-Sequenced Distance -Vector routing; დანიშნულება-თანმიმდევრული დისტანციურ-ვექტორული მარშრუტიზაცია
DSR	Dynamic Source Routing; წყაროდან დინამიკური მარშრუტიზაცია
End-to-end	გამჭოლი
EXE file	შესრულებადი ფაილი
ETSI	European Telecommunications Standards Institute; ევროპული ტელეკომუნიკაციების სტანდარტების ინსტიტუტე
Firewall	ქსელთაშორისი ეკრანი
FIPS	Federal Information Processing Standard; ინფორმაციის დამუშავების ფედერელური სტანდარტი
FSR	Fisheye State Routing; თევზისთვალა მდგომარეობის მარშრუტიზაცია
HDLC	High-Level Data Link Control – ბიტ ორიენტირებული ქსელური პროტოკოლი
HiperLAN	(HIgh PErformance Radio LAN) – უსადენო ქსელის სტანდარტი
HMAC	Hashed Message Authentication Code; ჰეშირებული შეტყობინების იდენტიფიკაციის კოდი
HomeRF	უსადენო ქსელის ორგანიზების სქემა სახლის მოწყობილობებისათვის
Hop-by-hop	ბიჯური
IARP	IntrAzone Routing Protocol; IntrAzone მარშრუტიზააციის პროტოკოლი
IERP	IntErzone Routing Protocol; IntErzone მარშრუტიზააციის პროტოკოლი
IP	Internet Protocol; ინტერნეტ პროტოკოლი
ISM radio band	industrial, scientific and medical; საწარმოო, სამეცნიერო, სამედიცინო რადიო სიხშირე

Java	პროგრამირების ენა
LAN	Local Area Network; ლოკალური ქსელი
LANMAR	LANdMARk routing; LANdMARk მარშრუტიზაცია
LSA	Link-State Advertisement; არხის მდგომარების შეტყობინება
LSU	Link-State Update message; არხის მდგომარეობის განახლების შეტყობინება
MAC	Message Authentication Codes; შეტყობინების აუთენტიფიკაციის კოდები
MedAC	Medium Access Control; გარემოს მისაწვდომობის მართვა
MID	Multiple Interface Declaration; მრავლობითი ინტერფეისის დეკლარაცია
MitM	Man-in-the-Middle; შუაში მდგომი თავდასხმა
MPR	Multipoint Relay; მრავალპუნქტიანი რელე
NIST	National Institute of Standards and Technology; სტანდარტების და ტექნოლოგიების ეროვნული ინსტიტუტე
NHA	Network Association; ქსელის ასოციაცია
NSA	National Security Agency; უსაფრთხოების ეროვნული სააგენტო
OLSR	Optimized Link State Routing protocol; ოპტიმიზირებული არხის მდგომარეობის მარშრუტიზაციის პროტოკოლი
OSI	Open Systems Interconnection; ღია სისტემების ურთიერთდაკავშირება
OSPF	Open Shortest Path First routing; თავისუფალი უმკოლესი მარშრუტი პირველად მარშრუტიზაცია
PRV	Primary Recovery Value; პირველადი აღდგენის სიდიდე
PV	Punishment Value; დასჯის სიდიდე
SHA	Secure Hash Algorithm; უსაფრთო ჰეშ ალგორითმი
SLSP	Secure Link-State routing Protocol; უსაფრთხო არხის მდგომარეობის მარშრუტიზაციის პროტოკოლი
SNMP	Simple Network Management Protocol; მარტივი ქსელის მართვის პროტოკოლი
SRV	Secondary Recovery Value; მეორადი აღდგენის სიდიდე
STAR	Source Tree Adaptive Routing ; საწყისი ხის ადაპტური მარშრუტიზაცია
SYN	სინქრონიზაცია
TBRPF	Topology dissemination Based on Reverse-Path Forwarding routing; ტოპოლოგიის განფანტვა, დაფუძნებული უკუმიმართულების გადაცემით მარშრუტიზაცია
TC	Topology Control; ტოპოლოგიის კონტროლი
TCP	Transmission Control Protocol; მონაცემთა გადაცემის მართვის პროტოკოლი
ThC	Threshold Criptography; ზღვრული კრიპტოგრაფია
TTL	Time To Live; არსებობის დრო
UDP	User Datagram Protocol; მომხმარებილის დეიტაგრამის პროტოკოლი
WIRP	Wireless Internet Routing Protocol; უსადენო ინტერნეტ მარშრუტიზაციის პროტოკოლი

WLAN	Wireless Local Area Network; სადენო ლოკალური ქსელი
WRP	Wireless Routing Protocol; უსადენო მარშრუტიზაციის პროტოკოლი
Wormhole Attack	თავდასხმა ჭიის ხვრელი
X.25	OSI ქსელური მოდელის არხული დონის პროტოკოლთა ჯგუფი
ZRP	Zone Routing Protocol; ზონური მარშრუტიზაციის პროტოკოლი

შესავალი

უსადენო ქსელები სწრაფად იქცა ჩვენი ცხოვრების აუცილებელ ნაწილად. ამის ნათელი დადასტურებაა მსგავსი ქსელების ფართოდ გამოყენება სხვადასხვა სფეროებში, იქნება ეს ოფისი, ბინა, უნივერსიტეტი, აეროპორტები, სასტუმროები თუ სხვა. თუმცა უსადენო ქსელების უსაფრთხოების საკითხები სერიოზულ ბარიერს წარმოადგენს მათი ფართოდ დანერგვისათვის. უსადენო ქსელები, ჩვეულებრივ, არ არის დამოკიდებული საოფისე გარემოში არსებულ ტრადიციულ ინფრასტრუქტურაზე, როგორიც არის დენის წყარო, მაღალი გამტარუნარიანობა, მუდმივი კავშირი, საერთო ქსელური სერვისი, სტატიკური კონფიგურაცია, სისტემის ადმინისტრირება და ფიზიკური უსაფრთხოება. ადექვატური უსაფრთხოების გარეშე საწარმოები უარს იტყვიან უსადენო ქსელების გამოყენებაზე, სამთავრობო უწყებები აკრძალავენ უსადენო ქსელების გამოყენებას, თავდაცვის ორგანიზაციებმა შესაძლოა ვერ უზრუნველყონ საკუთარი თანამშრომლების უსაფრთხოება ბრძოლის ველზე, ხოლო მომხმარებელი პასუხისმგებელი გახდეს ქმედებისთვის, რომელიც არასოდეს ჩაუდენია. შესაბამისად, მსგავსი უსადენო ქსელების უსაფრთხოება მნიშვნელოვანი საკითხია, რაზეც უნდა გამახვილდეს ყურადღება, თუკი ასეთი ქსელების გამოყენება ფართოდ დაინერგება.

წინამდებარე ნაშრომში ყურადღება გამახვილებულია უსადენო ქსელების ინფორმაციულ უსაფრთხოებაზე. უსადენო ქსელების უსაფრთხოების თემა საკმაოდ ფართოა და მოიცავს ისეთ სფეროებს, როგორიცაა ქსელის პროტოკოლების, უსადენო მოწყობილობების. ოპერაციული სისტემების და ა.შ. უსაფრთხოება.

ნაშრომში განხილულია უსადენო ქსელების პროტოკოლების უსაფრთხოება და კერძოდ მარშრუტიზაციის OLSR პროტოკოლოს უსაფრთხოება. OLSR პროტოკოლი შემუშავებული იქნა HIPERCOM პროექტის ჯგუფის მიერ კომპიუტერული მეცნიერების და კონტროლის ეროვნულ კვლევით ინსტიტუტში (INRIA), რომელიც მდებარეობს როსქუენქორტში, საფრანგეთში. OLSR პროტოკოლი არ იქნა შემუშავებული უსაფრთხოების პრინციპების გათვალისწინებით. შესაბამისად ადვილია ისეთი გზების მონახვა, რომლებიც ხელს შეუშლის ამ პროტოკოლის გამართულად ფუნქციონირებას. ამ ნაშრომის ძირითადი

მიზანია შესაძლებელი თავდასხმების გამოკვლევა და OLSR პროტოკოლის უსაფრთხოების სხვადასხვა გზების შემუშავება.

სადისერტაციო ნაშრომი შედგება შესავლისა და ოთხი თავისაგან.

პირველ თავში აღწერილია უსადენო ქსელები და მათი თავისებურებანი. აღწერილია შესაძლო საფრთხეები და ამავე თავში დახასიათებულია ის თავდასხმები, რომლებსაც შეიძლება ადგილი ჰქონდეს უსადენო ქსელებში. გარდა ამისა განხილულია კრიპტოგრაფიის მეთოდები, როგორც თავდაცვის ერთ-ერთი შესაძლო მეთოდი.

მეორე თავში განხილულია უსადენო ქსელებში მარშრუტიზაციის რეალიზაციის საკითხები. ამასთან დაკავშირებით აღნიშნულია ის მომატებული საფრთხეები, და ის მიზეზები, რომლებიც განაპირობებენ მომატებულ საშიშროებას უსადენო ქსელებში. დახასიათებულია არსებული მარშრუტიზაციის პროტოკოლები და მოყვანილია მათზე შესაძლო თავდასხმები.

მესამე თავში განხილულია უსადენო ქსელებისათვის შემუშავებული OLSR მარშრუტიზაციის პროტოკოლი. აღწერილია მისი ფუნქციები და შეტყობინებები, მოცემულია მათი ფორმატი.

მეოთხე თავში განხილულია OLSR-პროტოკოლის უსაფრთხოების ამღლების საკითხები. ზოგადად დახასიათებულია არსებული უსაფრთხოების ამაღლების მეთოდები, აღნიშნულია, რომ რეპუტაციის მექანიზმი შეიძლება იყოს გამოყენებული როგორც ერთ-ერთი საიმედო მიდგომა უსაფრთხოებოს უზრუნველსაყოფად. გამოყვანილია რეპუტაციის ფუნქციონალური დამოკიდებულება რეიტინგებზე. ამასთან ერთად მოყვანილია OLSR პროტოკოლის არსებული გაფართოვება შემუშავებული მისი საიმედობის ამაღლების მიზნით. წარმოდგენილია აღნიშნული გაფართოვების შემუშავებული უსაფრთხოების მოდიფიცირებული ალგორითმი.

და ბოლოს დასკვნებში აღნიშნულია, რომ სადისერტაციო ნაშრომში მიღებულია შემდეგი შედეგები:

- დასაბუთებულია, რომ რეპუტაციის კონცეფციაზე დაფუძნებული რეიტინგების სისტემა იძლევა უსაფრთხოების უზრუნველყოფის უფრო სრულყოფილ შესაძლებლობებს;
- შემუშავებული ალგორითმები უფრო სანდოს ხდის თავად შემოწმებას, თავიდან გვაცილებს გადაადგილების შეცდომების გენერირებას და ეფექტურია სხვადასხვა თავდასხმების წინააღმდეგ;

- გააჩნია მოქნილი დასჯისა და დაჯილდოვების მექანიზმები, რომლებიც უზრუნველყოფენ კვანძების უფრო ეფექტურ მუშაობას;
- ეკონომიურია ჭარბი ინფორმაციის გადაცემის და გამოყენებული რესურსების თვალსაზრისით;
- იცავს სწორად მომუშავე კვანძების უსაფუძვლოდ დადანაშაულებისაგან.

დანართში წარმოდგენილია ექსპერიმენტული ნაწილი, რომელიც ადასტურებს შემუშავებული მოდიფიცირებული ალგორითმის ეფექტურობას და რომელიც შედგება ორი ნაწილისგან: პირველ ნაწილში ნაჩვენებია მოდელირების დროს გამოყენებული კოდი, ხოლო მეორე ნაწილში – მოცემულია კვანძების გადაადგილების სცენარი.

1. უსადენო ქსელების უსაფრთხოების საკითხები

1.1 უსადენო ქსელები

დღეისათვის უსადენო ტექნოლოგიები ფართოდ გამოიყენება მთელ სამყაროში, რათა დააკმაყოფილოს მომხმარებელთა დიდი რაოდენობის საკომუნიკაციო მოთხოვნები. უნდა აღინიშნოს, რომ მსოფლიოს ზოგ ნაწილში უსადენო ტექნოლოგიები უფროა გავრცელებული, ვიდრე ტრადიციული კაბელური კავშირის ტექნოლოგიები. უსადენო ტექნოლოგიების არსებულ პოპულარობას რამდენიმე მიზეზი გააჩნია. უსადენო მოწყობილობების ფასი საგრძნობლად შემცირდა, რაც მომსახურების პროვაიდერს საშუალებას აძლევს მნიშვნელოვნად შეამციროს უსადენო სერვისის ღირებულება და იგი მეტად მისაწვდომი გახადოს მომხმარებლისთვის. მზარდ ბაზრებზე უსადენო ქსელების ინსტალაციის ღირებულება ასევე შემცირდა და უფრო ნაკლებია, ვიდრე კაბელური ქსელის დამონტაჟების ღირებულება, [1].

ერთ-ერთ ფართოდ გავრცელებულ უსადენო ტექნოლოგიას წარმოადგენს IEEE 802.11–ზე დაფუძნებული უსადენო ადგილობრივი ქსელი (WLAN), რომელსაც ასევე Wi-Fi-ს უწოდებენ, [2]. იგი უმეტესწილად გამოიყენება პერსონალურ კომპიუტერებსა და ლეპტოპებს შორის მონაცემთა უსადენოდ გადაცემისთვის შენობებს შიგნით. ფიჭურ ქსელთან შედარებით მოცემული ტექნოლოგია მოწყობილობებს საშუალებას აძლევს კავშირი პოტენციურად ძალიან მაღალი სიჩქარით დაამყარონ (მაგრამ შედარებით მოკლე მანძილებზე). ფაქტიურად ამ ქსელებს WLAN (Wireless Local Area Network) უსადენო ლოკალურ ქსელებს უწოდებენ, რადგან ისინი LAN-კავშირის ექვივალენტს უზრუნველყოფენ შენობებს შიგნით.

აქამდე განხილული უსადენო ქსელები მობილური კვანძების ერთმანეთთან დასაკავშირებლად დამოკიდებული არიან ფიქსირებულ კვანძებზე (რადიო ანძები და მიმღებ-გადამცემები). გარდა ამისა, მოცემული ქსელებისთვის აუცილებელია გარკვეული სახის ფიქსირებული ინფრასტრუქტურა, რათა ფიქსირებული კვანძები ერთმანეთს დააკავშიროს. კავშირგაბმულობის ისეთი ქსელის არსებობა, რომელიც ფიქსირებულ ინფრასტრუქტურას ეყრდნობა, ყველა სახის გამოყენებითი პროგრამისთვის მისაღები არ არის. უკანასკნელ წლებში შემოთავაზებულ იქნა ახალი უსადენო არქიტექტურა, რომელიც არ საჭიროებს არანაირ ფიქსირებულ

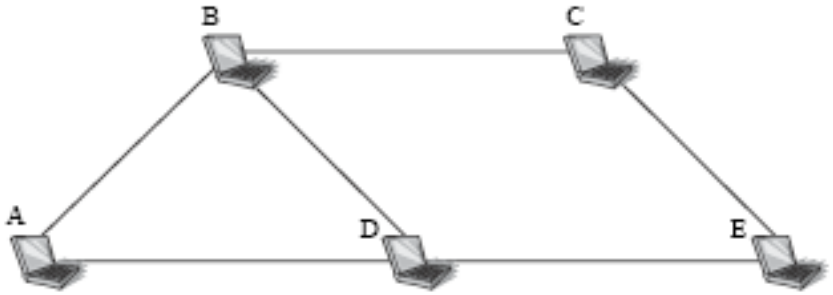
ინფრასტრუქტურას. ამ არქიტექტურის მიხედვით შესაძლოა, რომ ყველა კვანძი მობილური იყოს და არც ერთი მათგანი არ ასრულებდეს განსაკუთრებულ როლს. ასეთი არქიტექტურის ერთ მაგალითს წარმოადგენს მიზნობრივი არქიტექტურის მოდელი 802.11. ამ კვანძებს ერთმანეთთან კავშირისთვის უსადენო ქსელის მიმღებ-გადამცემები არ სჭირდებათ. კვანძების დასაკავშირებლად აუცილებელი კვანძი მისაწვდომი ხდება მეზობელი კვანძის მეშვეობით. ერთმანეთთან ახლოს მყოფი კვანძები მეზობლებს აღმოაჩენენ. როდესაც კვანძს სხვა კვანძთან კავშირი სჭირდება, იგი აგზავნის მონაცემებს მეზობლებისკენ, ეს უკანასკნელები კი, თავის მხრივ, გადასცემენ მონაცემებს მათ მეზობლებს და ა.შ. აღნიშნული მანამ გრძელდება, სანამ ამ მონაცემთა მიმღები იქნება მიღწეული. მსგავსი არქიტექტურა მოითხოვს, რომ ქსელის ყოველი კვანძი ასრულებდეს მარშრუტიზატორის როლს, რისთვისაც უნდა შეეძლოს მიმართულების განსაზღვრა, რაც პაკეტებს სჭირდებათ დანიშნულების პუნქტის მისაღწევად , [3].

არც თუ დიდი ხანია, რაც სახლისა და მცირე ოფისის ქსელებმა და მცირე სივრცეებში კომპიუტერიზაციამ ლეპტოპების გამოყენებით (მაგალითად, კონფერენცია საკლასო ოთახში, ცალკე მდგომ შენობაში და ა.შ.) სწრაფად მოიპოვა პოპულარობა, ისევე, როგორც გამოყენების სხვა ძირითადმა სფეროებმა. ეს მოიცავს კომერციულ გამოყენებით პროგრამებს, რომლებიც დაფუძნებულია ისეთ პროგრესულად განვითარებად ტექნოლოგიებზე, როგორიც არის Bluetooth, [4], WiMAX [5,6], WiFi, და ა.შ. გარდა ამისა, ადამიანები თავიდანვე მიხვდნენ, რომ მიზნობრივ ქსელს აშკარა პოტენციალი აქვს მონაცემთა მობილური კომპიუტერიზაციის ყველა ტრადიციულ სფეროში.

იმის გამო, რომ უსადენო ქსელების გამოყენება ხდება სამხედრო ან პოლიციური დანიშნულებით, განსაკუთრებით კი მზარდი კომერციული მიზნებისთვის, უსაფრთხოების სხვადასხვა საკითხები განხილვას საჭიროებენ. ამ საკითხების დაბალი დანახარჯებით გადაწყვეტა სათანადო დონეზე არსებითია უსადენო ქსელების ფართოდ გამოყენებისთვის.

უსადენო ქსელის ძირითადი სამუშაო პრინციპების აღწერა მოცემულია სურ.1.1, სადაც ასახულია მრავალბიჯიანი ქსელი. კვანძი A უშუალოდ ამყარებს კავშირს ასეთივე კვანძთან B (ერთი ბიჯი), როდესაც მათ შორის არსებობს კავშირი შესაბამისი მახასიათებლებით. საპირისპიროდ ამისა, აუცილებელია მრავალბიჯიანი კავშირი, როდესაც კავშირის დამამყარებელ კვანძებს შორის ერთმა

ან მეტმა კვანძმა მარშრუტიზატორის როლი უნდა შეასრულოს. მაგალითად, სურ.1.1 არ არსებობს არანაირი კავშირი კვანძებს A და C ან A და E შორის. შესაბამისად, კვანძებმა B და D კვანძებს A და C ან A და E კავშირის დასამყარებლად შუალედური მარშრუტიზატორის როლი უნდა შეასრულონ. და მართლაც, უსადენო ქსელების გამორჩეულ მახასიათებელს ის წარმოადგენს, რომ ყველა კვანძს შეუძლია, მოთხოვნისამებრ, მარშრუტიზატორის როლი შეასრულოს. პაკეტების მარშრუტის შესარჩევად უსაზღვროდ გრძელი მარშრუტების ძიებისგან დასაცავად, აშკარა და არსებით მოთხოვნას წარმოადგენს, რომ მარშრუტი ციკლურობისგან თავისუფალი იყოს. [7].



სურ.1.1 უსადენო ქსელის მაგალითი.

1.2 საფრთხეები, თავდასხმები და ხარვეზები

ნებისმიერ სისტემას, რომელსაც დაცვა სჭირდება, გააჩნია სისუსტეები ან ხარვეზები, რომელთა ნაწილს ან ყველას ერთად ამოირჩევს თავდამსმხმელი ობიექტად. შესაბამისად, სისტემის უსაფრთხოების მექანიზმების შექმნის ერთ-ერთ მიდგომას წარმოადგენს განხილვა იმ საფრთხეებისა და სავარაუდო თავდასხმებისა, რომელთა წინაშე დგას სისტემა, იმის გათვალისწინებით, რომ სისტემას ხარვეზები გააჩნია, [8, 9]. უსაფრთხოების მექანიზმებმა უნდა უზრუნველყონ სისტემის უსაფრთხოება მოცემული საფრთხეების, თავდასხმებისა და ხარვეზების გათვალისწინებით. სანამ განვიხილავდეთ უსაფრთხოების მექანიზმებს, შექმნილს უსადენო ქსელებში სხვადასხვა მიზნების მისაღწევად, განვიხილოთ საფრთხეები, თავდასხმები და ხარვეზები. პირველ რიგში განვსაზღვროთ ეს ცნებები: საფრთხე, ხარვეზი და თავდასხმა, [10-12]. შეგვიძლია მოვიყვანოთ შემდეგი განმარტებები:

- ყველა მეთოდი ან საგანი, რაც გამოიყენება სისტემის, მოწყობილობის ან მუშაობის სისუსტით სარგებლობისთვის, საფრთხეს წარმოადგენს. საფრთხეთა მაგალითები მოიცავს ჰაკერებს, განაწყენებულ თანამშრომლებს, საწარმოო შპიონაჟსა და კრიმინალურ ორგანიზაციებს;

- ხარვეზი არის რაიმე მოწყობილობა, ან პროგრამული უზრუნველყოფა, რაც ინფორმაციას ღიად ტოვებს მისით სარგებლობისთვის. სარგებლობას შესაძლოა სხვადასხვა სახე ჰქონდეს. ეს შეიძლება იყოს ინფორმაციის არაუფლებამოსილი მისაწვდომობა ან მონაცემთა დამუშავების ხელყოფა;

- თავდასხმა წარმოადგენს კომპიუტერის უსაფრთხოების კონტროლის მექანიზმების გვერდის ავლის მცდელობას. თავდასხმის შედეგად შესაძლოა შეიცვალოს, მოპარულ იქნას ან გაუქმდეს მონაცემი. თავდასხმათა მაგალითებს მიეკუთვნება მონაცემების მოპარვა გადასაცემი გარემოდან და მოწყობილობებიდან, უკანონო პრივილეგიების მიღება, მონაცემების მცდარად შეტანა, ინფორმაციის მოდიფიცირება, ქსელის ნაკადის ანალიზირება, და ა.შ. თავდასხმები ორ ძირითად კატეგორიად იყოფა:

- პასიური თავდასხმა – ასეთი თავდასხმისას თავდამსხმელი პასიურად აკვირდება პაკეტების ან ფრეიმების გაცვლას უსადენო სივრცეში, რისთვისაც საჰაერო ტალღებს აკვირდება. რადგან თავდამსხმელი მხოლოდ აკვირდება გადაცემად პაკეტებს და არ ახდენს მათ მოდიფიცირებას ან დაზიანებას, ასეთი თავდასხმისას უმთავრეს სამიზნეს სისტემის კონფიდენციალობა წარმოადგენს.

თუმცა, უნდა აღინიშნოს, რომ ინფორმაციის მოგროვების ასეთმა პროცესმა მოგვიანებით შესაძლოა უფრო აქტიური თავდასხმები გამოიწვიოს. როგორც წესი, ასეთი თავდასხმის წამოწყება უფრო ადვილია, ვიდრე თავდასხმების ქვემოთ განხილული სახეობებისა;

- აქტიური თავდასხმა – ეს არის თავდასხმა, რომლის დროს თავდამსხმელი მტრულ ქმედებას ახორციელებს, გარდა იმისა, რომ პასიურად აკვირდება ნაკადს (ტრაფიკს). მაგალითად, თავდამსხმელმა შესაძლოა გადაწყვიტოს პაკეტების მოდიფიცირება, ჩამატება ან სულაც ქსელური სერვისის დაზიანება.

უსადენო ქსელების უსაფრთხოება საგრძნობლად განსხვავდება მათი კაბელიანი ანალოგის უსაფრთხოებისგან, რის მიზეზსაც ფიზიკური გარემოს ბუნება წარმოადგენს. უსადენო გარემოში კავშირისას გადაცემული და მიღებული სიგნალები ჰაერში მოგზაურობს. შესაბამისად, ნებისმიერ კვანძს, რომელიც

გამგზავნი კვანძის გადაცემის დიაპაზონში მდებარეობს და იცის საოპერაციო სიხშირე და სხვა ფიზიკური დონის ატრიბუტები (მოდულაცია, კოდირება და ა.შ.), პოტენციურად შეუძლია სიგნალის გაშიფრვა იმგვარად, რომ გამგზავნს ან სავარაუდო მიმღებს არაფერი ეცოდინება აღნიშნული შეჭრის შესახებ. საპირისპიროდ ამისა, საკაბელო ქსელებში მსგავს შეჭრას ადგილი შესაძლოა ჰქონდეს მხოლოდ იმ შემთხვევისას, თუ თავდამსხმელისთვის მისაწვდომი გახდება გადაცემის ფიზიკური საშუალება (სადენი, ბოჭკო და ა.შ.), რისთვისაც, როგორც წესი, აუცილებელია ასეთ საშუალებასთან მიერთება.

უსადენო ქსელების დაცვის კიდევ ერთ პრობლემას წარმოადგენს ის, რომ უსაფრთხოების არსებული ტექნოლოგიები უმეტესწილად საკაბელო ქსელებზეა ორიენტირებული, რომლებიც მეტ-ნაკლებად სტატიკურია. არსებული ტექნოლოგიები ხშირად დამოკიდებულია ნაკადის ფილტრებზე, რომლებსაც ნაკადის უმეტესი ნაწილი გაივლის. ასეთ ფილტრულ წერტილებში განთავსებულ უსაფრთხოების მოწყობილობებს შეუძლიათ ნაკადის ინსპექტირება საეჭვო ქმედებების კუთხით, უსაფრთხოების პოლიტიკის შემუშავება და შესაბამისად რეაგირება. მაგრამ იგივე არ ხდება უსადენო ქსელებში, სადაც ქსელის ობიექტები უმეტესწილად გადაადგილდებიან. უსაფრთხოების ტრადიციული გადაწყვეტილებები ასევე ეფუძნება რამდენიმე ცენტრალურად განთავსებულ მოწყობილობას, რომლებიც ქსელის უსაფრთხოებას მართავს. მსგავსი გადაწყვეტილებები ვერ გამოიყენება უსადენო ქსელებისთვის, გამომდინარე ამ ქსელების მახასიათებლებიდან.

უსადენო ქსელები, რომლებიც ფართოდ გამოიყენებენ უსადენო ლინკებს, ქსელისათვის დამახასიათებელი ბუნებიდან გამომდინარე, დაუცველია სხვადასხვა სახის თავდასხმებისადმი. ისეთი მექანიზმები, როგორიც არის კოდირება და აუთენტიფიკაცია, საგრძნობლად ნიღბავს მსგავს საფრთხეს, მაგრამ ეს უსადენო ქსელში ერთადერთი საფრთხე არ არის.

ვინაიდან უსადენო ქსელები არ არის დამოკიდებული ინფრასტრუქტურაზე დაფუძნებულ რესურსებზე, როგორიცაა დენის სტაბილური წყარო, მაღალი სიხშირე, უწყვეტი კავშირი ან უცვლელი მარშრუტიზირება, მათ მიმართ თავდასხმების განხორციელება საკმაოდ ადვილია.

1.3 უსაფრთხოების ძირითადი კონცეფციები

უსაფრთხო სისტემა შეიძლება განისაზღვროს, როგორც სისტემა, რომელიც ზუსტად იმას აკეთებს, რაც მის შემქმნელებს აქვთ ჩაფიქრებული და მისთვის არ არის დამახასიათებელი მოულოდნელი ქმედებები, მაშინაც კი, როდესაც თავდამსხმელი ცდილობს, რომ სისტემა განსხვავებულად მოქმედებდეს, [11].

უსაფრთხოების განმარტება არასრული იქნებოდა იმის განსაზღვრის გარეშე, თუ ვისგან ან რისგან არის სისტემა დაცული. უფრო მეტიც: ვინაიდან აბსოლუტური უსაფრთხოების მიღწევა შეუძლებელია, უნდა გაკეთდეს ანგარიში ხარჯებისა და სარგებლის ბალანსის შესახებ. უნდა გვახსოვდეს, რომ უსაფრთხოება მოითხოვს, რომ დამცველმა მოიცვას შესაძლო თავდასხმის ყველა ასპექტი, როდესაც თავდამსხმელისათვის, წარმატების მისაღწევად, საკმარისია ძალისხმევის მიმართვა ერთი სუსტი წერტილისკენ. შესაბამისად, სისტემა იმდენად უსაფრთხოა, რამდენადაც უსაფრთხოა მისი ყველაზე ნაკლებად სანდო წერტილი.

1.3.1 უსაფრთხოების სერვისი

უსაფრთხოების სერვისი არის დამუშავების ან კომუნიკაციის სერვისი, რაც აუმჯობესებს ორგანიზაციის მიერ მონაცემთა დამუშავების სისტემებს და ინფორმაციის გადაცემას. სერვისმა, სავარაუდოდ, უსაფრთხოებაზე თავდასხმა უნდა დაძლიოს, რის გამოც ფუნქციონირებისთვის უსაფრთხოების ერთ ან მეტ მექანიზმს იყენებს.

ITU-T რეკომენდაციის X.800 უსაფრთხოების არქიტექტურა OSI მოწყობილობებისთვის ამ სერვისებს ყოფს ხუთ კატეგორიად და თოთხმეტ სპეციფიურ სერვისად. ცხრილი 1.1, [13].

ცხრილი 1.1 უსაფრთხოების სერვისები, X.800, [13].

აუთენტიფიკაცია
დარწმუნება იმაში, რომ საკომუნიკაციო ობიექტი ის არის, რადაც ასაღებს თავს.
იდენტური ობიექტის აუთენტიფიკაცია
გამოიყენება ლოგიკურ კავშირთან მიმართებით, რათა უზრუნველყოფილ იქნას დაკავშირებული ობიექტების იდენტურობა.
მონაცემთა წარმოშობის აუთენტიფიკაცია

უკავშირო გადაცემისას უზრუნველყოფს იმას, რომ მიღებული მონაცემის წყარო ის იყოს, რადაც ასაღებს თავს.

შეღწევის კონტროლი

წყაროს არაუფლებამოსილი გამოყენების პრევენცია (მაგალითად, ეს სერვისი კონტროლს უწევს იმას, თუ ვისთვის არის მისაწვდომი რესურსი, და რისი გაკეთების უფლება აქვთ მათ, ვისთვისაც ეს რესურსი მისაწვდომია).

მონაცემთა კონფიდენციალობა

მონაცემთა დაცვა არაუფლებამოსილი გაცემისგან.

კავშირის კონფიდენციალობა

მომხმარებლის ყველა მონაცემის დაცვა კავშირისას.

უკავშირო მონაცემთა კონფიდენციალობა

მონაცემთა ცალკეულ ბლოკში მომხმარებლის ყველა მონაცემის დაცვა.

შერჩეული ველის კონფიდენციალობა

მომხმარებელთა მონაცემებში შერჩეული ველის კონფიდენციალობა კავშირისას ან მონაცემთა ცალკეულ ბლოკში.

ნაკადის (ტრაფიკის) კონფიდენციალობა

დაცვა ინფორმაციისა, რომლის მიღება შესაძლებელია ტრაფიკზე დაკვირვებით.

მონაცემთა მთლიანობა

რწმენა იმისა, რომ მიღებული მონაცემი ზუსტად ის არის, რაც ავტორიზებულმა ობიექტმა გაგზავნა (მაგალითად, ადგილი არ ჰქონია მოდიფიცირებას, ჩანაცვლებას, წაშლას ან განმეორებას).

კავშირის მთლიანობა აღდგენით

უზრუნველყოფს მომხმარებლის ყველა მონაცემის მთლიანობას კავშირის დროს და იდენტიფიცირებას უკეთებს მონაცემის ნებისმიერ მოდიფიცირებას, ჩანაცვლებას, წაშლას ან განმეორებას მონაცემთა მიმდევრობაში და ცდილობს მონაცემთა აღდგენას.

კავშირის მთლიანობა აღდგენის გარეშე

ზემოთგანხილულის იდენტურია, მაგრამ უზრუნველყოფს მხოლოდ იდენტიფიცირებას, აღდგენის გარეშე.

შერჩეული ველის კავშირის მთლიანობა

უზრუნველყოფს შერჩეული ველის მთლიანობას, რომელიც მოთავსებულია კავშირისას გადაცემული მონაცემთა ბლოკის მომხმარებლის მონაცემში. განსაზღვრავს იმას, თუ შერჩეული ველი იყო თუ არა მოდიფიცირებული, ჩანაცვლებული, წაშლილი ან განმეორებული.

უკავშირო მთლიანობა

უზრუნველყოფს ცალკეული უკავშირო ბლოკის მთლიანობას და განსაზღვრავს მონაცემთა მოდიფიცირების ფაქტს. **შერჩეული ველის უკავშირო მთლიანობა** უზრუნველყოფს ცალკეული უკავშირო ბლოკის შერჩეული ველების მთლიანობას. განსაზღვრავს, იყო თუ არა შერჩეული ველი მოდიფიცირებული.
უარყოფა უზრუნველყოფს დაცვას კავშირში ჩართული ერთ-ერთი ობიექტის მიერ უარყოფისგან, რომელიც მონაწილეობდა მთელს კავშირსა ან მის ნაწილში. **ავტორობაზე უარის თქმის შეუძლებლობა, წარმოშობის წყარო** დადასტურება იმისა, რომ მოცემული გზავნილი გადაცემულ იქნა აღნიშნული მხარის მიერ. **ავტორობაზე უარის თქმის შეუძლებლობა, დანიშნულების პუნქტი** დადასტურება იმისა, რომ მოცემული გზავნილი მიღებულ იქნა აღნიშნული მხარის მიერ.

1.3.2 უსაფრთხოების მექანიზმები

უსაფრთხოების მექანიზმი არის პროცესი (ან მოწყობილობა, რომელიც ჩართულია ამ პროცესში), რომელიც შექმნილია, რათა აღმოაჩინოს, დაიცვას ან აღადგინოს მონაცემი უსაფრთხოების სისტემაზე თავდასხმის შემდეგ, [11, 14-16]. ცხრილში 1.2 ჩამოთვლილია უსაფთხოების მექანიზმები, განსაზღვრული X.800-ით. როგორც ვხედავთ, მექანიზმები დაყოფილია ისეთ მექანიზმებად, რომლებიც სრულდება კონკრეტული პროტოკოლის დონეზე და ისეთებად, რომლებიც არ არის დამახასიათებელი რომელიმე კონკრეტული პროტოკოლის დონის ან უსაფრთხოების სერვისისთვის.

ცხრილი 1.2. უსაფრთხოების მექანიზმები, X.800, [13].

უსაფრთხოების კონკრეტული მექანიზმები ეს მექანიზმები შესაძლოა ჩართულ იქნას შესაბამისი პროტოკოლის დონეზე, რათა უზრუნველყოს ზოგიერთი OSI (Open Systems Interconnection) უსაფრთხოების სერვისი. **დაშიფრვა** მათემატიკური ალგორითმების გამოყენება ინფორმაციის ტრანსფორმირებისათვის

ფორმად, რომელიც ჯერ არ არის აღქმადი. მონაცემთა ტრანსფორმირება და შემდგომი აღდგენა დამოკიდებულია ალგორითმსა და დაშიფრვის სხვა კოდებზე.

ციფრული ხელმოწერა

ციფრული ხელმოწერა არის მონაცემის კრიპტოგრაფიული ტრანსფორმირება, რაც საშუალებას იძლევა, რომ მონაცემის მიმღებმა დაადასტუროს მონაცემის წყარო და მთლიანობა და დაიცვას გაყალბებისგან (მაგალითად, მიმღების მიერ).

მისაწვდომობის კონტროლი

სხვადასხვა მექანიზმები, რომლებიც ახორციელებენ უფლებას რესურსების მისაწვდომობისა.

მონაცემთა მთლიანობა

სხვადასხვა მექანიზმები, რომლებიც გამოიყენება მონაცემის ერთეულის ან მონაცემის ერთეულთა ნაკადის მთლიანობის უზრუნველსაყოფად.

აუთენტიფიკაციის გაცვლა

მექანიზმი, რომელიც გამოიყენება ობიექტის იდენტურობის უზრუნველსაყოფად ინფორმაციის გაცვლის საშუალებით.

ნაკადის (ტრაფიკის) გადავსება

ბიტების ჩასმა მონაცემთა ნაკადის შუალედებში ნაკადის ანალიზის მცდელობის დასაბრკოლებლად.

მარშრუტიზაციის კონტროლი

იძლევა საშუალებას, რომ მოცემული მონაცემისთვის შერჩეულ იქნას კონკრეტული, ფიზიკურად უსაფრთხო მარშრუტები და ეს მარშრუტები შეიცვალოს იმ დროს, როდესაც უსაფრთხოების სისტემის დაზიანებაა მოსალოდნელი.

დადასტურება

სანდო მესამე მხარის გამოყენება მონაცემთა გაცვლის კონკრეტული მახასიათებლების უზრუნველსაყოფად.

უსაფრთხოების შემავსებელი მექანიზმები

მექანიზმები, რომლებიც არ არის დამახასიათებელი კონკრეტული OSI უსაფრთხოების სერვისის ან პროტოკოლის დონისთვის.

სანდო ფუნქციონალურობა

გარკვეული კრიტერიუმის შესაბამისად მუშაობა (მაგალითად, უსაფრთხოების პოლიტიკის შესაბამისად).

უსაფრთხოების იარლიყი

რესურსის (რაც შესაძლოა მონაცემის ერთეულს წარმოადგენდეს) მარკირება, რაც გულისხმობს რესურსისთვის სახელის დარქმევას ან განსაზღვრავს მოცემული რესურსის უსაფრთხოების ატრიბუტებს.

ღონისძიებების განსაზღვრა

უსაფრთხოების კუთხით მნიშვნელოვანი ღონისძიებების განსაზღვრა.

უსაფრთხოების აუდიტი

მონაცემები, რომლებიც მოგროვებულია და, სავარაუდოდ, გამოიყენება უსაფრთხოების აუდიტის ხელშეწყობისთვის, რაც სისტემის ჩანაწერებისა და ქმედებების დამოუკიდებელ კვლევას და შემოწმებას წარმოადგენს.

უსაფრთხოების აღდგენა

ამუშავებს მოთხოვნებს ისეთი მექანიზმებიდან, როგორიც არის მონაცემთა დამუშავება და მენეჯმენტის ფუნქციები, და აგრეთვე ახორციელებს აღდგენით ქმედებებს.

1.3.3 საფრთხეები და თავდასხმები

უსადენო ქსელები დაუცველია არა მხოლოდ გარე, არამედ შიდა თავდასხმების მიმართაც. უსადენო ქსელები შეიძლება გახდეს ორი განსხვავებული დონის თავდასხმის ობიექტი. თავდასხმის პირველ დონეზე მოწინააღმდეგე ცდილობს უსადენო ქსელის ძირითადი მექანიზმების, როგორიც მაგალითად, არის მარშრუტიზირება, გატეხვას, რაც არსებითია ქსელის სათანადო ფუქნციონირებისთვის, ხოლო თავდასხმის მეორე დონეზე მოწინააღმდეგე უკვე ცდილობს დააზიანოს ქსელის მიერ გამოყენებული მექანიზმები, როგორიც არის მენეჯმენტის საკვანძო სქემები ან გამოყენებადი კრიპტოგრაფიული ალგორითმები. აღნიშნული შეიძლება განვიხილოთ თავდასხმების კლასიფიცირების ერთ გზად. ამასთან ერთად, თავდასხმა შეიძლება იყოს, როგოც აქტიური, ასევე პასიური.

თავდასხმები ასევე კლასიფიცირდება თავდამსხმელის მიერ გამოყენებული საშუალებების მიხედვით. მაგალითად, თავდასხმა, წამოწყებული დისტანციურად დაშორებული მოწინააღმდეგის მიერ, კლასიფიცირდება, როგორც გარე თავდასხმა, როდესაც შეტევა, განხორციელებული ერთ-ერთი კვანძის მიერ, რომელიც ქსელის ნაწილს წარმოადგენს, განხილული იქნება, როგორც შიდა თავდასხმა. გარე თავდასხმები, როგორც წესი, აქტიურთა რიგს განეკუთვნება, რომელთა მიზანს ქსელის გადატვირთვა, არასწორი გადასაცემი ინფორმაციის გამრავლება, სერვისის სათანადოდ მუშაობისთვის ხელის შეშლა ან ქსელის სრულად გათიშვა წარმოადგენს. გარე თავდასხმებისგან თავდაცვა შესაძლებელია უსაფრთხოების სტანდარტული მექანიზმების გამოყენებით, როგორიც არის ქსელთაშორისი ეკრანი (firewall), დაშიფრვა და კრიპტოგრაფიაზე დაფუძნებული სხვა ალგორითმები და

ა.შ. შიდა თავდასხმები, როგორც წესი, უფრო მძლავრია, რადგან მტრულად განწყობილი შიდა კვანძი უკვე ეკუთვნის ქსელს, როგორც უფლებამოსილი მხარე და, შესაბამისად, დაცულია ქსელისა და მისი სერვისების უსაფრთხოების მექანიზმების მიერ. ამდენად, ასეთ მტრულად განწყობილ შიდა კვანძებს, რომლებიც შესაძლოა ჯგუფშიც კი მუშაობდნენ, თავადვე შეუძლიათ უსაფრთხოების სტანდარტული საშუალებების გამოყენება საკუთარი თავდასხმების დასაცავად.

1.3.4 სერვისის მტყუნება (DoS)

ეს არის საფრთხე სერვისის გათიშვისა, რის მიზეზსაც შესაძლოა წარმოადგენდეს, როგორც განუზრახველი ქმედება, ასევე მტრული აქტი, [17]. იგი ნებისმიერი სისტემის უსაფრთხოებისათვის დიდ რისკს წარმოადგენს, [1]. DoS თავდასხმის ორგანიზების კლასიკურ გზას ცენტრალიზებული რესურსის იმგვარი გადავსება წარმოადგენს, რომ მან სათანადოდ ვეღარ იმუშაოს ან სულაც გაითიშოს. თუმცა უსადენო ქსელებში ეს შეიძლება არ იყოს ეფექტური მიდგომა, გამომდინარე ცენტრალიზებული რესურსის ნაკლებობიდან. გადანაწილებული DoS თავდასხმა კიდევ უფრო დიდი საფრთხეა. თუკი თავდამსხმელებს საკმარისი კომპიუტერიზებული სიმძლავრე და სიხშირე გააჩნიათ, მცირე ზომის უსადენო ქსელი შესაძლოა შედარებით მარტივად გამოვიდეს მწყობრიდან. მტრულად განწყობილმა კვანძებმა შესაძლებელია მოახდინონ პროტოკოლის ან მისი ნაწილის ხელახალი კონფიგურირება, იმგვარად, რომ ინფორმაცია ძალიან იშვიათად გადასცენ, რაც გამოიწვევს ქსელის დახშობას, ხოლო კვანძები ვერ შეძლებენ უახლესი ინფორმაციის მიღებას ქსელის შეცვლილი ტოპოლოგიის შესახებ. თუ მტრულად განწყობილი კვანძებისა და მარშრუტის იდენტიფიცირება ვერ ხერხდება, ქსელი მძიმე შედეგებს მიიღებს, რადგან შეიქმნება შთაბეჭდილება, რომ სხვა კვანძებისთვის ქსელი გამართულად მუშაობს. ქსელის ასეთი სახის გაუმართავი მუშაობა, გამოწვეული მტრულად განწყობილი კვანძების მიერ, ცნობილია ბიზანტიურ ზიანად. მაგალითად, მტრულად განწყობილი კვანძი შესაძლოა მონაწილეობას იღებდეს სესიაში, მაგრამ გამოტოვოს პაკეტების გარკვეული რაოდენობა, რამაც შეიძლება გამოიწვიოს ქსელის მიერ შეთავაზებული მომსახურების ხარისხის დაცემა. სერვისის გათიშვის თავდასხმის ზოგი მაგალითია:

- SYN (Synchronization) გადავსება. ასეთი სახის DoS თავდასხმისას შემტევი კვანძი მსხვერპლ კვანძს უგზავნის დიდი რაოდენობით SYN პაკეტებს, რომელთა უკანა მისამართი გაყალბებულია. SYN პაკეტების მიღებისას მსხვერპლი კვანძი უგზავნის დამადასტურებელ (SYNACK) პაკეტებს კვანძებს, რომელთა მისამართები მიღებულ SYN პაკეტებში იყო მოცემული და ელოდება დადასტურებას (ACK) გამგზავნისგან, რასაც ვერასდროს მიიღებს;

- დახშობა. ასეთი სახის DoS თავდასხმის ინიციატორია მტრულად განწყობილი კვანძი. მას შემდეგ, რაც იგი მოახდენს მიმღების მიერ გამოყენებული კომუნიკაციის სიხშირის განსაზღვრას, იმავე სიხშირეს იყენებს მიმღებისთვის მონაცემების გასაგზავნად, რითაც აბრკოლებს ოპერაციას. ასეთი თავდასხმების გვერდის ავლის გავრცელებული ტექნიკაა სიხშირის ცვალებადობა;

- სერვისის გადანაწილებული მტყუნება. ასეთი სახის თავდასხმას იწყებს მტრულად განწყობილი კვანძების ჯგუფი, რომლებიც იმავე ქსელის ნაწილს წარმოადგენენ და თანხმდებიან გათიშონ ქსელი და სერიოზულად დააზიანონ იგი.

1.3.5 მიბაძვა

მიბაძვის თავდასხმა უსაფრთხოებისთვის სერიოზულ რისკს წარმოადგენს უსადენო ქსელის ყველა დონეზე. თუ არ გამოიყენება მხარეთა სათანადო აუთენტიფიკაცია, რისკის კვანძები შესაძლოა შეუერთდნენ ქსელს, გაგზავნონ მარშრუტიზაციის მცდარი ინფორმაცია და მიბაძონ სხვა, სანდო კვანძებს. რისკის კვანძისთვის შესაძლოა მისაწვდომი გახდეს ქსელის მართვის სისტემა და შესაძლოა იგი შეუდგეს სისტემის კონფიგურაციის შეცვლას, როგორც სუპერ-მომხმარებელი, რომელსაც განსაკუთრებული უფლებები გააჩნია. სერვისის დონეზე შესაძლოა მოხდეს ის, რომ მტრულად განწყობილი მხარის საჯარო გასაღების სერტიფიცირება მოხდეს სათანადო დამოწმების გარეშე. მტრულად განწყობილმა მხარემ შესაძლოა შენიღბოს საკუთარი თავი და წარმოდგეს, როგორც რომელიმე მეგობრული კვანძი და სხვა კვანძებს გადასცეს მცდარი ბრძანებები ან ინფორმაცია სტატუსის შესახებ. მიბაძვის საფრთხის შემცირება ხორციელდება აუთენტიფიკაციის ძლიერი მექანიზმების გამოყენებით, მხარემ უნდა შეძლოს ენდოს წარმომავლობას მონაცემისა, რომელიც მან მიიღო ან შეინახა. უფრო ხშირად აღნიშნული გულისხმობს ციფრული ხელმოწერის ან თითის კოდირებული ანაბეჭდის გამოყენებას გადასაცემ გზავნილებში, კონფიგურაციასა ან სტატუსის

ინფორმაციაში ან სერვისის გამოყენებად და გაცვლად მონაცემში პროტოკოლის იმდენ დონეზე, რამდენზეც ეს შესაძლებელია. კრიპტოგრაფიით გამოყენებული ციფრული ხელმოწერა პრობლემატური საკითხია უსადენო ქსელში, რამდენადაც იგი გასაღების მართვის ეფექტურ და უსაფრთხო სერვისს მოითხოვს და საჭიროებს კომპიუტერიზაციის შედარებით უფრო მძლავრ შესაძლებლობებს.

მიბაძვის ორი, კარგად ცნობილი თავდასხმაა Sybil [18] და Trust.

1.3.5.1 თავდასხმა "სიბილა" Sybil

"სიბილა" თავდასხმისას მტრულად განწყობილი კვანძი ისევე მოქმედებს, თითქოს ისეთივე იყოს, როგორც კვანძების უმეტესობა (ნაცვლად ერთისა), რისთვისაც ემსგავსება სხვა კვანძებს ან უბრალოდ ითვისებს ყალბ საიდენტიფიკაციო მახასიათებლებს. უარესი შემთხვევისას Sybil თავდამსხმელმა შესაძლოა მოახდინოს კვანძის იდენტურობის განმსზღვრელი დამატებითი შემთხვევითი მახასიათებლების გენერირება, რისთვი საც გამოიყენებს მხოლოდ ერთ ფიზიკურ მოწყობილობას. კვანძის მიერ მოპოვებულ დამატებით იდენტურობას Sybil კვანძი ეწოდება. Sybil თავდასხმა შესაძლოა დაწყებულ იქნას სამ განზომილებაში:

(1) პირდაპირი ან არაპირდაპირი კომუნიკაცია. პირდაპირი კომუნიკაცია. ასეთი შემთხვევისას Sybil თავდასხმის განხორციელების ერთი გზაა Sybil კვანძების უშუალო დაკავშირება კანონიერ კვანძებთან. როდესაც კანონიერი კვანძი უგზავნის გზავნილს Sybil კვანძს, ერთ-ერთი მტრულად განწყობილი მოწყობილობა უსმენს მას. ამდენად, Sybil კვანძიდან გაგზავნილი შეტყობინება რეალურად არის შეტყობინება, გაგზავნილი ერთ-ერთი მტრულად განწყობილი მოწყობილობის მიერ.

არაპირდაპირი კომუნიკაცია. ამ ტიპის თავდასხმისას Sybil კვანძსა და კანონიერ კვანძს შორის კავშირი მყარდება არაპირდაპირ, მაგალითად, სხვა მტრულად განწყობილი კვანძის გავლით. სხვა სიტყვებით რომ ვთქვათ, კანონიერ კვანძებს არ აქვთ საშუალება პირდაპირ დაუკავშირდნენ Sybil კვანძებს. Sybil კვანძისთვის გაგზავნილი შეტყობინება გაივლის ერთ-ერთ მტრულად განწყობილ კვანძს, რომელიც გადასცემს მას Sybil კვანძს.

(2) გაყალბებული ან მოპარული იდენტურობა. Sybil კვანძს ორი არჩევანი აქვს საკუთარი თავისთვის იდენტურობის მოპოვებისა. პირველს წარმოადგენს

საკუთარი თავისთვის გაყალბებული იდენტურობის მინიჭება. მეორეს კი წარმოადგენს ლეგიტიმური კვანძის იდენტურობის მოპარვა. პირველი შემთხვევისას Sybil კვანძს იდენტიფიკატორის სახით შეუძლია შექმნას შემთხვევით შერჩეული 32-ბიტიანი მთელი რიცხვი თუკი ქსელის კვანძების 32-ბიტიან იდენტიფიკატორებს იყენებს.

Sybil კვანძმა რამენაირად უნდა მოახერხოს ლეგიტიმური იდენტიფიკაციის მოპოვება, რათა შეძლოს სხვა ლეგიტიმურ კვანძებთან დაკავშირება. ერთ-ერთ გზას მისი მოპარვა წარმოადგენს. იდენტურობის მოპოვების უიოლეს გზას მიმსგავსებული კვანძის იდენტურობის მოპოვება წარმოადგენს, თუკი ასეთი კვანძი ქსელში არსებობს. მოპარულ იდენტურობას ვერ აღმოაჩენენ, თუკი მიმსგავსებული კვანძი გაუქმდება ან დროებით გაითიშება ქსელიდან. თუკი ლეგიტიმურ იდენტურობათა ერთობლიობა რაიმე სახის უსაფრთხოების მექანიზმით არის შეზღუდული, იდნტურობის გაყალბება საკმაოდ გართულდება.

(3) ერთდროულობა. თავდამსხმელმა შესაძლოა განიზრახოს, რომ ქსელში ერთდროულად მისმა ყველა Sybil ერთეულმა მიიღოს მონაწილეობა. როდესაც კონკრეტულ მოწყობილობას შეუძლია იმოქმედოს, როგორც მხოლოდ ერთმა ერთეულმა დროის მოცემულ მომენტში, მას შეუძლია შექმნას შთაბეჭდილება, რომ ყველა მათგანი ერთდროულად არის წარმოდგენილი.

არა-ერთდროულობა. ასეთი ტიპის თავდასხმისას კვანძების ზოგი იდენტურობების გამოყენება ხდება დროის ერთი ინტერვალით, დანარჩენებისა კი დროის სხვა ინტერვალით. აგრეთვე, თუკი თავდამსხმელებს რამდენიმე მტრულად განწყობილი კვანძი აქვთ, ასეთ კვანძებს იდენტურობების შეცვლა პერიოდულად შეუძლიათ და, შესაბამისად, შეუმჩნეველი რჩებიან.

1.3.5.2 Trust (ნდობა) თავდასხმა

Trust თავდასხმა მიბაძვის თავდასხმის კიდევ ერთ სახეობას წარმოადგენს. მარტივი უსაფრთხოების პროგრამებში, სადაც მიზანს მოცემული შეტყობინების ან ობიექტის დაცვა წარმოადგენს აქტიური ან პასიური თავდასხმისგან, მომხმარებლის ნდობა შესაძლოა ჩამოყალიბდეს, როგორც აუთენტიფიკაციის პროცედურა სისტემასა და მომხმარებელს შორის. ნდობის იერარქია უმთავრესად წარმოადგენს ნდობის დონეების ხილულ წარმოდგენას, რაც ორგანიზაციულ პრივილეგიებს ასახავს. იგი პრივილეგიის თითოეულ დონეს შესაბამის რიცხვს უკავშირებს, რათა

ასახოს უსაფრთხოება, მნიშვნელობა და კვანძებისა და მარშრუტების შესაძლებლობები. ნდობის იერარქიაზე თავდასხმა შესაძლოა კლასიფიცირებულ იქნას, როგორც გარე ან შიდა თავდასხმა, დაფუძნებული ნდობის წონაზე, რომელიც ასოცირებულია იდენტურობასთან ან თავდასხმის წყაროსთან. ასევე საჭიროა მომხმარებელთა იდენტურობებსა და ასოცირებულ ნდობის დონეებს შორის კავშირი. ასეთი კავშირის გარეშე ნებისმიერი მომხმარებელი შეძლებს მიბაძოს სხვას და მოიპოვოს უფრო მაღალ ნდობის დონესთან ასოცირებული პრივილეგიები. აღნიშნულისგან თავის დასაცავად აუცილებელია მისაწვდომობის კონტროლის უფრო ძლიერი მექანიზმები (აუთენტიფიკაცია, ავტორიზება, კალკულირება). იმისათვის, რომ ვაიძულოთ კვანძები და მომხმარებლები, პატივი სცენ ნდობის იერარქიას, შესაძლებელია გამოყენებულ იქნას კრიპტოგრაფიული ტექნიკა, მაგალითად, დაშიფრვა, გასაღებთა საჯარო სერტიფიკატები და ა.შ. ტრადიციულად, გარე თავდასხმებთან გასამკლავებლად აუთენტიფიკაციის რთული სქემები გამოიყენება, [7].

1.3.6 თავდასხმა გადაცემად ინფორმაციაზე

თავდასხმის დასაწყებად, გარდა ხარვეზების გამოყენებისა, რაც დაკავშირებულია Trust დონეების დაცვასთან, მტრულად განწყობილი კვანძები შესაძლოა იყენებდნენ ინფორმაციას, რომელიც მარშრუტიზაციის პროტოკოლთა პაკეტებში ინახება. მსგავსმა შეტევებმა შესაძლოა გამოიწვიოს ინფორმაციის დაზიანება, ინფორმაციის გაცემა, ლეგიტიმური სერვისის მოპარვა პროტოკოლის სხვა ობიექტებისგან ან ქსელის სერვისის გათიშვა პროტოკოლის ობიექტებისთვის. გადაცემადი ინფორმაციის საფრთხეები მოიცავს:

ხელის შეშლას - გადაცემად პროტოკოლთა პაკეტები, განსაკუთრებით მარშრუტის განმსაზღვრელი შეტყობინებები და განახლებები, შესაძლოა შეფერხებულ იქნან ან დაიბლოკონ ან მარშრუტიზაციის პროტოკოლებმა აიძულონ, რომ მცდარად იმოქმედონ,

გადამისამართებას - მარშრუტიზაციის ტრაფიკის პროტოკოლებს და საკონტროლო გზავნილებს, მაგალითად, “Keep alive” და “Are you up?” შესაძლოა მიმართულება შეუცვალონ,

მოდიფიცირებას - მარშრუტიზაციის პროტოკოლთა პაკეტებში არსებული ინფორმაციის მთლიანობა შესაძლოა რისკის ქვეშ დადგეს თავად ამ პაკეტების

თვითმოდიფიცირების გამო. შესაძლებელია მცდარი მარშრუტების გავრცელება და ლეგიტიმური კვანძების გვერდის ავლა,

გაყალბებას - პროტოკოლთა ლეგიტიმურ პაკეტებში მტრულად განწყობილი შიდა კვანძების მიერ შეიძლება შეტანილი იქნას მცდარი მარშრუტები და სიდიდეები.

1.3.7 თავდასხმა მარშრუტიზაციის ან ქსელის დონეზე

მარშრუტიზაციის წინააღმდეგ თავდასხმა ორი ძირითადი სახის არის: შიდა და გარე. გარე თავდასხმა, ადრინდელის მსგავსად, კლასიფიცირებულ უნდა იყოს, როგორც აქტიური ან პასიური, [1].

1.3.7.1 შიდა თავდასხმა

შიდა თავდასხმა უფრო დიდი საფრთხეა უსადენო ქსელისთვის. თავდამსხმელმა შესაძლოა მარშრუტიზაციის მცდარი ინფორმაცია გადასცეს ქსელის სხვა კვანძებს. მტრულად განწყობილი კვანძი შესაძლოა განხილულ იქნას, როგორც შიდა თავდასხმის წყარო. მარშრუტიზაციის პროტოკოლებში მოდიფიცირებული ინფორმაციის აღმოჩენა ძირშივე რთულია, რადგან მტრულად განწყობილ კვანძებს შესაძლებლობა აქვთ საკუთარი კერძო გასაღებების მეშვეობით სწორი ხელმოწერების გენერირება მოახდინონ. ასევე შესაძლოა რთული გახდეს მონაცემების მოდიფიკაციებს შორის დიფერენცირება, რისი მიზეზი შეიძლება იყოს მიმდინარე თავდასხმა ან უსადენო ლინკის დაზიანება.

1.3.7.2 გარე თავდასხმა

მარშრუტიზაციაზე გარე თავდასხმა შეიძლება ორ კატეგორიად დაიყოს: პასიურად და აქტიურად. პასიური თავდასხმა გულისხმობს მარშრუტიზაციის პაკეტების არაუფლებამოსილ "მოსმენას".

აღნიშნული შესაძლოა წარმოადგენდეს მცდელობას მარშრუტიზაციის ინფორმაციის მოპოვებისა, საიდანაც თავდამსხმელს საშუალება ეძლევა მოახდინოს თითოეული კვანძის სხვა კვანძების მიმართ პოზიციის შესახებ მონაცემის ამოღება. ქვემოთ ჩამოთვლილია უსადენო ქსელის წინააღმდეგ ზოგი იოლად განხორციელებადი თავდასხმა:

- მარშრუტიზაციის ცხრილის გადატვირთვა;

- ადგილსამყოფელის გაცემა;
- ჭიის ხვრელი.

1.4 კრიპტოგრაფია

კრიპტოგრაფია არის მათემატიკური ტექნიკის ნაწილი, დაკავშირებული თავდამსხმელებისგან დაცული ინფორმაციის/მონაცემების შენახვასთან, [19]. მაგალითად, კრიპტოგრაფიული მექანიზმები შემუშავებულ იქნა მონაცემთა კონფიდენციალობის დასაცავად. კრიპტოგრაფიული სქემები იმგვარად იქნა შემუშავებული, რომ ეთერით (მაგალითად, უსადენო სისტემის მეშვეობით) გადაცემული ინფორმაცია კოდირებულია და თავდამსხმელების მიერ მისი ინტერპრეტირება შეუძლებელია. და ეს მიუხედავად იმისა, რომ თავდამსხმელმა შესაძლოა მოიპოვოს დაშიფრული მონაცემი ეთერით გადაცემული მონაცემის ხელთ ჩაგდების გზით. კრიპტოგრაფია ასევე შესაძლოა გამოყენებულ იქნას იმაში დასარწმუნებლად, რომ მონაცემი მართლაც იმ სუბიექტის მიერ იყო შექმნილი, რომელმაც მისი შექმნის შესახებ განაცხადა. ამ მახასიათებელს აგრეთვე მონაცემთა აუთენტიფიკაციას უწოდებენ. კრიპტოგრაფია ასევე გამოიყენება უსაფრთხოების სხვა სერვისების უზრუნველსაყოფად, როგორიც არის მონაცემთა მთლიანობა და ავტორობაზე უარის თქმის შეუძლებლობა, რის შესახებ ქვემოთ იქნება საუბარი. კრიპტოგრაფიის სპეციალისტი ფოკუსირებას ახდენს კრიპტოგრაფიული ალგორითმებისა და პროტოკოლების დიზაინსა და ანალიზზე. ანალიზმა შესაძლოა ხელი შეუწყოს არსებული კრიპტოგრაფიული პროტოკოლების გატეხვის გზების აღმოჩენას.

მიუხედავად იმისა, რომ კრიპტოგრაფია წარმოადგენს ერთ-ერთ უმთავრეს საშუალებას, რომლითაც უსაფრთხოების ინჟინრები საინფორმაციო სისტემების დაცვას ახერხებენ, იგი არ გახლავთ ერთადერთი იარაღი, რომელიც მსგავსი ამოცანის გადასაწყვეტად გამოიყენება. სხვა გამოყენებადი ღონისძიებებიდან უნდა აღვნიშნოთ ბიომეტრია და სტენოგრაფია. გარდა ამისა, სისტემების დაცვისთვის შესაძლოა აუცილებელი იყოს იურიდიული ღონისძიებების გატარება, როგორიც არის პასუხისმგებლობის რეგულირება და ასევე დაზღვევა. სისტემის უსაფრთხოების უზრუველსაყოფად ასევე მნიშვნელოვან როლს თამაშობს ორგანიზაციული ღონისძიებები, როგორიც არის უსაფრთხოების სათანადო პოლიტიკა და ინფორმაციის მართებული კლასიფიცირება. ასევე არ შეიძლება

იგნორირებულ იქნას ადამიანურ რესურსთან დაკავშირებული ღონისძიებები, როგორიც არის კონტროლი, მოტივირება და განათლება.

1.4.1 კრიპტოგრაფიის ძირითადი კონცეფციები

კრიპტოგრაფიული მექანიზმები შემუშავებულია კონკრეტული მიზნების მისაღწევად. ამ მიზნებს, როგორც წესი, კრიპტოგრაფიულ მექანიზმებთან ასოცირებულ ატრიბუტებს უწოდებენ. მათ შორისაა , [11]:

- კონფიდენციალობა;
- მთლიანობა;
- აუთენტიფიკაცია;
- ავტორობაზე უარის თქმის შეუძლებლობა;
- ხელმისაწვდომობა.

ამ ფუნდამენტალური ატრიბუტების გარდა არსებობს კიდევ რამდენიმე სხვა ატრიბუტი. მათ შორის უნდა გამოვყოთ:

ანონიმურობა, რაც გულისხმობს რაიმე პროცესში ჩართული ობიექტის იდენტურობის გაუცხადებლობას,

ავტორიზება, რაც წარმოადგენს სხვა ობიექტისთვის რაიმეს ნების დართვის ოფიციალური სანქციის გადაცემის შესაძლებლობას,

დროის დანიშვნა, რაც დროის აღრიცხვის შესაძლებლობას გულისხმობს,

მისაწვდომობის კონტროლი ანუ პრივილეგირებული ობიექტებისთვის რესურსების მისაწვდომობის შეზღუდვა,

ანულირება, რაც ავტორიზების გაუქმებას ნიშნავს და ა.შ.

ზემოთ აღნიშნული მიზნების მისაღწევად ფუნქციის რამდენიმე კრიპტოგრაფიული პრიმიტივი იქნა შემუშავებული. ეს პრიმიტივები შესაძლოა სამ ოჯახად დაიყოს. ესენია:

1. კრიპტოგრაფია სიმეტრიული გასაღებით – გამოიყენება ერთი გასაღები;
2. კრიპტოგრაფია ასიმეტრიული გასაღებით – გამოიყენება ორი გასაღები;
3. შეტყობინების პროფილი – არ გულისხმობს გასაღებთა გამოყენებას.

გასაღების გამოყენებაზე დაფუძნებული კრიპტოგრაფიული სისტემები შეიძლება განვიხილოთ, როგორც კომბინირებული ბოქლომის ანალოგი. მათში გამოიყენება, როგორც ალგორითმი, ასევე საიდუმლო მნიშვნელობა. საიდუმლო მნიშვნელობა არის გასაღები (ანალოგი რიცხვისა, რომელიც კომბინირებულ

ბოქლომს აღებს) და იგი დაცული უნდა იყოს თავდამსხმელებისგან. კრიპტოგრაფიული ალგორითმი შეიძლება განხილულ იქნას, როგორც კომბინირებული ბოქლომის მუშაობის პრინციპის ანალოგი. ითვლება, რომ ალგორითმის დიზაინი ფართოდ არის ცნობილი. ალგორითმის დიზაინის საჯაროობა საზოგადოების მიერ მისი სისუსტეების ანალიზის შესახებ მსჯელობას იწვევს. ამდენად, სისტემის უსაფრთხოება დამოკიდებულია მხოლოდ გასაღების საიდუმლოობაზე და თავდამსხმელმა, რომელსაც სურვილი აქვს გატეხოს კრიპტოგრაფიული სისტემა, კომუნიკაციის უსაფრთხოებისთვის გამოყენებული გასაღები უნდა განსაზღვროს.

მიუხედავად იმისა, რომ გასაღების ზომას დიდი მნიშვნელობა ენიჭება, სხვადასხვა კრიპტოგრაფიული პრიმიტივების შეფასებისთვის გამოიყენება რიგი ფაქტორებისა, რომელთა შორისაა:

- უსაფრთხოების დონე;
- ფუნქციონალურობა;
- მუშაობის მეთოდი;
- განხორციელების სიადვილე;
- წარმადობა.

უსაფრთხოების დონე შესაძლოა ხარჯებთან იყოს დაკავშირებული. უსაფრთხოების უფრო მაღალი დონე შედეგად მოგვცემს იმას, რომ სისტემა უსაფრთხო იქნება კარგად დაფინანსებული თავდამსხმელების წინააღმდეგაც. თუმცა უნდა გვახსოვდეს, რომ უსაფრთხოების დონე, თავისთავად, ხარისხობრივი განზომილებაა.

ფუნქციონალურობა დაკავშირებულია ატრიბუტებთან და მიიღწევა შემოთავაზებული პრიმიტივების გამოყენებით. მაგალითად, ავტორობაზე უარის თქმის შეუძლებლებლობას სჭირდება გასაღების ასიმეტრიული კრიპტოგრაფიის კონცეფციის გამოყენება.

ტიპიურად, დაშიფრული სისტემის მიმართ თავდასხმის მიზანს გასაღების აღდგენა და არა უბრალოდ დაშიფრული ტექსტის აღქმად ტექსტად აღდგენა წარმოადგენს. სტანდარტული დაშიფრული სქემის წინააღმდეგ თავდასხმის ორი ძირითადი მიდგომა არსებობს:

კრიპტოანალიზი. კრიპტოანალიტიკური თავდასხმა ემყარება ალგორითმის ბუნებისა და შესაძლოა მოითხოვდეს არაკოდირებული ან კოდირებული ტექსტის

ძირითადი მახასიათებლების მცირედ ცოდნას. ამ სახის თავდასხმისას ალგორითმის მახასიათებლების გამოყენება ხდება სპეციფიური არაკოდირებული ტექსტის ან გამოყენებადი გასაღების მოპოვებისათვის;

უხეში ძალის თავდასხმა. თავდამსხმელი ყველა სავარაუდო გასაღებს ცდის კოდირებული ტექსტის ნაწილზე სანამ გარდაქმნის არაკოდირებულ ტექსტად. საშუალოდ, წარმატების მისაღწევად არსებულ გასაღებთა ნახევარზე მეტი უნდა იქნას გამოყენებული.

თუკი ორივე სახის შეტევა წარმატებით დასრულდება, შედეგი კატასტროფული იქნება. მოცემული გასაღებით დაშიფრული ყველა წარსული და მომავალი შეტყობინება საფრთხის ქვეშ იქნება.

პრობლემა ყველაზე რთულია, როდესაც მხოლოდ დაშიფრული ტექსტია ხელმისაწვდომი, [11]. ერთ-ერთ სავარაუდო თავდასხმას ასეთ გარემოებებში წარმოადგენს უხეში ძალის თავდასხმა ყველა სავარაუდო გასაღების გამოყენებით. თუმცა მოწინააღმდეგეს შეუძლია თავად კოდირებული ტექსტის ანალიზს დაეყრდნოს და მის მიმართ სხადასხვა სტატისტიკური ტესტები გამოიყენოს. მსგავსი მიდგომის გამოსაყენებლად მოწინააღმდეგეს გარკვეული ზოგადი წარმოდგენა უნდა გააჩნდეს დაფარული კოდირებული ტექსტის შინაარსის შესახებ. მაგალითად, ინგლისურია იგი თუ ფრანგული, EXE (შესრულებადი) ფაილებია თუ Java (პროგრამირების ენა) წყაროების ჩამონათვალი, საბუღალტრო ფაილი და ა.შ.

ცხრილი 1.3 გვიჩვენებს დროის რა რაოდენობაა საჭირო გასაღებთა სხვადასხვა ზომისათვის, [11]. 56-ბიტიანი გასაღებისთვის გამოიყენება DES Data Encryption Standard (მონაცემთა კოდირების სტანდარტი) ალგორითმი, ხოლო 168-ბიტიანი გასაღებისთვის – სამმაგი DES-ი. გასაღების მინიმალური ზომა, განსაზღვრული AES-თვის (Advanced Encryption Key-კოდირების თანამედროვე სტანდარტი), არის 128 ბიტი. შედეგები ასევე ნაჩვენებია იმის მიხედვით, რასაც ჩამანაცვლებელ კოდებს უწოდებენ და მათთვის 26-ნიშნიანი გასაღები გამოიყენება, სადაც 26 ნიშნის ყველა შესაძლო გადანაცვლება გასაღებს წარმოადგენს. გასაღების ყველა შესაძლო ზომისათვის შედეგი ნაჩვენებია იმის დაშვებით, რომ გაშიფრვის თითოეულ ოპერაციას 1 მიკროწამი სჭირდება, რაც საკმარის სიდიდეს წარმოადგენს დღევანდელი მანქანებისთვის. პარალელური სტრუქტურების მქონე მიკროპროცესორების გამოყენებით, გაცილებით უფრო მაღალი მნიშვნელობის მიღწევა შეიძლება. ცხრილის უკანასკნელ სვეტში მოცემულია შედეგები

სისტემისთვის, რომელსაც 1 მილიონი გასაღების დამუშავება შეუძლია მიკროწამის განმავლობისას. როგორც ვხედავთ, DES აღარ შეიძლება გამოთვლის კუთხით უსაფრთხოდ ჩაითვალოს.

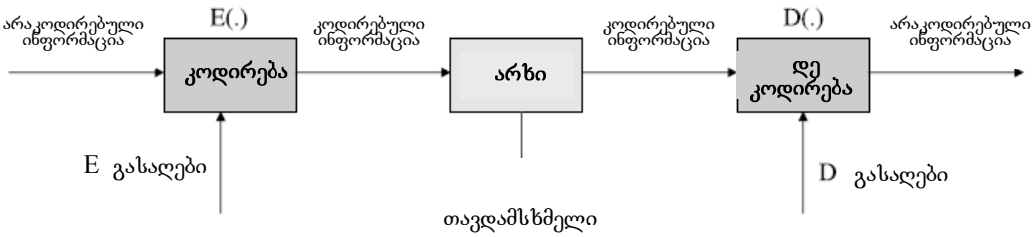
ცხრილი 1.3 საშუალო დრო, აუცილებელი გასაღების სრული ამოხსნისთვის.

გასაღების ზომა (ბიტი)	ალტერნატულ გასაღებთა რაოდენობა	დრო, საჭირო 1 გაშიფრვისთვის/μs	დრო, საჭირო 10^6 გაშიფრვისთვის μs-ში
32	2^{32} = 4.3 x 10^9	2^{31} μs = 35.8 წუთი	2.15 მიკროწამი
56	2^{56} = 7.2 x 10^{16}	2^{55} μs = 1142 წელი	10.01 საათი
128	2^{128} = 3.4 x 10^{38}	2^{127} μs = 5.4 x 10^{24} წელი	5.4 x 10^{18} წელი
168	2^{168} = 3.7 x 10^{50}	2^{167} μs = 5.9 x 10^{36} წელი	5.9 x 10^{30} წელი
26 ნიშანი (გადანაცვლება)	26! = 4 x 10^{26}	2 x 10^{26} μs = 6.4 x 10^{12} წელი	6.4 x 10^6 წელი

1.4.2 სიმეტრიული კრიპტოგრაფია

სიმეტრიულ-გასაღებიანი კრიპტოგრაფიის მუშაობა ნაჩვენებია სურ.1.2-ზე. აქ ხდება გასაგზავნი არაკოდირებული ტექსტის კოდირება კოდირების E გასაღებით. შედეგად მიღებული კოდირებული ტექსტი შესაძლოა გადაცემულ იქნას არხის (უსადენო, საკაბელო ან კომბინირებული) მეშვეობით. იგულისხმება, რომ მტრულად განწყობილი პირებისთვის მისაწვდომია არხით გადაცემადი კოდირებული ტექსტი, ისე, როგორც ეს ნახატზეა ნაჩვენები. პასიურ თავდამსხმელებს კოდირებული ტექსტის მხოლოდ ჩაწერა შეუძლიათ, როდესაც აქტიური თავდამსხმელები მის მოდიფიცირებას შეეცდებიან. შემდგომ ამისა, კოდირებული ტექსტი აღწევს მიმღებს, სადაც ხდება მისი გაშიფრვის D გასაღებით გაშიფვრა, როგორც ნახატზეა ნაჩვენები. გაშიფრვის შედეგად ხდება ორიგინალური

არაკოდირებული ტექსტის აღდგენა, იმის დაშვებით, რომ კოდირებული ტექსტი გადაცემისას არ იქნა მოდიფიცირებული აქტიური თავდამსხმელის მიერ. ჩვეულებრივ, ორივე გასაღები – E და D – ერთი და იგივეა და ამ საერთო გასაღებს ერთობლივი გასაღები ეწოდება. ასეთ სიმეტრიულ გასაღებთა სქემები შესაძლოა გამოყენებულ იქნას კონფიდენციალობის, მთლიანობისა და აუთენტიფიკაციის მისაღწევად. სიმეტრიულ-გასაღებიანი სქემების ძირითადი მოთხოვნა ის გახლავთ, რომ კომუნიკაციაში ჩართული მხარეები ერთობლივ გასაღებს უნდა იყენებდნენ. აღნიშნული კი გულისხმობს იმას, რომ ერთობლივი გასაღები უსაფრთხო საკომუნიკაციო არხით იქნას გაგზავნილი, როგორც ეს ნაჩვენებია სურ.1.3-ზე. აღნიშნული უკავშირდება გასაღების გადაცემის პრობლემას. ეს უმთავრესი პრობლემაა, განსაკუთრებით უსადენო ქსელებში. მიუხედავად ამისა, შეიძლება წარმოიშვას შეკითხვა: "რატომ არ შეიძლება მონაცემთა გადასაცემად იმავე უსაფრთხო არხის გამოყენება, რომლითაც საიდუმლო გასაღების გადაცემა ხდება?" სავარაუდოდ, აღნიშნული შეუძლებელია ორიდან ერთ-ერთი მიზეზის გამო: ასეთ არხებზე სიხშირის შეზღუდვის ან იმის გამო, რომ, როდესაც მონაცემებია გადასაცემი, არხი ხელმისაწვდომი არ არის, [13].



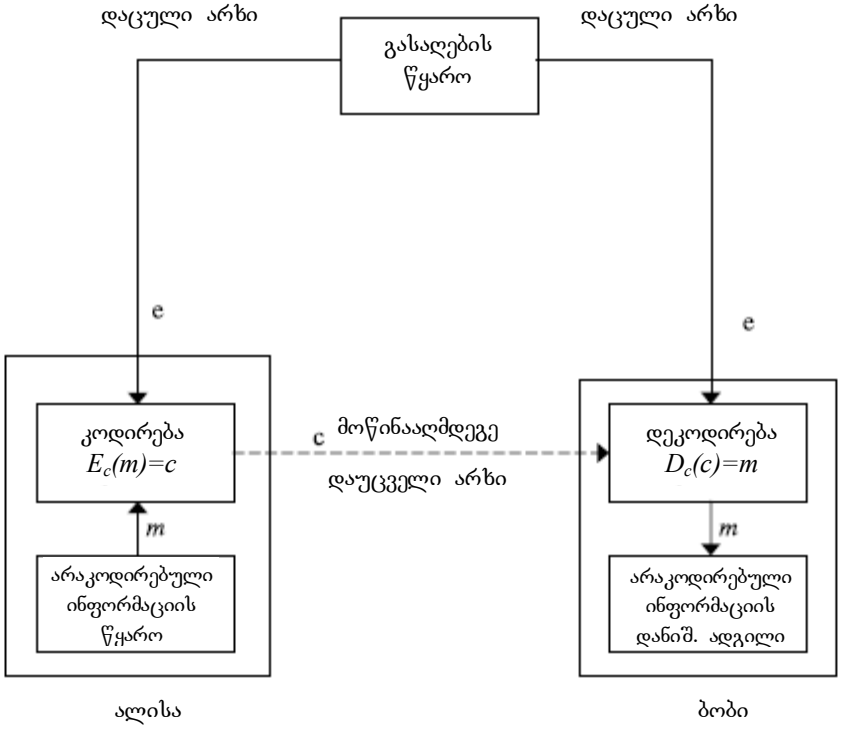
სურ.1.2. სიმეტრიულ-გასაღებიანი კრიპტოგრაფიის ძირითადი ფუნქციონირება.

სიმეტრიულ გასაღებთა ალგორითმები ორი სახისაა, კერძოდ, ბლოკ-კოდები და ნაკადოვანი კოდები, [20].

ბლოკ-კოდები მონაცემთა ბლოკებზე მუშაობს ერთდროულად. ყველა ბიტი, რომელიც ბლოკს შეადგენს, ხელმისაწვდომი უნდა იყოს მანამ, სანამ ბლოკის დამუშავება დაიწყება. შესაბამისად, ბლოკ-კოდი შესაძლოა განხილულ იქნას, როგორც ფუნქცია, რომელიც n-ბიტ არაკოდირებულ ტექსტს გარდაქმნის n-ბიტ კოდირებულ ტექსტად. ფუნქცია უნდა იყოს ერთგვაროვნად შესაბამისი, რათა უზრუნველყოფილი იყოს ერთგვაროვანი გაშიფრვა. ბლოკის სიგრძის ზუსტი ზომა

მნიშვნელოვანია უსაფრთხოების, შესრულებისა და სირთულის თვალსაზრისით. ბლოკის მცირე სიგრძე, სავარაუდოდ, ხელს შეუწყობს თავდამსხმელს მარტივად შეიმუშაოს გაშიფრვის ცხრილი, რომელიც შეიცავს არაკოდირებულ-კოდირებულ ტექსტთა წყვილებს. ბლოკის დიდი სიგრძე მოუხერხებელია მონაცემთა დაშიფრვისა და გაშიფრვის გამოთვლების შესრულების სირთულის გამო და ასევე მსგავსი გამოთვლების შესრულებასთან დაკავშირებული ვარიმების მიზეზით. როგორც წესი, გამოყენებული ბლოკის სიგრძეა 64 ან 128 ბიტი. ეს არის კრიპტოგრაფიული ალგორითმის ყველაზე ფართოდ გავრცელებული ფორმა.

ნაკადოვანი კოდები ერთდროულად მუშაობს შეტყობინების ბიტზე ან ბაიტზე. შესაბამისად, ხდება მონაცემის, როგორც "ნაკადის" დამუშავება. თუკი ბლოკი მცირე ზომისაა (ბიტი ან ბაიტი), ნაკადოვანი კოდები შესაძლოა არაეფექტური იყოს. ისინი უფრო სწრაფია, ვიდრე ბლოკ-კოდები მოწყობილობაში და მოითხოვს ნაკლებად რთულ სქემებს. ნაკადოვანი კოდების უმეტესობა კერძო და კონფიდენციალურია.



სურ.1.3 სიმეტრიულ-გასაღებიანი კოდირების სისტემის მოდელი.

1.4.3 ასიმეტრიული კრიპტოგრაფია

ასიმეტრიული კოდირება კრიპტოგრაფიის სამათასწლიანი ისტორიის მანძილზე ძალიან მნიშვნელოვან წინგადადგმულ ნაბიჯს წარმოადგენს. ადრე ჩვენ ვნახეთ, რომ ტრადიციული სიმეტრიულ-გასაღებიანი კრიპტოგრაფია ერთ გასაღებს იყენებს, რომლითაც გამგზავნი და მიმღები ერთობლივად სარგებლობენ. გარდა ამისა, აღნიშნული მიდგომა ვერ უზრუნველყოფს დაცვას, თუ მიმღები გააყალბებს შეტყობინებას და განაცხადებს, რომ იგი გამგზავნისგან არის. აღნიშნული შესაძლებელია, რადგან ორი მხარე ფლობს გასაღებს და ამ ერთობლივი გასაღების გამოყენებით თითოეულ მათგანს შეუძლია შეტყობინების შექმნა. ამდენად, სიმეტრიულ-გასაღებიანი სქემები საკუთარ თავს არ ანიჭებენ გამგზავნის იოლად აუთენტიფიკაციის შესაძლებლობას, [7].

მოცემულ სქემებთან დაკავშირებული კიდევ ერთი პრობლემა უკავშირდება გასაღებთა უსაფრთხო განაწილებას. ამ პრობლემებთან ბრძოლა ასიმეტრიულ-გასაღებიანი კრიპტოგრაფიის მიდგომის გამოყენებით შეიძლება. ასიმეტრიული მიდგომა ორ გასაღებს იყენებს – საჯარო და კერძო გასაღებებს. საჯარო გასაღები შესაძლოა ცნობილი იყოს ნებისმიერისთვის, როდესაც იგულისხმება, რომ კერძო გასაღები მხოლოდ შეტყობინების შემქმნელი ობიექტისთვის არის ცნობილი. ქსელის ყოველ ობიექტს, რომელსაც შეტყობინების გაგზავნის სურვილი აქვს, ეს ორი გასაღები გააჩნია, სახელდობრ კი საჯარო და კერძო გასაღები. გასაღებები ერთმანეთისგან განსხვავდება, მაგრამ ერთმანეთთან მათემატიკურად არის დაკავშირებული. გარდა ამისა, გამოთვლის კუთხით არაპრაქტიკულია კერძო გასაღების ამოხსნა მხოლოდ საჯარო გასაღებისა და კრიპტოგრაფიული ალგორითმის ცოდნით. კერძო გასაღების ამოხსნას დამატებითი ინფორმაცია დასჭირდება. თუმცა, მიუხედავად ზემოთქმულისა, გამოთვლის თვალსაზრისით ადვილი უნდა იყოს შეტყობინების დაშიფრვა ან გაშიფრვა, როდესაც შესაბამისი გასაღები ცნობილია.

აღნიშნულ მიდგომას ასევე ასიმეტრიული ეწოდება, რადგან შეტყობინებათა დაშიფრვისას იმისგან განსხვავებული გასაღები გამოიყენება, რაც მისი გაშიფრვისას. შესაბამისად, სხვა ობიექტებს შეუძლიათ კვანძისკენ მიმართული შეტყობინების კოდირება ამ კვანძის საჯარო გასაღების გამოყენებით. ამის შემდეგ მხოლოდ კვანძს შეეძლება შეტყობინების გაშიფრვა მისი კერძო გასაღების

გამოყენებით. ასეთ შემთხვევებში მათ, ვისაც შეტყობინების კოდირება შეუძლიათ, არ შეუძლიათ მისი გაშიფრვა.

სიმეტრიული კოდირების სქემებს უსაფრთხო არხები სჭირდებათ კოდირებისთვის გამოსაყენებელი გასაღების გადასაცემად. მეორეს მხრივ, საჯარო გასაღების კოდირებას არ სჭირდება ასეთი უსაფრთხო არხი და არხის მიმართ მოთხოვნაც უფრო რბილია. მას მხოლოდ ისეთი არხი სჭირდება, რომელიც აკეთებს აუთენტიფიკაციას. აღნიშნული საჭიროა მეორე მხარის საჯარო გასაღების ჭეშმარიტობის დასადასტურებლად. ასეთი არხით გადაცემული ინფორმაციის კონფიდენციალობის დაცვა აუცილებელი არ არის.

ასიმეტრიულ-გასაღებიან სქემებს შეუძლიათ უზრუნველყონ ავტორობაზე უარის თქმის შეუძლებლობა, კონფიდენციალობა, მთლიანობა და აუთენტიფიკაცია. თუმცა ასიმეტრიული კოდირება შესამჩნევად ნელია.

ამ ფაქტის გამოისობით, ზოგადად საჯარო გასაღების კოდირების სქემები მხოლოდ მონაცემთა მცირე რაოდენობის კოდირებისთვის გამოიყენება, მაგალითად, გასაღებები, გამოყენებული სიმეტრიული კოდირებისთვის.

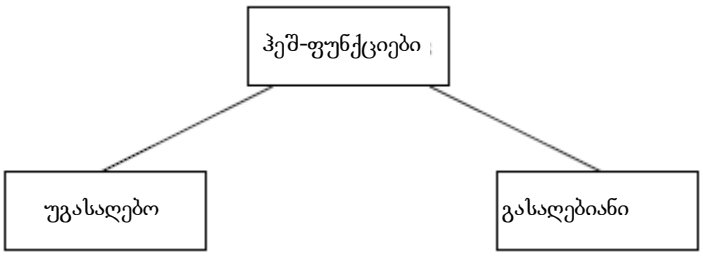
როგორც ზემოთ აღვნიშნეთ, კრიპტოგრაფია ასიმეტრიული გასაღებით, აუთენტიფიკაციის გარდა, ავტორობაზე უარის თქმის შეუძლებლობის უნარსაც უზრუნველყოფს. აღნიშნული მიიღწევა ისეთი კონცეფციის გამოყენებით, რასაც ციფრული ხელმოწერა ეწოდება და განზრახულია იმისათვის, რომ ხელით ხელმოწერის ციფრული ასლი უზრუნველყოს. ასეთი შემთხვევისას ობიექტს შეტყობინების ტრანსფორმირება კერძო გასაღების მეშვეობით შეუძლია. აღნიშნული გამოიყენება ობიექტის ხელმოწერად შეტყობინებაზე. ამის შემდეგ ნებისმიერ სხვა ობიექტს შეუძლია ხელმომწერის საჯარო გასაღების მეშვეობით გადაამოწმოს ტრანსფორმაცია ხელმოწერილ შეტყობინებაზე. აღნიშნული ციფრული ხელმოწერის კონცეფციას წარმოადგენს. კონცეფცია ეწოდება, რადგან იგულისხმება, რომ მხოლოდ ობიექტისთვის არის ხელმისაწვდომი მისი კერძო გასაღები. ამდენად, ასეთი შემთხვევისას კერძო გასაღები ხელმოწერის შესაქმნელად გამოიყენება, ხოლო საჯარო გასაღები – ხელმოწერის შესამოწმებლად. უნდა გვახსოვდეს, რომ ციფრულ ხელმოწერას გარკვეული მახასიათებლები უნდა გააჩნდეს. მაგალითად, იგი დამოკიდებული უნდა იყოს ხელმოწერილ შეტყობინებაზე, უნდა იყენებდეს გამგზავნისთვის უნიკალურ ინფორმაციას, რათა დაცული იყოს, როგორც გაყალბების, ასევე უარყოფისგან,

მისი შექმნა, ცოდნა და გადამოწმება შედარებით იოლი უნდა იყოს, მისი გაყალბება გამოთვლის თვალსაზრისით არაპრაქტიკული უნდა იყოს (როგორც ახალი შეტყობინებით არსებული ციფრული ხელმოწერისთვის, ისე გაყალბებული ციფრული ხელმოწერით მოცემული შეტყობინებისთვის) და მისი შენახვა მოსახერხებელი უნდა იყოს. შემოთავაზებულია რიგი ალგორითმებისა ასიმეტრიული გასაღებებით, მაგრამ მხოლოდ რამდენიმეა უსაფრთხოც და პრაქტიკულიც. გარდა ამისა, ზოგი ალგორითმი კოდირებისთვის უფროა მოსახერხებელი, როდესაც სხვები უფრო ეფექტურია ციფრული ხელმოწერის გათვალისწინებით. არსებობს სამი ალგორითმი, რომელიც კარგად მუშაობს, როგორც კოდირების, ასევე ციფრული ხელმოწერებისთვისაც. ესენია RSA ალგორითმი, ElGamal ალგორითმი და Rabin ალგორითმი.

1.4.4 შეტყობინების პროფილი

ალგორითმთა კიდევ ერთი კლასი, რომელიც ფართოდ გამოიყენება კრიპტოგრაფიულ პროტოკოლებში, არის ჰეშირებულ ფუნქციათა კლასი, [22]. ასეთი ალგორითმები ორი სხვადასხვა სახისაა, როგორც ეს ნაჩვენებია სურათზე 1.4. უგასაღებო ჰეშ-ფუნქციებს არ სჭირდებათ საიდუმლო გასაღები, როდესაც გასაღებიან ჰეშ-ფუნქციებს საიდუმლო გასაღები სჭირდებათ.

უგასაღებო ჰეშ-ფუნქციებს უბრალოდ ჰეშ-ფუნქციებს უწოდებენ, ხოლო გასაღებიან ჰეშ-ფუნქციებს – შეტყობინების აუთენტიფიკაციის კოდებს (MAC) Message Authentication Code, [11,19].



სურ.1.4. ჰეშ-ფუნქციის კლასიფიკაცია.

1.5 გასაღების მართვა

ქსელის კვანძებისთვის აუცილებელია, რომ ერთმანეთთან უსაფრთხო კომუნიკაცია შეეძლოთ, [1,11]. უსაფრთხო საკომუნიკაციო არხების არსებობა განსაკუთრებით მნიშვნელოვანია უსადენო ქსელებში, რის მიზეზსაც უსადენო ლინკებისა და ასეთი ქსელების სხვა მახასიათებლების გამოყენება წარმოადგენს, [25-27]. ასეთი არხები საჭიროა მრავალი ოპერაციისთვის, როგორიც არის მონაცემების გაცვლა ან საკონტროლო პაკეტების გაცვლა მარშრუტიზირების დროს. იმისათვის, რომ ასეთი უსაფრთხო კომუნიკაცია შესაძლებელი იყოს, აუცილებელია, რომ კვანძებისთვის ხელმისაწვდომი იყოს გასაღებთა სათანადო მასალა. აღნიშნული წარმოადგენს გასაღების მართვის პროცესის მიზანს. უკანასკნელ დროს იგი იქცა უსადენო ქსელების კვლევის ძალიან აქტიურ არეალად, [28].

გასაღების მართვის მნიშვნელობის გადაჭარბებით შეფასება, როგორც კაბელიანი, ისე უსადენო ქსელებისთვის, შეუძლებელია. კრიპტოგრაფიული სქემების გამოყენებისას, როგორიც არის კოდირება და ციფრული ხელმოწერები, კონტროლისა და მონაცემთა ტრაფიკის დასაცავად, გასაღების მართვის სერვისი ყოველთვის აუცილებელია. ნებისმიერ ორ მხარეს შორის უსაფრთხო კომუნიკაციისთვის ორივე ობიექტს უნდა გააჩნდეს საიდუმლო სიდიდე ან გასაღები. სავარაუდო გზები, რომელთა გამოყენებით არის შესაძლებელი ასეთი უსაფრთხო კომუნიკაციის დამყარება, განსახილველი ობიექტებისთვის ნიშნავს ან ერთობლივი გასაღების ფლობას (სიმეტრიულ-გასაღებიანი სისტემა) ან განსხვავებული გასაღების ფლობას (ასიმეტრიულ-გასაღებიანი სისტემა). გასაღების მართვი არის პროცესი, რომლის მეშვეობითაც ხდება გასაღებთა მიწოდება ქსელის კვანძებამდე, მათი გაუმჯობესება (აუცილებლობისას), წაშლა და ა.შ. არსებობს რამდენიმე საფეხური, რომლითაც უნდა იყოს დაკავებული გასაღების მართვი, როგორც სიმეტრიულ-გასაღებიანი სისტემის, ასევე ასიმეტრიულ-გასაღებიანი სისტემისთვის. მათ შორისაა:

1. სისტემის მომხამრებელთა ინიციალიზაცია;
2. გასაღების მასალის შექმნა, განაწილება და ინსტალაცია;
3. გასაღების მასალის გამოყენების ორგანიზება;
4. გასაღების მასალის მოდერნიზება, გაუქმება და განადგურება;

5. გასაღების მასალის დაარქივება.

პირველი საფეხურის დანიშნულებაა სისტემის გაშვება. იგი შესაძლოა მოიცავდეს სხვადასხვა არაკრიპტოგრაფიულ ოპერაციებს, როგორიც არის მომხარებელთა ინფორმაციის შემოწმება, სისტემის მომხმარებელთა იდენტურობის უზრუნველყოფა, დადასტურება იმისა, რომ მათ სათანადო პროგრამული უზრუნველყოფა გააჩნიათ გასაღების მართვის პროცესში მონაწილეობისთვის და ა.შ. აღნიშნულს შემდეგ მოსდევს გასაღების მასალის შექმნა და განაწილება. გასაღების მასალის შექმნა და განაწილება შესაძლოა მოხდეს ცენტრალიზებული ან არაცენტრალიზებული სახით. გასაღების მასალა ინსტალირდება სხვადასხვა კვანძებზე. ამას მოყვება მესამე საფეხური, რომელზეც გასაღების მასალა გამოიყენება სხვადასხვა კვანძებს შორის კომუნიკაციის დასაცავად. მეოთხე საფეხური გადამწყვეტია იმ საფრთხეებზე რეაგირებისთვის, რომლებმაც შესაძლოა გამოიწვიონ გასაღების რისკის ქვეშ დაყენება. გასაღების რისკის ქვეშ დაყენებამ შესაძლოა მიგვიყვანოს კონფიდენციალობის ნაკლებობამდე, ისევე როგორც გასაღების არაუფლებამოსილ გამოყენებამდე. ასეთი შემთხვევებისას გასაღების მართვის პროცესებმა გასაღების გაუქმება (ანულირება) უნდა უზრუნველყონ. გარდა ამისა, ზოგჯერ შესაძლოა რისკის ქვეშ დაყენებული გასაღების ჩანაცვლება გახდეს აუცილებელი. არ არის აუცილებელი იმ კვანძის რისკის ქვეშ დაყენებული გასაღების ჩანაცვლება, რომელიც თავდამსხმელის კონტროლს ქვეშ იმყოფება. საბოლოოდ, შესაძლოა აუცილებელი გახდეს მეხუთე საფეხური ისეთი შემთხვევებისას, როდესაც გასაღების მასალის შენახვაა საჭირო. აღნიშნული განსაკუთრებით მნიშვნელოვანია ისეთ სიტუაციებში, როდესაც გასაღების ასეთი მასალა აუცილებელია აუდიტის მიზნებისთვის, როგორც ეს, მაგალითად, ხდება სამართალწარმოებისას.

როგორც ადრე უკვე იქნა ახსნილი, არსებობს კრიპტოგრაფიული სისტემის ორი ძირითადი კატეგორია, კერძოდ სიმეტრიული და ასიმეტრიული კრიპტოგრაფიული სისტემები. გასაღების მართვის პროცესი განსხვავებულ ტექნიკას გულისხმობს ამ ორი სახის კრიპტოგრაფიული სისტემებისთვის. სიმეტრიულ გასაღებთა სისტემებს შორის Kerberos გასაღების მართვის ფართოდ გავრცელებული სისტემაა ტრადიციულ ქსელებში. უნდა აღინიშნოს, რომ გასაღების მართვა უსადენო ქსელებში უფრო რთულია, ვიდრე ტრადიციულ

სისტემებში. ამის მიზეზი რამდენიმე ფაქტორია, როგორიც არის უსადენო ლინკების კაპრიზულობა, ცენტრალური უფლებამოსილების ნაკლებობა, სიმწირე რესურსებისა, როგორიც არის ელექტროენერგია, მეხსიერება და სიხშირე, და შეუძლებლობა გაშვების შემდეგ კვანძის მეზობლების წინასწარ განსაზღვრისა. წინამდებარე თავში განხილულია გასაღების მართვის პრობლემა უსადენო ქსელებში და ახსნილია რამდენიმე სავარაუდო მიდგომა აღნიშნული პრობლემის გადასაჭრელად.

1.5.1 ასიმეტრიულ გასაღებზე დაფუძნებული მიდგომა

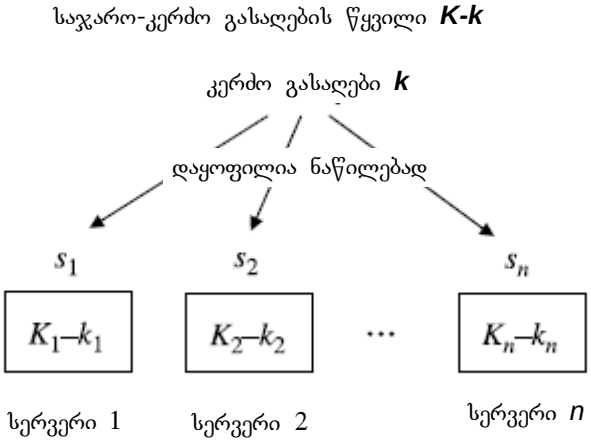
საჯარო გასაღების კრიპტოგრაფია მოითხოვს უსაფრთხო ობიექტის არსებობას, რაც ცნობილია გასაღების მართვის სერტიფიცირების ორგანოს (CA) სახელით. CA-ს გააჩნია საჯარო/კერძო გასაღებთა წყვილი, რომლის საჯარო გასაღები ცნობილია ყველა კვანძისთვის. CA გასცემს სერტიფიკატებს, რომლებიც საჯარო გასაღებს კვანძს აბამენ. ასიმეტრიულ გასაღებზე დაფუძნებული სისტემის ტრადიციული მიდგომა ემყარება CA-ს გამოყენებას. მიუხედავად ამისა, მსგავსი მიდგომა რამდენიმე მიზეზით არის არაპრაქტიკული უსადენო ქსელებისთვის. პირველ რიგში, CA ქსელის დაუცველი პუნქტი იქნება, განსაკუთრებით ისეთი შემთხვევისას, თუ არ ხდება მისი განაწილება. CA-ს რისკის ქვეშ დაყენება საშუალებას მისცემს თავდამსხმელს ხელი მოაწეროს ნებისმიერ სერტიფიკატს და ამგვარად შეძლოს ნებისმიერი კვანძის მიბაძვა ან ნებისმიერი სერტიფიკატის გაუქმება. კიდევ უფრო მნიშვნელოვანია ის, რომ გასაღების მართვის ოპერაციების შესრულებისთვის CA ნებისმიერ დროს უნდა იყოს ხელმისაწვდომი. თუ იგი ხელმისაწვდომი არ არის, სისტემის კვანძებმა შესაძლოა ვერ შეძლონ გასაღებთა განახლება/შეცვლა. ვერც ახალი კვანძები შეძლებენ სერტიფიკატების მოპოვებას. მისაწვდომობის გაუმჯობესებისათვის შეიძლება გაკეთდეს CA-ს სერვისების დუბლირება, მაგრამ მარტივმა დუბლირებამ შესაძლოა კიდევ უფრო მეტ პრობლემებამდე მიგვიყვანოს. ნებისმიერი ცალკეული ასლის რისკის ქვეშ დაყენებამ შესაძლოა მთლიანი სისტემის კოლაფსი გამოიწვიოს. აღნიშნული პრობლემის გადაჭრის გზაა CA-ს ნდობის გადანაწილება კვანძების ჯგუფში კვანძებს შორის და ამგვარად კვანძებისთვის პასუხისმგებლობის განაწილების საშუალების მიცემა, [7].

1.5.1.1 ნაწილობრივ გადანაწილებული უფლებამოსილება

წინამდებარე თავში მოკლედ არის აღწერილი ზღვრული კრიპტოგრაფიის (ThC) threshold cryptography კონცეფციის სქემა, რაც საშუალებას აძლევს n მხარეებს გაინაწილონ კრიპტოგრაფიული ოპერაციის შესრულების უნარი. მაგალითისთვის განვიხილოთ ციფრული ხელმოწერა შეტყობინებაზე. არსებობს ტექნიკა, რომლის მეშვეობით ცალკეულ მომხმარებელს შეუძლია ციფრული ხელმოწერის შექმნა. თუმცა პრობლემა წარმოიქმნება ისეთ დროს, როდესაც მოცემული მომხმარებელი რისკს წარმოადგენს ან მისი ნდობა არ შეიძლება. ასეთი შემთხვევისას უკეთეს მიდგომას წარმოადგენს ცალკეული მომხმარებლის ნდობის განაწილება მრავლობით მომხმარებლებს შორის. სწორედ ამის მისაღწევად იბრძვის ზღვრული კრიპტოგრაფია. ზღვრული კრიპტოგრაფიის მიზანს წარმოადგენს ინფორმაციის დაცვა მისი განაწილებით n ჯგუფის ობიექტებს შორის. გარდა ამისა, არსებობს ზღვარი t, ასოცირებული ThC სქემებთან, სადაც n ჯგუფის ნებისმიერ t-ს შეუძლია კრიპტოგრაფიული ოპერაციის შესრულება. ასეთ სქემებს (n,t) ThC სქემებს უწოდებენ. (n,t) ThC სქემების შემთხვევისას t ჯგუფში შემავალ წევრებზე ნაკლები ჯგუფი ვერ შეძლებს კრიპტოგრაფიული ოპერაციის წარმატებით შესრულებას. მიუხედავად ამისა, ThC შესაძლოა განხილული იყოს, როგორც საიდუმლო ინფორმაციის უსაფრთხო განაწილების მიდგომა. აქვე შეგვიძლია დავინახოთ, რომ იმ შემთხვევაშიც კი, როდესაც ობიექტების გარკვეული რაოდენობა (ზღვრულ t-ზე ნაკლები) ქსელში რისკის ობიექტებს წარმოადგენენ, სისტემა არ დგას რისკის ქვეშ. ქსელის გარკვეული რაოდენობის კვანძების მიუწვდომლობას (უფრო ზუსტად კი n- t კვანძი) ასევე არ ექნება გავლენა სისტემის მუშაობაზე. ThC სქემები კრიპტოგრაფიულ ოპერაციას განაწილების მეთოდით ასრულებენ, [1, 29].

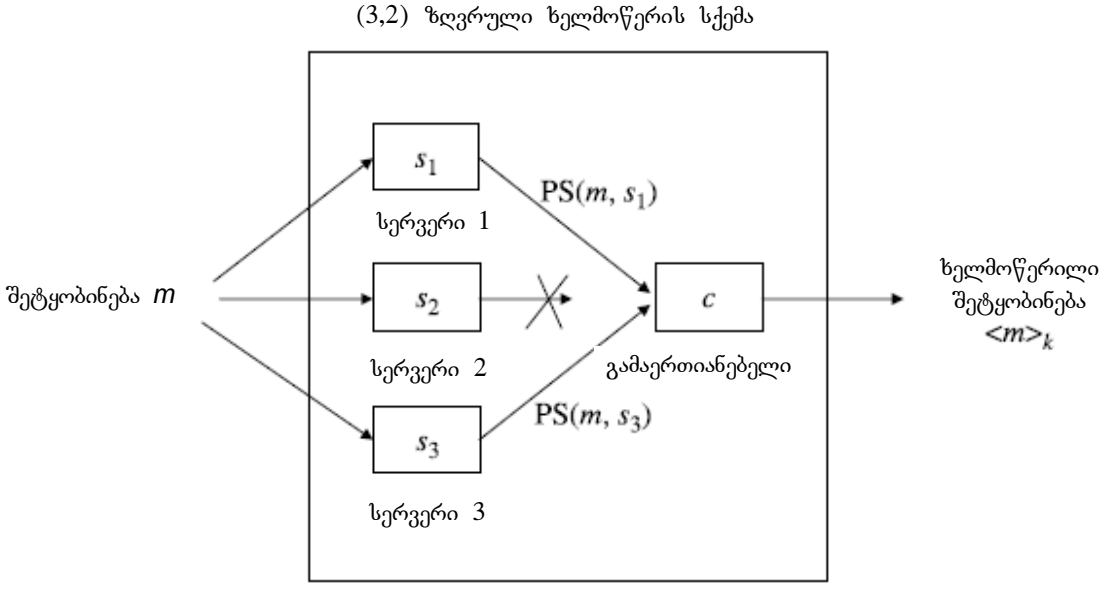
შესაძლოა გამოყენებულ იქნას სქემა, დაფუძნებული ზღვრული კრიპტოგრაფიის ტექნიკაზე. სისტემას, რომელიც ქსელის კვანძებისგან შედგება, სავარაუდოდ საჯარო/კერძო გასაღების წყვილი უნდა გააჩნდეს. გასაღებთა აღნიშნული წყვილი თავდაპირველად, კვანძების გაშვებამდე, შექმნილია სანდო უფლებამოსილი ობიექტის მიერ. ამის შემდეგ კერძო გასაღები იყოფა n ნაწილად (n, t+1) ზღვრული კრიპტოგრაფიის სქემის გამოყენებით. შემდეგ ეს n ნაწილები განთავსდება შემთხვევით არჩეულ კვანძებში უფლებამოსილი ობიექტის მიერ,

რომელმაც საჯარო/კერძო გასაღების წყვილი შექმნა. ამ შერჩეულ კვანძებს სერვერები ეწოდება. კერძო გასაღების ნაწილების სერვერებისთვის განაწილების შემდეგ ცენტრალური უფლებამოსილი ობიექტი აღარ არის საჭირო. შესაბამისად, ცენტრალური უფლებამოსილი ობიექტი საჭიროა მხოლოდ ჩატვირთვის ფაზაზე. თითოეულ სერვერს ასევე გააჩნია გასაღებთა საკუთარი წყვილი და ინახავს ქსელის ყველა კვანძის საჯარო გასაღებებს. კერძოდ, თითოეულმა სერვერმა (შერჩეულმა კვანძმა) იცის სხვა სერვერთა საჯარო გასაღებები. შედეგად, სერვერებს ერთმანეთს შორის უსაფრთხო ლინკების ჩამოყალიბება შეუძლიათ. აღნიშნული სერვისის საწყისი კონფიგურაცია მოცემულია სურათზე 1.5, [30]. სერვისს, როგორც მთლიანობას, გააჩნია საჯარო/კერძო გასაღებთა წყვილი K–k. საჯარო გასაღები K ცნობილია ყველა კვანძისთვის, როდესაც კერძო გასაღები k დაყოფილია ნაწილებად s1, . . . , sn, სადაც ყოველ სერვერს ერთი ნაწილი აქვს. ყოველ სერვერს ასევე გააჩნია საჯარო/კერძო გასაღებთა წყვილი Ki–ki. როდესაც საჭიროა, რომ სერტიფიკატი ხელმოწერილ იქნას სისტემის კერძო გასაღების გამოყენებით, ხდება სერვერებთან დაკავშირება. თითოეული სერვერი ახდენს სერტიფიკატის ნაწილობრივი ხელმოწერის გენერირებას კერძო გასაღების იმ ნაწილის გამოყენებით, რომელიც სერვერს გააჩნია. ნაწილობრივი ხელმოწერა შემდეგ გამაერთიანებელს მიეწოდება, რომელიც ნაწილობრივი ხელმოწერებიდან ახდენს სრული ხელმოწერის გამოთვლას. უნდა აღინიშნოს, რომ გამაერთიანებელი ვერ შეძლებს სრული ხელმოწერის შექმნას ნაწილობრივი ხელმოწერების გარეშე.



სურ.1.5. ნაწილობრივ განაწილებული უფლებამოსილების კონფიგურაცია.

ვინაიდან მოცემული სქემა ემყარება ზღვრული კრიპტოგრაფიის კონცეფციას, სისტემას შეუძლია სარისკო სერვერების განსაზღვრული რაოდენობის დაშვება. შესაბამისად, t ან t-ზე ნაკლები რაოდენობის სარისკო სერვერები ვერ შეძლებენ სისტემის კერძო გასაღების მიღებას. ეს ხდება, რადგან სარიკსო სერვერებს (იგულისხმება, რომ t-ზე მეტი არ არის) თავად არ შეუძლიათ სწორად ხელმოწერილი სერტიფიკატების გენერირება, გამომდინარე იქიდან, რომ მათ შეუძლიათ არაუმეტეს t ნაწილობრივი ხელმოწერის გენერირებისა. გარდა ამისა, არა ყველა n სერვერია საჭირო სრული ხელმოწერის გენერირებისთვის. საკმარისი იქნება ნებისმიერი t+1 სერვერი. სურ.1.6 გვიჩვენებს, როგორ ახდენს სერვერი ხელმოწერის გენერირებას (3, 2) ტიპის ზღვრული ხელმოწერის სქემის გამოყენებით. თითოეული სერვერი ახდენს ნაწილობრივი ხელმოწერის $PS(m,s_i)$ გენერირებას m შეტყობინებისთვის, რისთვისაც იყენებს გასაღების ნაწილს. c გამაერთიანებელს შეუძლია ხელმოწერის $<m>_k$ გენერირება მიუხედავად იმისა, რომ სერვერი 2 არ უზრუნველყოფს ნაწილობრივ ხელმოწერას.



სურ.1.6 მაგალითი ხელმოწერის გენერირებისა ზღვრული კრიპტოგრაფიის გამოყენებით.

გარდა ამისა, სარისკო სერვერმა შესაძლოა მოახდინოს არასწორი ნაწილობრივი ხელმოწერის გენერირება. ასეთი ნაწილობრივი ხელმოწერის გამოყენება შედეგად მოგვცემს არასწორ ხელმოწერას. შესაბამისად, გამაერთიანებელს უნდა შეეძლოს

გამოთვლილი ხელმოწერის ნამდვილობის შემოწმება საჯარო გასაღების სერვისის გამოყენებით. თუ შემოწმება წარუმატებლად დასრულდება, გამაერთიანებელი ცდის t+1 ნაწილობრივი ხელმოწერის სხვა ერთობლიობას. ეს პროცესი მანამდე გრძელდება, სანამ გამაერთიანებელი მართებულ ხელმოწერას ააგებს t+1 მართებული ნაწილობრივი ხელმოწერებიდან ან გამაერთიანებელი წარუმატებლობას მოგვახსენებს. შესაძლოა შემუშავებულ იქნას აღნიშნულის მიღწევის უფრო ეფექტური გზები.

[30]-ში ავტორები გამოდიან გასაღებთა პროაქტიული განახლების წინადადებით, რათა მოხდეს მობილურ თავდამსხმელთა მოგერიება, რადგან თავდამსხმელმა იმავე ინტერვალის განმავლობისას უნდა მოახერხოს მრავლობითი ნაწილების შეპყრობა. შესაბამისად, თუ მოხდება თავდამსხმელის კონტროლს ქვეშ არსებული ძველი გასაღების განახლება, ისინი უსარგებლო გახდება ახალ გასაღებებთან, რომელთა რისკის ქვეშ დაყენება შესაძლოა თავდამსხმელმა მოახდინოს. გარდა ამისა, სისტემა შესაძლოა იმგვარად იქნას კონსტრუირებული, რომ მოახდინოს კონფიგურაციის მისადაგება ქსელის ცვლილებებისადმი. მაგალითად, გასაღების მართვის სერვისი შესაძლოა დაიწყოს (7, 3) კონფიგურაციით და მოგვიანებით მოდიფიცირებულ იქნას (4, 2) კომბინაციად, თუკი აღმოჩნდება, რომ ზოგი სერვერი სარისკოა, ხოლო სხვები – მიუწვდომელი.

1.5.1.2 თვითგამოშვებადი სერტიფიკატები

ეს გახლავთ თვითორგანიზებული მიდგომა, რომელიც საშუალებას აძლევს მოხმარებელს შექმნას, შეინახოს, გაანაწილოს და გააუქმოს საკუთარი საჯარო გასაღები რაიმე სანდო უფლებამოსილის დახმარების გარეშე, [1].

თითოეული მომხმარებელი თავად ირჩევს საკუთარ საჯარო-კერძო გასაღებთა წყვილს. როდესაც მომხმარებელს A სჭირდება მომხმარებელ B-ს კუთვნილი სერტიფიკატის აუთენტიფიკაცია, მომხმარებელი A ამოწმებს, აქვს თუ არა გაცემული ადრე სერტიფიკატი B-სთვის. თუ არა, მომხმარებელი A ამოწმებს, არსებობს თუ არა საცავში სერტიფიკატი რომელიმე ნებისმიერი მხარისათვის C, რომელსაც თავისთავად გააჩნია B-ს სერტიფიკატი. ამგვარად, მომხმარებელი A იძენს ძალმოსილი საჯარო სერტიფიკატების ჯაჭვს ისე, რომ ჯაჭვის პირველი სერტიფიკატი არის ის, რომელიც A-ს მიერ იქნა გამოშვებული. გარდა ამისა,

ჯაჭვის ყოველი დარჩენილი სერტიფიკატის შემოწმება შეიძლება ჯაჭვის წინამორბედ სერტიფიკატში შემავალი საჯარო გასაღების გამოყენებით. ამდენად, ჯაჭვის უკანასკნელი სერტიფიკატი უნდა შეიცავდეს B მომხმარებლის საჯარო გასაღებს. ამგვარად, სერტიფიკატების გადაჯაჭვა გამოიყენება მომხმარებლების საჯარო გასაღებთა აუთენტიფიკაციისთვის. თუ A-ს არ შეუძლია სერტიფიკაციული ჯაჭვის ფორმირება B-სთან, მას არ შეეძლება B-ს საჯარო გასაღების აუთენტიფიკაცია. გარდა ამისა, პრობლემა უკავშირდება კვანძის სერტიფიკატთა არქივის ჩატვირთვას აუთენტიფიკაციის ცალკეულ არხზე დამოკიდებულების გარეშე.

1.5.2 სიმეტრიულ გასაღებზე დაფუძნებული მიდგომა

წინამდებარე თავში განხილულია გასაღების მართვის სიმეტრიულ გასაღებზე დაფუძნებული სქემები, შემოთავაზებული უსადენო ქსელის სისტემებისთვის, [1]. ამ სფეროში გაწეული შრომის დიდი ნაწილი ეთმობა სენსორულ ქსელებს. როგორც ასიმეტრიულ გასაღებზე დაფუძნებული სისტემების შემთხვევებში, შეუძლებელია ინფრასტრუქტურის ჩამოყალიბება გასაღებთა მართვისთვის, რომლებიც ტრადიციული სტილით კოდირებისთვის გამოიყენება (Kerberos). აღნიშნულის მიზეზია შეზღუდულობა, რაც ზემოთ იქნა ნახსნები და რაც განსაკუთრებით მკაცრდება სენსორული ქსელების შემთხვევებში.

პრაქტიკულ გადაწყვეტილებას, როდესაც მსგავსი შეზღუდვები არსებობს, წარმოადგენს გასაღებთა ჩატვირთვა კვანძებში მანამ, სანამ კვანძების გაშვება მოხდება. თუმცა კვანძები გარკვეულ საიდუმლო ინფორმაციას შეიცავენ, აღნიშნულის გამოყენებით ისინი უსაფრთხო ინფრასტრუქტურას ქმნიან ქსელის მუშაობის დროს გამოყენებისთვის.

ზემოთხსენებულზე დამოკიდებული რამდენიმე მიდგომა იქნა შემოთავაზებული, მათ შორის მიდგომა, დაფუძნებული ყველა კვანძისთვის ერთობლივი გლობალური გასაღების გამოყენებაზე, მიდგომა, სადაც ყველა კვანძი უნიკალურ გასაღებს ინაწილებს ქსელის ერთ ან მეტ კვანძთან და მიდგომა, დაფუძნებული ყოველი კვანძის გაშვებაზე გასაღებთა შემთხვევითი ნაკრების მინიჭებით. აღნიშნული მიდგომები შესაძლოა თავისუფლად დაიყოს ორ ძირითად კატეგორიად: დეტერმინისტულ და ალბათობით სქემებად. დეტერმინისტულ სქემებს დეტერმინისტული ურთიერთობა აქვთ კვანძზე ჩატვირთულ გასაღებსა და ამ

კვანძის იდენტურობას შორის. უფრო ზუსტად რომ ვთქვათ, ქსელის ნებისმიერ ორ კვანძს შორის უსაფრთხო ლინკის არსებობა შესაძლოა ზუსტად იქნას ნაწინასწარმეტყველები. ასეთ ქსელებში კვანძების რისკის ქვეშ დაყენებამ შესაძლოა გამოიწვიოს უსაფრთხო კვანძებს შორის კომუნიკაციის დაუცველობა, თუმცა ასეთი უსაფრთხო კვანძების განსაზღვრა, რომლებზე ზემოქმედება მტრულად განწყობილი კვანძების მიერ მოხდა, ძირითადად შესაძლებელია, რომ ზუსტად განისაზღვროს. ალბათობითი სქემის შემთხვევაში კვანძებზე ჩატვირთული გასაღებები შემთხვევით არის შერჩეული. შესაბამისად, ქსელის ორ ნებისმიერ კვანძს შორის უსაფრთხო ლინკი არსებობს გარკვეული ალბათობით. თავდამსხმელის მიერ კვანძების შეპყრობა ასეთ ქსელებში ასევე გამოიწვევს უსაფრთხო კვანძებს შორის უსაფრთხო კომუნიკაციის საფრთხის ქვეშ დაყენებას, თუმცა ზუსტად განსაზღვრა უსაფრთხო კვანძებისა, რომლებზეც ზემოქმედება მოხდა, შეუძლებელი იქნება. იმ შემთხვევაში, თუ კვანძის მეზობელთა ერთობლიობა გაშვების შემდეგ ზუსტად არის ცნობილი, გასაღების წინასწარი განაწილება ჩვეულებრივ ამბად იქცევა. ასეთ შემთხვევაში, როდესაც კვანძი მოცემულია, ჩვენ უნდა მოვახდინოთ წყვილი გასაღების გენერირება, რომელსაც მოცემული კვანძი გაინაწილებს მის თითოეულ მეზობელთან, და მისი ჩატვირთვა კვანძზე, ისევე, როგორც იმ კვანძებზე, რომელთა მეზობელი გახდება მოცემული კვანძი. მიუხედავად ამისა, პრობლემას ის წარმოადგენს, რომ ასეთი დაშვება (კვანძის მეზობლების სრულყოფილი ცოდნისა) არარეალურია, რადგან ისეთი ქსელების კვანძების გაშვება, როგორიც არის სენსორული ქსელი, შემთხვევითობის საფუძველზე ხდება.

1.6 ამოცანის დასმა

მიუხედავად იმისა, რომ უსადენო LAN-ების კონცეფცია 1970-იანი წლების ბოლოდან არსებობს, WLAN ტექნოლოგიამ ძალების მოკრება მხოლოდ 1990-იანების ბოლოდან დაიწყო და დღეისათვის საყოველთაოდ გავრცელებულ ქსელურ ტექნოლოგიას წარმოადგენს. აღნიშნული ტექნოლოგიის უკანასკნელი დროის ასეთი ფეთქებადი ზრდის მიზეზები შესაძლოა მრავალ ფაქტორს მივაწეროთ, რომელთა შორისაა ტექნოლოგიური წინსვლა და, რაც ყველაზე მეტად მნიშვნელოვანია, რაიმე სახის უკაბელო კავშირისა და მობილურობის აუცილებლობა.

დღეისათვის არსებოს მრავალი სახის უკაბელო LAN ტექნოლოგია, როგორიც არის Wi-Fi, Bluetooth, HiperLAN, HomeRF და ა.შ. ყველა ეს ტექნოლოგია 2.4GHz ISM (სამრეწველო, სამეცნიერო და სამედიცინო) რადიო დიაპაზონზე ოპერირებს.

მარშრუტიზაცია ნებისმიერი ქსელის მნიშვნელოვანი ფუნქციაა, როგორც საკაბელოსი, ასევე უსადენოსი, [7]. მიუხედავად ამისა, მარშრუტიზაციის პროტოკოლებს, შექმნილს ამ ორი სახის ქსელისთვის, სრულიად განსხვავებული მახასიათებლები გააჩნია. საკაბელო ქსელების მარშრუტიზაციის პროტოკოლები, როგორც წესი, არ საჭიროებენ სისტემის შიგნით კვანძების მობილურობით მანიპულირებას. ეს პროტოკოლები ასევე არ საჭიროებენ ისეთ დიზაინს, რომ მინიმიზებულ იქნას კომუნიკაციის მიმდინარე ხარჯები, რადგან საკაბელო ქსელებს, ჩვეულებრივ, მაღალი სიხშირე გააჩნიათ. ძალიან მნიშვნელოვანია იმის გათვალისწინება, რომ საკაბელო ქსელების მარშრუტიზაციის პროტოკოლები სრულდებოდეს სანდო ობიექტებზე, კერძოდ კი მარშრუტიზატორებზე.

აღნიშნული მახასიათებლები სრულად იცვლება, როდესაც საქმე ეხება უსადენო ქსელებს. მობილურობა ასეთი ქსელის ძირითადი მახასიათებელია. რესურსის შეზღუდვები ასევე მოქმედებს მსგავსი ქსელების მარშრუტიზაციის პროტოკოლების დიზაინზე. უსადენო ქსელებს ასევე არ გააჩნიათ მარშრუტიზატორის მსგავსი სანდო ობიექტები, რადგან ითვლება, რომ ქსელის ყოველი კვანძი იღებს მონაწილეობას მარშრუტიზაციის ფუნქციაში. შესაბამისად, უსადენო ქსელის მარშრუტიზაციის პროტოკოლები სპეციფიურ დიზაინს საჭიროებენ.

ამ ორი ტიპის ქსელისთვის შემუშავებული მარშრუტიზაციის ალგორითმები შეიძლება პირობითად იყოს დაყოფილი ორ ჯგუფად: რეაქტიული და პროაქტიული. რეაქტიული პროტოკოლების შემთხვევაში მარშრუტის განსაზღვრა იწყება მხოლოდ მაშინ, როდესაც დგება ამის საჭიროება. სამარშრუტო ინფორმაცია აგრეთვე გადაიცემა მხოლოდ საჭიროების შემთხვევაში. სრულიად საპირისპირო ხასიათი გააჩნია პროაქტიულ მარშრუტიზაციას. სამარშრუტო ინფორმაცია აქ გადაიცემა მუდმივად გარკვეული ინტერვალით, ამიტომაც, როდესაც საჭირო ხდება სასარგებლო ინფორმაციის გადაცემა, მარშრუტი პრაქტიკულად უკვე დადგენილია და ამაზე დრო არ იხარჯება.

უსადენო ქსელები, ისევე როგორც ჩვეულებრივი ქსელები, ექვემდებარება მტრულ თავდასხმებს. ძალიან ხშირად ამ ტიპის ქსელებში თავდასხმა წარმოებს ზუსტად მარშრუტიზაციის პროცესის მსვლელობისას. აქ ადგილი აქვს სამარშრუტო ინფორმაციის დამახინჯებას, ტოპოლოგიის არასწორ განსაზღვრას და ა.შ. ვინაიდან უსადენო ქსელებში მარშრუტიზაციას აქვს განსაკუთრებული მნიშვნელობა ცვალებადი ტოპოლოგიიდან გამომდინარე, ამით არის გამოწვეული უსაფრთხოების ამაღლების აუცილებლობა.

ყოველივე ზემოთქმულის გათვალისწინებით მოცემულ სადისერტაციო ნაშრომში დასმულია და გადაწყვეტილია შემდეგი ამოცანა: უსადენო ქსელების მარშრუტიზაციის პროცესის უსაფრთხოების ამაღლება. ამ მიზნით შემოთავაზებულია უსაფრთხოების უზრუნველყოფის მეთოდიკა, რომელიც ეყრდნობა რეპუტაციის კონცეფციას და რეიტინგების სისტემას. აღნიშნული მეთოდიკიდან გამომდინარე შემუშავებულია ალგორითმები, რომლებიც ქსელში უზრუნველყოფენ ერთობლივ უსაფრთხოებას, და ეფუძნებიან ჯარიმის ფუნქციის გამოყენებასა და უკუკავშირის გათვალისწინებას.

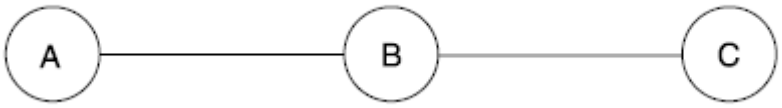
2 უსაფრთხო მარშრუტიზაცია

2.1 დისტანციურ-ვექტორული და არხის მდგომარეობის მარშრუტიზაცია

არსებობს მარშრუტიზაციის ტრადიციული პროტოკოლების ორი ძირითადი კატეგორია: დისტანციურ-ვექტორული და არხის მდგომარეობის პროტოკოლები. დისტანციურ-ვექტორული მარშრუტიზაცია არის სახეობა მარშრუტიზაციის პროტოკოლისა, რაც თავიდანვე გამოიყენება ინტერნეტში. დისტანციურ-ვექტორული მარშრუტიზაციისას თითოეული კვანძს გააჩნია ცხრილი, რომელშიც მოცემულია მანძილი მოცემული კვანძიდან ქსელის ყველა დანარჩენ კვანძამდე, [36]. როდესაც კვანძი მეზობლისგან მარშრუტიზაციის განახლებას იღებს, იგი სწავლობს ცხრილს, რათა ნახოს, შეუძლია თუ არა დანიშნულების დამატებითი პუნქტების მიღწევა მოცემული კვანძის გავლით ან თუ არსებობს გზა ზოგიერთ დანიშნულების პუნქტამდე მოცემული კვანძის გავლით, რაც არსებულ მარშრუტზე მოკლეა. თუ ეს ასეა, კვანძი ანახლებს მარშრუტიზაციის საკუთარ ცხრილს და განახლებულ ცხრილს მის ყველა მეზობელს უგზავნის. შემდეგ ისინი რიგ-რიგობით ანახლებენ საკუთარ ცხრილებს და შესაძლოა განახლება საკუთარ მეზობლებს გაუგზავნონ. DSDV (Destination-Sequenced Distance-Vector Routing) (დანიშნულება-თანმიმდევრული დისტანციურ-ვექტორული მარშრუტიზაცია) არის უკაბელო ქსელების დისტანციურ-ვექტორული მარშრუტიზაციის პროტოკოლის ტიპიური მაგალითი.

დისტანციურ-ვექტორული მარშრუტიზაციის პროტოკოლის პრობლემას წარმოადგენს ის, რომ, ჩვეულებრივ, მისი კონვერგირება ნელა ხდება. ნელი კონვერგირების მარტივი მაგალითია შემთხვევა, როდესაც კვანძი ქსელიდან არის გათიშული. განვიხილოთ სამი კვანძი, ნაჩვენები სურ. 2.1. თუ ჩავთვლით, რომ თითოეული ლინკის ღირებულება 1-ის ტოლია, მანძილი B-დან C-მდე არის 1, ხოლო მანძილი A-დან C-მდე – 2. თუ C გაითიშება ქსელიდან (მაგალითად იმ მიზეზით, რომ B-ს და C-ს დამაკავშირებელი ლინკი გატყდება), B მიხვდება, რომ იგი აღარ არის პირდაპირ დაკავშირებული C-სთან, ამდენად, B მიიღებს შეტყობინებას A-სგან იმის შესახებ, რომ მას შეუძლია დაუკავშირდეს C-ს ორის ფასად. შესაბამისად, B იძლევა შეტყობინებას C კვანძიდან 3 მანძილის შესახებ. რადგან A-ს მარშრუტი გადის B-ზე, როდესაც A გაიგებს, რომ B-დან C-მდე მანძილი ახლა 3-ის ტოლია, იგი განაახლებს საკუთარ ცხრილს და შეატყობინებს

C-დან მანძილს, რაც ოთხის ტოლია. ეს პროცესი მანამ გაგრძელდება, სანამ B-დან C-მდე მანძილი უსასრულობას მიაღწევს (ჩვეულებრივ, მარშრუტიზაციის პროტოკოლებში წარმოდგენილი დიდი რიცხვით) და A-ც და B-ც მიხვდებიან, რომ C მიუღწეველია. აღნიშნულ პროცესს შესაძლოა რამდენიმე საფეხური დასჭირდეს.



სურ. 2.1 დისტანციურ-ვექტორული მაგალითი

მარშრუტიზაციის პროტოკოლების მეორე კატეგორია, კერძოდ კი არხის მდგომარეობის მარშრუტიზაციის პროტოკოლები, ეხმიანება დისტანციურ-ვექტორული მარშრუტიზაციის პროტოკოლის შეზღუდვებს, თუმცა მას საკუთარი ნაკლოვანებები გააჩნია. არხის მდგომარეობის მარშრუტიზაციის პროტოკოლები შემდეგნაირად მოქმედებს: თითოეული კვანძი მეზობელს აღმოაჩენს ტრანსლირებული შეტყობინების გზით (ჩვეულებრივ, მას Hello შეტყობინება ეწოდება), რომელსაც ყოველი კვანძი აგზავნის, ხოლო მეზობელი ისმენს (თუ ისინი მოცემული კვანძის გადაცემის საზღვრებში ექცევა). როგორც კი კვანძი აღმოაჩენს მეზობლებს, იგი უგზავნის შეტყობინებას, რასაც, ჩვეულებრივ, არხის მდგომარეობის შეტყობინება (LSA) ეწოდება, ქსელის ყველა სხვა კვანძს, სადაც ჩამოთვლილია მეზობლების სია და ამ მეზობლების მიღწევის ღირებულება. შემდეგ თითოეულ კვანძს შეუძლია გამოიყენოს LSA მთელი ქსელის ტოპოლოგიისა და ყველა სხვა კვანძამდე მარშრუტების გამოსათვლელად. OLSR წარმოადგენს უკაბელო ქსელების არხის მდგომარეობის მარშრუტიზაციის პროტოკოლის ტიპიურ ნიმუშს, [37]. არხის მდგომარეობის მარშრუტიზაციის პროტოკოლებს ტენდენცია აქვთ უფრო სწრაფად მოახდინონ კონვერგირება, ვიდრე დისტანციურ-ვექტორულმა პროტოკოლებმა. ზემოთ აღწერილ მაგალითში (იხილე სურ. 2.1) B იძლევა შეტყობინებას, რომ C-ს არ შეუძლია B-ს მიღწევა. როგორც კი ეს ინფორმაცია ქსელში გავრცელდება, სხვა კვანძები (A-ს ჩათვლით) დაუყოვნებლივ მიხვდებიან, რომ C მიუღწეველია. ეს გაცილებით უფრო სწრაფი პროცესია, ვიდრე ის, რომელიც ადრე იყო აღწერილი დისტანციურ-ვექტორული მარშრუტიზაციის პროტოკოლებთან მიმართებაში. მიუხედავად ამისა, არხის მდგომარეობის

მარშრუტიზაციის პროტოკოლები, ჩვეულებრივ, გენერირებენ უფრო მაღალ მიმდინარე ხარჯებს, რადგან მარშრუტიზაციის ცხრილები ქსელში ვრცელდება.

როგორც დისტანციურ-ვექტორულ, ასევე არხის მდგომარეობის მარშრუტიზაციის პროტოკოლებს, გააჩნიათ საკუთარი ნაკლოვანებები და უპირატესობები. მარშრუტიზაციის სპეციფიური პროტოკოლი, რომელიც საუკეთესოდ მუშაობს, დამოკიდებულია ტოპოლოგიაზე, გამოყენებითი პროგრამის მოთხოვნასა და კვანძის შესაძლებლობებზე, [36].

2.2 პროაქტიული და რეაქტიული მარშრუტიზაციის შედარება

მარშრუტიზაციის პროტოკოლების სხვადასხვა კლასიფიკაცია ეფუძნება ხარისხს, რომლის მიხედვითაც არის შექმნილი მარშრუტები. გამომდინარე აქედან, არსებობს ორი განსხვავებული კატეგორია, კერძოდ: პროაქტიული და რეაქტიული, [7]. პროაქტიული მარშრუტიზაციის პროტოკოლებში კვანძები, ჩვეულებრივ, ცდილობენ პროაქტიულად შექმნან მარშრუტები სანამ წარმოიშვება აუცილებლობა სპეციფიური წყაროდან სპეციფიურ დანიშნულებამდე ტრაფიკის მარშრუტიზირებისა. კვანძები, ჩვეულებრივ, აღნიშნულს ახორციელებენ მარშრუტიზაციის პერიოდული განახლების გაგზავნით. გარდა ამისა, მარშრუტიზაციის განახლება იგზავნება ყოველთვის, როდესაც ტოპოლოგია იცვლება. აღნიშნული განახლება უზრუნველყოფს, რომ კვანძს უახლესი მარშრუტი ჰქონდეს სხვა კვანძებამდე. ოპტიმიზირებული არხის მდგომარეობის მარშრუტიზაციის პროტოკოლები (OLSR) მსგავსი პროტოკოლის კარგი მაგალითია.

მეორეს მხრივ, რეაქტიული პროტოკოლები ორ კვანძს შორის მარშრუტს ქმნიან მხოლოდ ისეთ დროს, როდესაც ამ ორ კვანძს შორის რეალური ტრაფიკის გაგზავნის აუცილებლობა დგება. კვანძები, რომლებიც რეაქტიული მარშრუტიზაციის პროტოკოლებს იყენებენ, აღნიშნულს, ჩვეულებრივ, ახორციელებენ ქსელში მარშრუტის მოთხოვნის შეტყობინებების გავრცელებით, რომლითაც წყაროდან დანიშნულების პუნქტამდე მარშრუტის შესახებ ინფორმაციის მოთხოვნა ხდება. მარშრუტის მოთხოვნის შეტყობინებების წარმოქმნა წყაროში ხდება და ქსელში ვრცელდება მაშინ, როდესაც წყაროს დანიშნულების პუნქტისთვის მონაცემთა გადაცემა სჭირდება. საბოლოოდ დანიშნულების პუნქტი (ან კვანძი, რომელიც უკანასკნელად დაუკავშირდა დანიშნულების პუნქტს), იღებს

მარშრუტის მოთხოვნის შესახებ შეტყობინებას და პასუხობს მას მიმართულების შესახებ აუცილებელი ინფორმაციით. ასეთი პროტოკოლის კარგ მაგალითს წარმოადგენს მიზნობრივი მოთხოვნით დისტანციურ-ვექტორული (AODV) პროტოკოლი.

2.2.1 რეაქტიული პროტოკოლები

რეაქტიული (ასევე მოთხოვნით) პროტოკოლების მიხედვით ტოპოლოგიის მონაცემები მოცემულია მხოლოდ მაშინ, როდესაც ეს აუცილებელია, [7]. ყოველთვის, როდესაც კვანძს ესაჭიროება მარშრუტის ცოდნა დანიშნულების კვანძამდე, იგი ავრცელებს ქსელში მარშრუტის მოთხოვნის შეტყობინებას. აღნიშნულს თან ახლავს დამატებითი დაგვიანება იმ ფაქტიდან გამომდინარე, რომ მარშრუტი არ არის დაუყოვნებლივ ხელმისაწვდომი.

- DSR (წყაროდან დინამიური მარშრუტიზაცია) იყენებს წყაროს მარშრუტიზაციის მექანიზმებს ანუ პაკეტის მთელი მარშრუტი პაკეტის თავსართშია. აღნიშნული თავიდან გვაცილებს მარშრუტის ციკლურობას. მარშრუტის დასადგენად კვანძი ავრცელებს მარშრუტის მოთხოვნას და ელოდება პასუხს. ნებისმიერი მიმღები კვანძი საკუთარ მისამართს უმატებს მარშრუტის მოთხოვნას და აგზავნის პაკეტს. როგორც კი პაკეტი მიაღწევს საბოლოო დანიშნულების კვანძს, ეს უკანასკნელი ახდენს მარშრუტის რევერსირებას და აგზავნის მარშრუტის საპასუხო პაკეტს. აღნიშნული შესაძლებელია, თუ MAC პროტოკოლი ორმხრივ კომუნიკაციას უშვებს. წინააღმდეგი შემთხვევისას დანიშნულების კვანძი ასრულებს სხვა მარშრუტის ძებნას უკან, შემქმნელამდე. ყოველი კვანძი ასევე ინახავს მარშრუტის საცავს, რაც თავიდან გვაცილებს იმ მარშრუტების ძებნას, რომლებიც უკვე ცნობილია. მარშრუტის სერვისის მექანიზმი საშუალებას იძლევა, რომ შემქმნელი კვანძი შეტყობინებულ იქნას მარშრუტის ლინკის დაზიანების შესახებ.

- AODV (Ad hoc On-demand Distance Vector routing) (მიზნობრივი მოთხოვნით დისტანციურ-ვექტორული მარშრუტიზაცია) წარმოადგენს დისტანციურ-ვექტორული მარშრუტიზაციის პროტოკოლს, ანუ მარშრუტები წარმოდგენილია, როგორც მიმართულებისა და მანძილის ვექტორი. იმისათვის, რომ აცილებულ იქნას ბელმან-ფორდის "უსასრულობამდე თვლის" პრობლემა და მარშრუტიზაციის ციკლურობა, შეტყობინებების კონტროლისთვის ხდება რიგითი

ნომრების გამოყენება. დანიშნულების პუნქტამდე მარშრუტის საპოვნელად კვანძი გადასცემს RREQ (Route REQuest) შეტყობინებას. RREQ-ის გადაცემა მიმღები კვანძების მიერ ხდება მანამ, სანამ იგი დანიშნულების პუნქტს ან შუალედურ კვანძს მიაღწევს, რომელსაც ახალი მარშრუტი (ანუ მარშრუტი, მასთან ასოცირებული რიგითი ნომრით) აქვს დანიშნულების პუნქტამდე. ამის შემდეგ ხდება RREP (Route REPly) შეტყობინების გაშვება დანიშნულების პუნქტის მიერ RREQ-ის შემქმნელისაკენ. RERR (Route ERRor) შეტყობინებები გამოიყენება კვანძების შეტყობინებისთვის ლინკების გატეხვის შესახებ.

- DSDV (Destination-Sequenced Distance -Vector routing) (დანიშნულება-თანმიმდევრული დისტანციურ-ვექტორული მარშრუტიზაცია) წარმოადგენს კიდევ ერთ დისტანციურ-ვექტორული მარშრუტიზაციის პროტოკოლს, რომელიც მოითხოვს, რომ ყოველმა კვანძმა მარშრუტიზაციის ცხრილი მეზობლებს შეატყობინოს. მარშრუტიზაციის ინფორმაცია შეიცავს მარშრუტის რიგით ნომერს, დანიშნულების პუნქტის მისამართს, დანიშნულების პუნქტის მანძილს ბიჯებით და რიგით ნომერს მიღებული ინფორმაციისა, რაც დაკავშირებულია დანიშნულების პუნქტთან, როგორც ეს თავად დანიშნულების პუნქტის მიერ არის მარკირებული.

2.2.2 პროაქტიული პროტოკოლები

საპირისპიროდ, პროაქტიული პროტოკოლები (მათ ასევე პერიოდულ ან ცხრილურ პროტოკოლებს უწოდებენ) ხასიათდება ტოპოლოგიის კონტროლის შეტყობინებების პერიოდული გაცვლით. კვანძები პერიოდულად ანახლებენ მარშრუტიზაციის საკუთარ ცხრილებს. შესაბამისად, კონტროლირებადი ტრაფიკი უფრო დატვირთული, მაგრამ მუდმივია, ხოლო მარშრუტები – მყისიერად ხელმისაწვდომი. განვიხილოთ მათი მაგალითები:

- OLSR (Optimized Link State Routing) (ოპტიმიზირებული არხის მდგომარეობის მარშრუტიზაცია) არის არხის მდგომარეობის მარშრუტიზაციის პროტოკოლი, რომელიც მომდევნო თავშია აღწერილი, [37];

- OSPF (Open Shortest Path First) (თავისუფალი უმოკლესი მარშრუტი პირველად) არის კიდევ ერთი არხის მდგომარეობის მარშრუტიზაციის პროტოკოლი, რომელიც გამოიყენება პაკეტური გადაცემის ქსელში ARPANET. OSPF ქსელის ტოპოლოგიის შესახებ ინფორმაციას ინახავს მონაცემთა ბაზაში, რომელიც ყოველ კვანძშია დაცული. მონაცემთა ამ ბაზიდან თითოეული კვანძი

აგებს უმოკლესი მარშრუტის ხეს დანიშნულების პუნქტებამდე პაკეტების მარშრუტიზირებისთვის. მეზობლის აღმოჩენა ხორციელდება HELLO პაკეტების გაცვლის გზით;

- FSR (Fisheye State Routing) (თევზისთვალა მდგომარეობის მარშრუტიზირება) წარმოადგენს კიდევ ერთ არხის მდგომარეობის პროტოკოლს. თითოეული კვანძი დანიშნულების პუნქტის არხის მდგომარეობის ინფორმაციას მეზობლებს გადასცემს სიხშირით, რომელიც უკუპროპორციულია დანიშნულების პუნქტის მანძილისა განსაზღვრულს ბიჯებით. სხვა სიტყვებით რომ ვთქვათ, ინფორმაცია დაშორებული კვანძების შესახებ ნაკლებად ხშირად გადაეცემა. შესაბამისად, ყოველ კვანძს ზუსტი ცოდნა გააჩნია მისი ადგილობრივი გარემოცვის შესახებ, როდესაც ცოდნა დაშორებული კვანძების შესახებ ნაკლებად ზუსტია (აქედან არის სახელი თევზისთვალა). აღნიშნული ზუსტს ხდის პაკეტების მარშრუტიზაციას წყაროსა და დანიშნულების პუნქტის სიახლოვეს. FSR სათანადოა დიდ ქსელებში ოპერირებისას;

- TBRPF (Topology dissemination Based on Reverse-Path Forwarding) (ტოპოლოგიის განფანტვა, დაფუძნებული უკუმიმართულებით გადაცემაზე) წარმოადგენს არხის მდგომარეობის პროტოკოლს, სადაც თითოეული კვანძი აგებს წარმომავლობის ხეს, იყენებს რა ტოპოლოგიის ცხრილში შენახულ ნაწილობრივ ტოპოლოგიურ ინფორმაციას. აღნიშნული ხე უზრუნველყოფს მარშრუტებს ყველა მისაწვდომ კვანძამდე და მისი გამოთვლა ხდება მოდიფიცირებული Dijkstra (დეიქსტრა) ალგორითმის გამოყენებით. თითოეული კვანძი პერიოდულად ინაწილებს მოცემული ხის ნაწილს მეზობლებთან. HELLO შეტყობინებები, რომლებიც მხოლოდ მეზობელთა სტატუსის ცვლილებას გვამცნობენ, გამოიყენება მეზობელთა აღმოჩენისთვის;

- ADV (Adaptive Distance Vector routing) (ადაპტური დისტანციურ-ვექტორული მარშრუტიზაცია) პროაქტიული პროტოკოლია, მაგრამ ზოგი რეაქტიული მახასიათებლით. ყოველი კვანძი ინაწილებს მარშრუტის ინფორმაციას მეზობლებთან, ბელმან-ფორდის განაწილებული დისტანციურ-ვექტორული ალგორითმის შესაბამისად. მიუხედავად ამისა, ADV-ში კვანძი ინახავს მარშრუტს მხოლოდ იმ კვანძებამდე, რომლებიც რაიმე აქტიური კავშირის მიმღებები არიან. გარდა ამისა, მარშრუტის განახლების სიხშირე მერყეობს ქსელის დატვირთვისა და

მობილურობიდან გამომდინარე. ამდენად, ADV სწრაფად ახდენს ადაპტირებას ქსელის დატვირთვის უცაბედი ცვლილებებისადმი;

- STAR (Source Tree Adaptive Routing) (საწყისი ხის ადაპტური მარშრუტიზაცია) იყენებს ყველა კვანძის მიერ გამოთვლილ საწყის ხეს, რათა მოახდინოს პაკეტების მარშრუტიზაცია. ყოველი კვანძი მთლიან ხეს ინაწილებს მეზობლებთან;

- LANMAR (LANdMARk მარშრუტიზაცია) წარმოადგენს მარშრუტიზაციის პროტოკოლს, რომელიც ლოგიკურ ჯგუფებად დაყოფილი მსხვილი ქსელებისთვის არის განკუთვნილი. იგულისხმება, რომ ყოველი კვანძი იდენტიფიცირებულია დამისამართების სქემით, რომელიც შეიცავს ჯგუფის საიდენტიფიკაციო ნომერს და მასპინძლის საიდენტიფიკაციო ნომერს. ახლომდებარე კვანძების მარშრუტების შესასწავლად კვანძები იყენებენ მარშრუტიზაციის რთულ პროტოკოლებს, მაგალითად FSR-ს. ყოველი ჯგუფი ირჩევს ორიენტირს. პაკეტების მარშრუტიზაცია ხდება ორიენტირების მიმართულებით, რომლებიც დანიშნულების პუნქტის ჯგუფის საიდენტიფიკაციო ნომერს შეესაბამება, შემდეგ კი უშუალოდ დანიშნულების პუნქტს გადაეცემა;

- WRP (Wireless Routing Protocol) (უსადენო მარშრუტიზაციის პროტოკოლი) ეფუძნება მიმართულების ძებნის ალგორითმს, რაც ამცირებს მარშრუტიზაციის ციკლურობის ალბათობას. WRP- ში თითოეული კვანძი მეზობლებთან ინაწილებს მარშრუტიზაციის ცხრილს დანიშნულების თითოეულ პუნქტამდე მანძილისა და მეორედან უკანასკნელამდე ბიჯის გადაცემის მეშვეობით. კვანძები დასტურს გზავნიან განახლებული მარშრუტების მიღების შემდეგ. ყოველი კვანძი ინახავს მანძილის, მარშრუტიზაციის და ლინკის ღირებულების ცხრილებს და შეტყობინების ხელახლა გადაცემის სიას;

- WIRP (Wireless Internet Routing Protocol) (უსადენო ინტერნეტ მარშრუტიზაციის პროტოკოლი) წარმოადგენს მარშუტიზაციის პროტოკოლს, შემუშავებულს უსადენო ინტერნეტ შლიუზებთან (WINGs) ოპერირებისთვის, რომელიც გაუმჯობესებულ თვითადაპტირებად მარშრუტიზატორს წარმოადგენს უსადენო გარემოში. რადიომოწყობილობა კონტროლდება FAMA-NCS პროტოკოლის მიერ, რაც აღმოფხვრავს დაფარული სადგურების პრობლემას ერთარხიან ქსელებში. ყოველი კვანძი აგებს მარშრუტიზაციის იერარქიულ ხეს და ბიჯურად უნაწილებს მას მეზობლებს, რისთვისაც ახდენს დანიშნულების

თითოეულ პუნქტამდე მხოლოდ მანძილისა და მეორედან-უკანასკნელამდე ბიჯის გადაცემას. მარშრუტის განახლება ყველა კვანძმა უნდა დაადასტუროს.

2.2.3 ჰიბრიდული პროტოკოლები

ჰიბრიდულ პროტოკოლებს აქვთ, როგორც რეაქტიული, ასევე პროაქტიული ბუნება. ჩვეულებრივ, ქსელი იყოფა რეგიონებად და კვანძი პროაქტიულ პროტოკოლს იყენებს ახლო მეზობლებთან მარშრუტიზირებისთვის, ხოლო რეაქტიულ პროტოკოლს – ამ რეგიონს გარეთ მარშრუტიზირებისთვის, [7]:

- ZRP (Zone Routing Protocol) (ზონური მარშრუტიზაციის პროტოკოლი) ყოველი კვანძისთვის განსაზღვრავს რადიუსს (ბიჯების რაოდენობით), რომლის შიგნით ხდება პაკეტების მარშრუტიზაცია პროაქტიული მარშრუტიზაციის პროტოკოლის გამოყენებით. რადიუსის გარეთ მყოფი კვანძების მარშრუტების დადგენა ხდება მარშრუტიზაციის რეაქტიული პროტოკოლის გამოყენებით. ZRP-ს სამუშაო რეჟიმი ადგილობრივად განისაზღვრება IARP-ის (IntrAzone მარშრუტიზაციის პროტოკოლი) მიერ, ხოლო ქსელის დარჩენილი ნაწილისთვის (რადიუსს გარეთ) – IERP-ის (IntErzone მარშრუტიზაციის პროტოკოლი) მიერ;

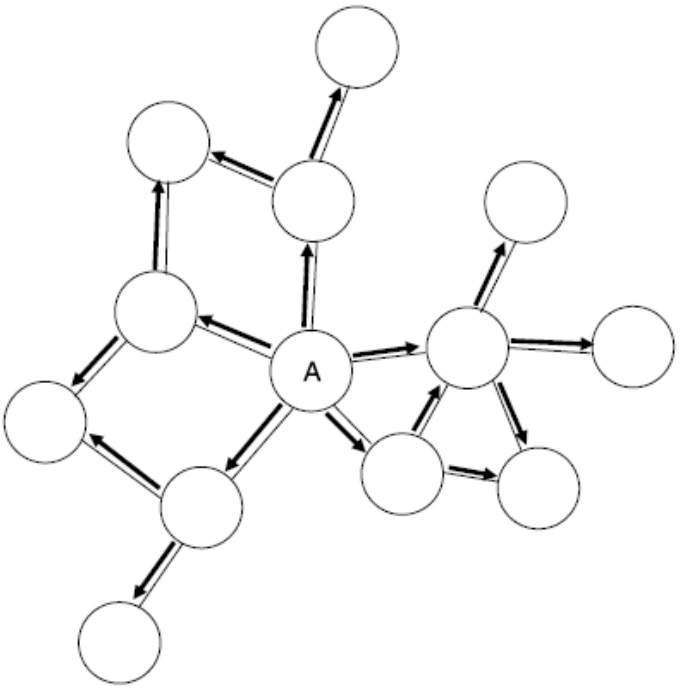
- CBRP (Cluster Based Routing Protocol) (კლასტერზე დაფუძნებული მარშრუტიზაციის პროტოკოლი) ქსელს ყოფს გადაფარვისა და გათიშვის საკვანძო კლასტერებად, სადაც თითოეული კლასტერი დიამეტრით 2 ბიჯია. თითოეული კლასტერისთვის კლასტერის სათაო კვანძს აკისრია ვალდებულება მარშრუტიზაციის დამდგენი შეტყობინებები სხვა კლასტერებს გაუცვალოს. თითოეული კლასტერის შიგნით გამოიყენება პროაქტიული მარშრუტიზაციის პროტოკოლი, როდესაც კლასტერთაშორისი მარშრუტების დადგენა რეაქტიულად ხდება, მარშრუტის მოთხოვნის გზით.

2.2.4 ოპტიმიზირებული არხის მდგომარეობის მარშრუტიზაციის პროტოკოლი (Optimized link-state routing protocol)

ოპტიმიზირებული არხის მდგომარეობის მარშრუტიზაციის პროტოკოლი (OLSR) წარმოადგენს ტიპიურ არხის მდგომარეობის მარშრუტიზაციის პროტოკოლს, რომელიც ოპტიმიზირებულ იქნა უსადენო გარემოში გამოყენებისთვის, [37, 38]. არხის მდგომარეობის მარშრუტიზაციის პროტოკოლებში კვანძები გზავნიან მარშრუტიზაციის შეტყობინებებს, სადაც მათი

უშუალო მეზობლებია ჩამოთვლილი. ეს შეტყობინებები, რომლებსაც არხის მდგომარეობის შეტყობინებები (LSA) ეწოდებათ, ქსელში ვრცელდება. რადგან უსადენო ქსელებს, ჩვეულებრივ, ხელმისაწვდომი სიხშირ შეზღუდული აქვთ, OLSR-ის კონცეფციაა - ქსელში მარშრუტიზაციის შეტყობინებების ეფექტური გავრცელება. ზემოაღნიშნული მრავალბიჯიანი გადაცემის (MPR-Multipoint Relay) კონცეფციას ეფუძნება, რაც წინამდებარე თავში იქნება აღწერილი.

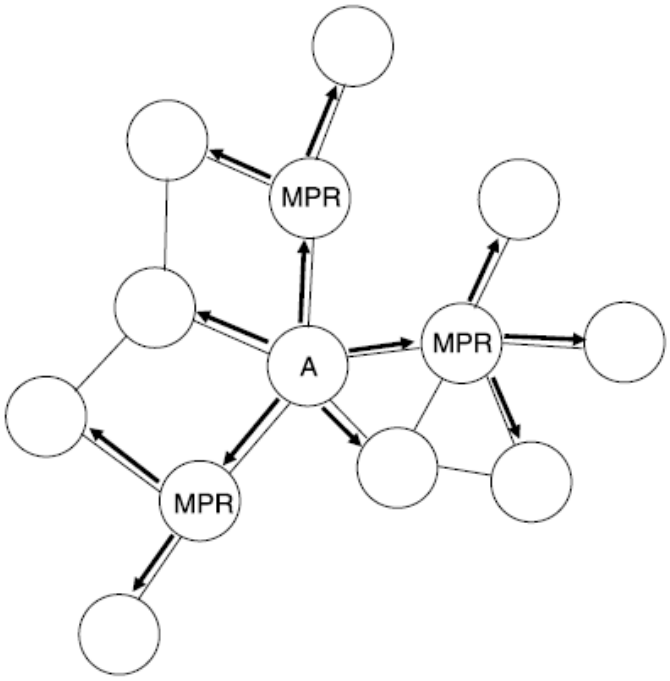
MPR-ს დანიშნულებაა არხის მდგომარეობის განახლების გავრცელების ოპტიმიზაცია. მარშრუტიზაციის ტიპიურ პროტოკოლში მარშრუტიზაციის შეტყობინებები შემდეგნაირად ვრცელდება: ყოველი კვანძი, რომელიც იღებს შეტყობინებას, თავისთავად ყველა მიმართულებით ავრცელებს მარშრუტიზაციის შეტყობინებას (გადასცემს ყველა კვანძს გადაცემის არეალში Broadcast). სურ. 2.2 გვიჩვენებს, რომ კვანძის A მიერ გენერირებული შეტყობინებები ქსელში იქნება გავრცელებული. როგორც სურათიდან ჩანს, გავრცელების ასეთი მექანიზმი არ არის საკმარისად ეფექტური, რადგან სხვადასხვა კვანძებმა ერთი და იგივე შეტყობინება შეიძლება მრავალჯერ მიიღონ.



სურ.2.2 მარშრუტიზაციის შეტყობინებების გავრცელება.

OLSR-ში შექმნილ იქნა მარშრუტიზაციის ინფორმაციის გავრცელების უფრო ეფექტური სქემები. აქ თითოეული კვანძი LSA-ს გავრცელების ამოცანას

აკისრებს მხოლოდ რამდენიმე ერთბიჯიან სიმეტრიულ მეზობელს. ამ სპეციალური კვანძების შერჩევა ხდება იმგვარად, რომ უზრუნველყოფილი იყოს LSA-ს მიერ ყველა ორბიჯიანი მეზობლის მიღწევა. LSA-ს გადაცემისთვის შერჩეულ კვანძებს MPR (MultiPoint Relay-მრავალპუნქტიანი რელე) ეწოდებათ. მაგალითად, როგორც ეს სურათზე 2.3-ია ნაჩვენები, როდესაც A გადასცემს მარშრუტიზაციის განახლებას, იგი გადასცემს ამას ყოველ ერთბიჯიან მეზობელს. ყველა ეს კვანძი იღებს და ამუშავებს შეტყობინებას, მაგრამ მხოლოდ ის კვანძები გადასცემენ განმეორებით შეტყობინებას, რომლებიც A-სთვის MPR-ს წარმოადგენენ. აღნიშნული ამცირებს განმეორებითი შეტყობინებების რაოდენობას და, ამგვარად, OLSR-ს მიერ გენერირებულ მიმდინარე ხარჯებსაც.



სურ.2.3 OLSR მარშრუტიზაციის პროტოკოლი.

OLSR-ში თითოეული კვანძი Hello შეტყობინებას გზავნის პერიოდულად (მაგალითად, ყოველ წამს) თითოეული კვანძის ინტერფეისით. Hello შეტყობინების მთავარი მიზანია საშუალება მისცეს კვანძს უშუალო (ერთბიჯიანი) მეზობელი აღმოაჩინოს. Hello შეტყობინებების გადაცემა ხდება მხოლოდ ერთბიჯიანი მეზობლებისთვის და არ გადაიცემა კვანძის ერთბიჯიან მეზობლებზე შორს. Hello შეტყობინება შეიცავს შემქმნელი კვანძის სახელს, ერთბიჯიან მეზობლებს,

რომლებიც შემქმნელმა კვანძმა უკვე აღმოაჩინა და კვანძებს, რომლებიც შემქმნელმა კვანძმა MPR-დ აირჩია. როგორც კი კვანძი გაიგებს Hello შეტყობინებას, იგი ამოწმებს, არის თუ არა შეტყობინება შექმნილი ახალი მეზობლის მიერ და თუ ასეა, იგი ანახლებს ერთბიჯიანი მეზობლების სიას. Hello შეტყობინება ასევე ძალიან მნიშვნელოვანია MPR კონცეფციის მხარდასაჭერად. თითოეული კვანძი ამოწმებს მეზობლებისგან მიღებულ Hello შეტყობინებას, რათა დაინახოს, იყო თუ არა ის არჩეული რომელიმე მეზობლის მიერ MPR-დ. თუ ასეა, კვანძმა უნდა გაავრცელოს მარშრუტიზაციის განახლება, გენერირებული იმ მეზობლების მიერ, რომლებმაც იგი MPR-დ აირჩიეს. თითოეულ კვანძს ასევე შეუძლია Hello შეტყობინების გამოყენებით გამოთვალოს, რომელი კვანძი იმყოფება მისგან ორი ბიჯის დაშორებით, რადგან თითოეული ერთბიჯიანი მეზობელი Hello შეტყობინებაში ჩამოთვლის ყველა კვანძს, რომელიც მისგან ერთი ბიჯით არის დაშორებული. თითოეული კვანძი მის MPR-ს ირჩევს გამომდინარე ორბიჯიანი მეზობლობიდან. ამდენად, თითოეული ორბიჯიანი მეზობლის მიღწევა MPR-ს მეშვეობით შეიძლება. ქსელში არხის მდგომარეობის განახლების გადაცემა ხდება შეტყობინების მეშვეობით, რომელსაც ტოპოლოგიის კონტროლის (TC) შეტყობინება ეწოდება. TC შეტყობინებების გავრცელება ქსელში ხდება, შემდეგ კი ყოველ კვანძს შეუძლია აღნიშნული ინფორმაციის გამოყენებით მარშრუტიზაციის საკუთარი ცხრილის გადათვლა. გავრცელების პროცესი ხორციელდება MPR-ების მეშვეობით, როგორც ეს სურათზე 2.3-ია ნაჩვენები. OLSR-ს შემთხვევისას არ არის აუცილებელი, რომ თითოეულმა კვანძმა ყველა მეზობელს შეატყობინოს. საკმარისია შეტყობინებულ იქნას ის კვანძი, რომელიც მოცემულმა კვანძმა MPR-დ აირჩია. OLSR ასევე შეიცავს შეტყობინებათა ორ დამატებით სახეობას: მასპინძლისა და ქსელის (HNA) (host and network association) შეტყობინებებს, რომლებიც კვანძების მიერ გარე ქსელებთან კავშირის შეტყობინებისთვის გამოიყენება ანუ ქსელებისა, რომლებიც მონაწილეობას არ იღებენ OLSR მარშრუტიზაციის პროტოკოლში, და მრავლობითი ინტერფეისის დეკლარაციის (MID) (multiple interface declaration) შეტყობინებებს, რომლებიც გამოიყენება მხოლოდ იმ კვანძების მიერ, რომლებსაც მრავლობითი ინტერფეისები გააჩნიათ და მონაწილეობას იღებენ OLSR მარშრუტიზაციის პროტოკოლში იმგვარად, რომ სხვა კვანძებს შეუძლიათ განსხვავებული ინტერფეისების ასოცირება ამ კვანძთან.

OLSR არ უზრუნველყოფს შეტყობინების აუთენტიფიკაციას და, შესაბამისად, დაუცველია თავდასხმათა სახესხვაობებისთვის.

2.3 თავდასხმები მარშრუტიზირებაზე

წინამდებარე თავი ყურადღებას ამახვილებს ზოგიერთი სახის თავდასხმაზე და ამ თავდასხმათა გავლენაზე უსადენო ქსელის მარშრუტიზაციაზე, [7]. განვიხილოთ სამი სახის თავდასხმა:

- ჭიის ხვრელი;
- ელვისებური თავდასხმა;
- "სიბილა".

2.3.1 თავდასხმა "ჭიის ხვრელი" (wormhole)

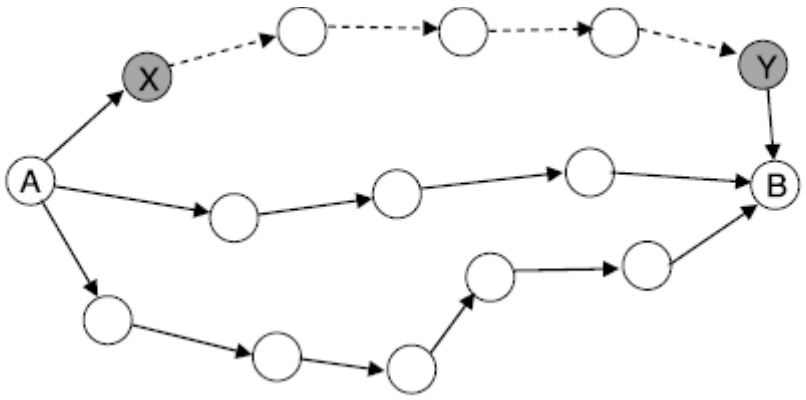
თავდასხმა "ჭიის ხვრელი", ჩვეულებრივ, მოითხოვს მიზნობრივ ქსელში სულ ცოტა ორი კონსპირაციული კვანძის არსებობას, [1,7]. მტრულად განწყობილი კვანძები გეოგრაფიულად განცალკევებული უნდა იყოს, რათა თავდასხმა ეფექტური აღმოჩნდეს. ასეთი თავდასხმისას მტრულად განწყობილი კვანძი იპყრობს პაკეტებს რაიმე პოზიციიდან და "გვირაბულად" გადასცემს მათ სხვა მტრულად განწყობილ კვანძს, რომელიც, იგულისხმება, რომ გარკვეული მანძილის დაშორებით მდებარეობს. შემდგომ, იგულისხმება, რომ მეორე მტრულად განწყობილი კვანძი "დაგვირაბებულ" პაკეტებს ადგილობრივად გადასცემს. არსებობს რამდენიმე გზა, რომლის მეშვეობითაც არის შესაძლებელი აღნიშნული გვირაბის შექმნა, [39-41].

გვირაბის შესაქმნელად პირველი მეთოდის გამოყენებისას, რაც ნაჩვენებია სურათზე 2.4, მტრულად განწყობილი კვანძი, აღნიშნული X-ით, ახდენს მეზობელ კვანძ A-დან მიღებული პაკეტის ინკაფსულირებას. ამის შემდეგ კვანძი X უგზავნის ინკაფსულირებულ პაკეტს მტრულად განწყობილ კვანძს Y. კვანძი Y ავრცელებს დეკაფსულირებულ პაკეტს მეზობლებს შორის. ამდენად, საწყისი პაკეტი, გადაცემული A კვანძის მიერ მეზობლებისთვის, ვრცელდება Y კვანძის მიერ მის მეზობლებს შორის, მათ შორისაა B კვანძიც. მაგალითად, თუ კვანძ A-ს მიერ გადაცემული (და X-ის მიერ "დაგვირაბებული") საწყისი პაკეტი Hello პაკეტი იყო, კვანძი B ამ პაკეტის მიღებისას ჩათვლის, რომ კვანძი A მისი მეზობელია, რაც არ შეესაბამება სიმართლეს. სხვა მაგალითი განვიხილოთ: თუ კვანძი A გადასცემს მარშრუტის მოთხოვნის პაკეტს კვანძ B-ს, კვანძი X შეძლებს

ასეთი პაკეტის "დაგვირაბებას" და Y კვანძისთვის მიწოდებას პაკეტის ინკაფსულირების გზით. შედეგად, მარშრუტის მოთხოვნის მოცემული პაკეტი ნაკლები ბიჯებით მიაღწევს დანიშნულების კვანძს B, ვიდრე მარშრუტის მოთხოვნის სხვა პაკეტი, რომელიც სხვა მარშრუტით მიემართება. აღნიშნული ხდება მარშრუტიზაციის ნებისმიერი უსაფრთხო პროტოკოლის გამოყენების მიუხედავად, ისეთების, რომლებიც ადრე იყო ნახსენები. შენიშვნა: X-ს და Y-ს შორის არსებულ კვანძებს, რომლებიც პაკეტს გადასცემენ, არ შეუძლიათ პაკეტის ინტერპრეტირება, რადგან იგი ინკაფსულირებულია. შესაბამისად, მათ არ შეუძლიათ ბიჯების რაოდენობის გაზრდა.

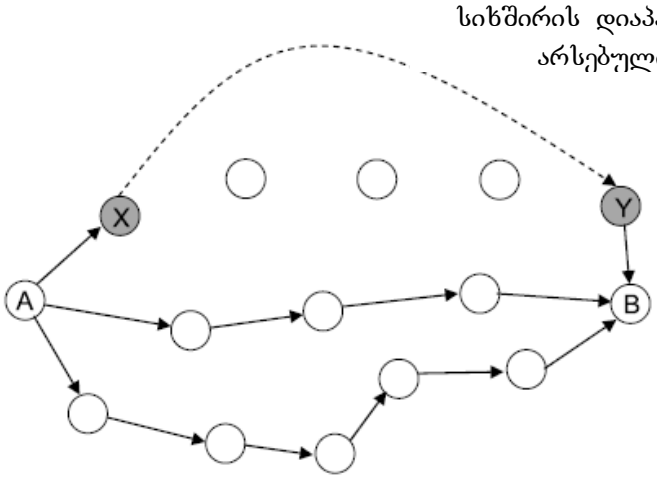
გვირაბის შესაქმნელად მეორე მეთოდის გამოყენებისას, რომელიც სურათზე 2.4-ია ნაჩვენები, იგულისხმება, რომ ორი მტრულად განწყობილი კვანძის – X-ისა და Y-სთვის - ხელმისაწვდომია სიხშირის დიაპაზონს გარეთ არსებული მაღალი სიხშირის არხი. აღნიშნული შეიძლება მიღწეულ იქნას, მაგალითად, ორ კვანძს შორის საკაბელო ლინკის არსებობით ან დიდი დიაპაზონის მქონე მაღალი სიხშირის უსადენო ლინკის მეშვეობით, რომელიც განსხვავებულ სიხშირეზე ოპერირებს. ამგვარად, აღნიშნული მეთოდი მოითხოვს სპეციალიზებული შესაძლებლობების მქონე მოწყობილობას და, ამდენად, უფრო რთულია, ვიდრე წინამორბედი მეთოდი. მოცემული შემთხვევის დროსაც ასევე, A-ს მიერ გადაცემული Hello პაკეტი შეიძლება განმეორებით იქნას გადაცემული B კვანძის ახლო მდებარეობაში. შედეგად, კვანძი B ჩათვლის, რომ კვანძი A მისი მეზობელია.

ინკაფსულირებული პაკეტები



სურ.2.4 თავდასხმა "ჭიის ხვრელი" (ინკაფსულირებული პაკეტები).

მსგავსად ამისა, კვანძ B-სთვის მარშრუტის მოთხოვნის პაკეტმა კვანძ A-დან შესაძლოა უფრო სწრაფად მიაღწიოს კვანძს B-ს (რაც დანიშნულების პუნქტს წარმოადგენს მარშრუტის მოთხოვნის პაკეტებისთვის) და, სავარაუდოდ, ნაკლები ბიჯებით, ვინაიდან ორ მტრულად განწყობილ კვანძს შორის მაღალი სიხშირის პირდაპირი ლინკი გამოიყენება. შედეგად, შეიძლება აღმოჩნდეს, რომ გვირაბის ორი სასრული წერტილი ერთმანეთთან ძალიან ახლოს არის. აღნიშნულის ნათლად დასანახად იხილეთ სურ. 2.5. აქ კვანძი B იღებს მარშრუტის სამ მოთხოვნას. ნათელია, რომ მარშრუტის მოთხოვნას, მიღებულს "ჭიის ხვრელით", ყველაზე ნაკლები ბიჯი ექნება. იქმნება შთაბეჭდილება, რომ მტრულად განწყობილი კვანძები სასარგებლო სამსახურს ეწევიან პაკეტების დაგვირაბებით. ეს ასე იქნებოდა კვანძები მსგავს სამსახურს მტრული განზრახვის გარეშე რომ ეწეოდნენ, მაგრამ მტრულად განწყობილმა კვანძებმა მოცემული თავდასხმა შესაძლოა უსადენო ქსელების სხვადასხვა პროტოკოლთა მართებული მუშაობის ძირის გამოსათხრელად გამოიყენონ. უმნიშვნელოვანეს პროტოკოლს, რომელზეც შესაძლოა ზემოაღნიშნულმა თავდასხმამ გავლენა იქონიოს, წარმოადგენს მარშრუტიზაციის პროტოკოლი, როგორც ეს ადრე განხილული მაგალითებიდან ჩანს. მონაცემთა მოგროვება, მონაცემთა მიწოდება და ა.შ. წარმოადგენს მაგალითებს სერვისებისა, რომლებიც შესაძლოა გავლენის ქვეშ მოექცნენ. თავდასხმა "ჭიის ხვრელი" შესაძლოა წარმატებული აღმოჩნდეს იმ შემთხვევაშიც, თუ მას წვდომა არ აქვს კვანძის რაიმე კრიპტოგრაფიული მასალაზე. მაგალითად, ზემოთნახსენებ სურათებზე (2.4 და 2.5) თავდასხმა "ჭიის ხვრელი" შესაძლოა წარმატებული აღმოჩნდეს სისტემის სრულუფლებიანი კვანძების (როგორიც არიან კვანძები A და B) მიერ გამოყენებული გასაღების ცოდნის გარეშეც. გარდა ამისა, ქსელის კვანძების რისკის ქვეშ დაყენება აუცილებელი არ არის. ამგვარად, იმავე ნახატებზე, კვანძები X და Y შესაძლოა წარმოადგენდნენ გარე კვანძებს, რომლებიც არ არიან რეგულარული ქსელის ნაწილი.



სურ.2.5 თავდასხმა ჭიის ხვრელი (სიხშირის დიაპაზონს გარეთ არსებული არხი).

2.3.2 ელვისებური თავდასხმა

ელვისებურ თავდასხმას გავლენა აქვს რეაქტიული მარშრუტიზაციის პროტოკოლზე, [1,7]. რეაქტიული მარშრუტიზაციის პროტოკოლის შემთხვევისას, კვანძი, რომელსაც მარშრუტი სჭირდება დანიშნულების პუნქტამდე, ქსელში მარშრუტიზაციის მოთხოვნის პაკეტებს ავრცელებს. მარშრუტიზაციის მოთხოვნის მსგავსი პაკეტების გავრცელება ქსელში კონტროლირებადი გზით ხდება. ამგვარად, თითოეული კვანძი მიღებულთაგან გზავნის მარშრუტის აღდგენის მხოლოდ პირველ პაკეტს, დანარჩენებს კი უკუაგდებს. თავდამსხმელს შეუძლია რეაქტიული მარშრუტიზაციის პროტოკოლების ამ მახასიათებლის გამოყენება. აღნიშნული სრულდება მარშრუტის მოთხოვნის პაკეტების დანიშნულების პუნქტისკენ ელვის სისწრაფით გაგზავნით. შედეგად, კვანძები, რომლებიც ასეთ "ელვისებურ" მოთხოვნას იღებენ, გადასცემენ მას და უარყოფენ მარშრუტის ყველა მოთხოვნას, რომელიც მოგვიანებით მოდის. შედეგად მიღებული მარშრუტები უკვე შეიცავენ თავდამსხმელს, სადაც თავდამსხმელს უპირატესი პოზიცია გააჩნია.

აღნიშნული თავდასხმის წამოწყება რთული არ არის. ყველაფერი, რაც ამისთვის არის საჭირო, გახლავთ ის, რომ თავდამსხმელმა მარშრუტის მოთხოვნის პაკეტების გადაცემა უფრო სწრაფად შეძლოს, ვიდრე ამას ძალმოსილი კვანძები აკეთებენ. თავდამსხმელს ამის გაკეთება "ჭიის ხვრელების" ფორმირებით შეუძლია. იგივეს გაკეთება თავდამსხმელს ასევე შეუძლია მარშრუტის მოთხოვნის პაკეტების მიღებასა და გადაცემას შორის ინტერვალის იგნორირების გზით. აღნიშნული

ინტერვალი განისაზღვრება მარშრუტიზაციის პროტოკოლების მიერ, რათა თავიდან იქნას აცილებული მარშრუტის მოთხოვნის პაკეტების კოლიზია. თავდამსხმელს ასევე შეუძლია პროტოკოლების მიერ განსაზღვრული ინტერვალის იგნორირება უსადენო არხის მისაწვდომად. ამგვარად, ყველა ასეთი შემთხვევისას შესაძლებელია მარშრუტის მოთხოვნის პაკეტებზე "ელვისებური თავდასხმა". "ელვისებურ თავდასხმაზე" რეაგირების მარტივი გზაა მარშრუტის მოთხოვნის შეტყობინებების შემთხვევითი შერჩევის დაშვება. ამდენად, იგულისხმება, რომ ყოველი კვანძი მიიღებს მარშრუტის მოთხოვნათა ზღვრულ რაოდენობას. ამის შემდეგ კვანძს შეეძლება შემთხვევითად ამოირჩიოს მარშრუტის მოთხოვნა მიღებული მოთხოვნებიდან და გაგზავნოს. თაიმაუთიც უნდა იყოს აღნიშნულთან ასოცირებული, რადგან თუ კვანძი მარშრუტების მოთხოვნის ზღვრულ რაოდენობას თაიმაუთის განმავლობისას ვერ მიიღებს, იგი შეძლებს მარშრუტის მიღებული მოთხოვნებიდან ამორჩევას. როგორც ზღვრული რაოდენობა, ისე თაიმაუთის მნიშვნელობა სიფრთხილით უნდა იქნას შერჩეული. მიუხედავად ამისა, მექანიზმი შესაძლოა ადვილად იქნას ინტეგრირებული ნებისმიერ რეაქტიული მარშრუტიზაციის პროტოკოლში "ელვისებური თავდასხმებისგან" თავის დასაცავად.

2.3.3 თავდასხმა "სიბილა" (Sybil)

"სიბილა" თავდასხმა გულისხმობს ერთი კვანძის არსებობას, რომელიც თავს წარმოაჩენს რამდენიმე კვანძად და გააჩნია შესაბამისი იდენტიფიკატორები, ანუ იდენტურობები. [18, 42-43]. დამატებითი იდენტურობების მოპოვება შესაძლებელია სხვა კვანძების მიბაძვით ან ყალბი იდენტურობების გამოყენებით. ყველა ამ იდენტურობის გამოყენება შეიძლება ერთდროულად ან დროის გარკვეული პერიოდის განმავლობისას. მოცემულმა თავდასხმამ შესაძლოა გავლენა იქონიოს უსადენო ქსელის რამდენიმე სერვისზე. მაგალითად, მან შესაძლოა გავლენა იქონიოს მრავალმიმართულებიან მარშრუტიზაციაზე, სადაც სავარაუდოდ გათიშული მიმართულებები შესაძლოა ყველა გადიოდეს მტრულად განწყობილ კვანძზე, რომელიც რამდენიმე "Sybil-იდენტურობას" იყენებს. მოცემულმა თავდასხმამ ასევე შესაძლოა გავლენა იქონიოს მონაცემთა მოგროვებაზე, სადაც ერთსა და იმავე კვანძს შეუძლია ხელი შეუწყოს მრავლობით წაკითხვას და თითოეულ შემთხვევაში განსხვავებული იდენტურობები იქნება გამოყენებული. შესაძლოა ზემოქმედების ქვეშ აღმოჩნდეს რესურსების სამართლიანი განაწილების

მექანიზმიც, რადგან კვანძს შეუძლია მოითხოვოს ერთ კუთვნილ ნაწილზე მეტი, გამოიყენებს რა სხვადასხვა "Sybil-იდენტურობას".

"Sybil-იდენტურობების" აღმოჩენის მარტივი მიდგომა შეიძლება იყოს სავჭარო გასაღების სერტიფიკატების გამოშვება ყველა იდენტურობისთვის. მიუხედავად ამისა, პრობლემას წარმოადგენს ცენტრალური უფლებამოსილი ორგანოს არსებობის აუცილებლობა, რომელიც სერტიფიკატების განაწილებას მოახდენდა. აღნიშნული მიდგომა იყენებს იმ ფაქტს, რომ ყოველი კვანძი რაღაც რესურსით არის შეზღუდული. შემდეგ ტესტით ხდება იმის გადამოწმება, გააჩნია თუ არა თითოეულ იდენტურობას ტესტირებული რესურსის სათანადო რაოდენობა. ამდენად, დაშვებულია, რომ ყოველ იდენტურობას ტესტირებული რესურსის ერთი და იგივე რაოდენობა გააჩნია. ზოგ რესურსს განეკუთვნება გამოთვლა, შენახვა და კომუნიკაცია, მაგრამ შესაძლოა ამ რესურსების ტესტირება უსადენო და სენსორულ ქსელებში სწორი არ იყოს ქსელებში სავარაუდოდ არსებული არაერთგვაროვნების მიზეზით, რის გამო ფიზიკურ მოწყობილობებს ტესტირებული რესურსის რაოდენობის განსხვავებული სიდიდეები გააჩნიათ.

კიდევ ერთ შემოთავაზებულ რესურსს რადიომიმღები წარმოადგენს. აქ დაშვება ის არის, რომ ყოველ ფიზიკურ მოწყობილობას მხოლოდ ერთი რადიომიმღები აქვს, რომელსაც არ შესწევს უნარი ერთდროულად გადასცეს და მიიღოს შეტყობინება ერთზე მეტ არხზე. ამგვარად, კვანძს, რომელსაც სურს გადაამოწმოს, წარმოადგენს თუ არა რომელიმე მისი მეზობელი "Sybil-იდენტურობას", ყველა მეზობელთან არხს გამოყოფს. მოსალოდნელია, რომ მეზობელი კვანძი შეტყობინებას გამოყოფილი არხით გადასცემს. შემამოწმებელი კვანძი მოსასმენად შემთხვევით არხს ირჩევს. თუკი არჩეულ არხზე არანაირი შეტყობინება არ ისმის, შესაბამისი კვანძის იდენტურობა Sybil-იდენტურობად ჩაითვლება. კიდევ ერთ მიდგომას წარმოადგენს Sybil-იდენტურობის დადგენა გასაღების შემთხვევითი წინასწარი განაწილების ტექნიკით. შემთხვევითი გასაღების მართვის სქემების გამოყენებისას კვანძში მის გაშვებამდე იტვირთება შემთხვევით გასაღებთა ერთობლიობა. კვანძმა, რომელიც პრეტენზიას აცხადებს რაიმე იდენტურობაზე, უნდა დაამტკიცოს ეს იდენტურობა, რისთვისაც ასევე უნდა მოახდინოს იმის დემონსტრირება, რომ მას გააჩნია მოცემული იდენტურობის შესაბამისი გასაღებები. აღნიშნულს კვანძი აკეთებს დაშიფვრის ან გაშიფვრის ოპერაციებში გასაღებით მონაწილეობის გზით. ასეთი შემთხვევისას თავდამსხმელმა

პირველ რიგში უნდა მოახდინოს მრავალი კვანძის გატეხვა, რითაც მისთვის მისაწვდომი გახდება თითოეული იდენტურობის შესაბამისი გასაღები. ამის შემდეგ თავდამსხმელმა შესაძლოა შეძლოს ყალბი იდენტურობის შექმნა.

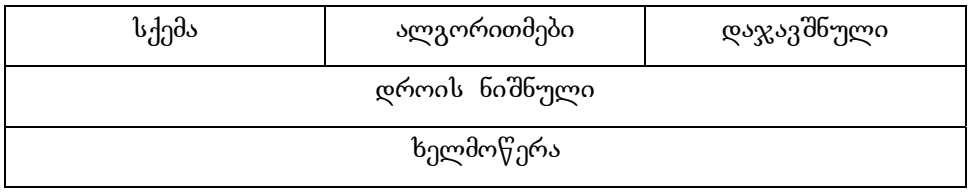
Sybil-თავდასხმის აღმოჩენის კიდევ ერთ მიდგომას წარმოადგენს ის, როდესაც სქემაში თითოეული კვანძი ცენტრალური უფლებამოსილი ორგანოს მიერ უზრუნველყოფილია უნიკალური საიდუმლო ინფორმაციით. კვანძის მიერ ამ საიდუმლო ინფორმაციიდან ხდება ჰეშ-ჯაჭვის მიღება. კვანძი ცენტრალური უფლებამოსილის მიერ ასევე უზრუნველყოფილია იდენტურობის სერტიფიკატით, რომელიც კვანძის იდენტურობას საიდუმლო ინფორმაციას აბამს. იგულისხმება, რომ კვანძმა, რომელიც პრეტენზიას აცხადებს მოცემულ იდენტურობაზე, უნდა წარმოადგინოს იდენტურობის სერტიფიკატი და უნდა დაადასტუროს, რომ იგი ფლობს უნიკალურ ინფორმაციას, სერტიფიცირებულს იდენტურობის სერტიფიკატში. დასტური შესაძლოა საჭირო იყოს ორ კვანძს შორის ყოველი ურთიერთქმედებისას.

2.4 უსაფრთხო OLSR

ზოგი სქემა OLSR-ს გაფართოებას გვთავაზობს თავდასხმებისადმი მედეგობისთვის, [44-46]. მათ მიერ შემოთავაზებულ მთავარ იდეას OLSR მარშრუტების შეტყობინებების აუთენტიფიკაციისთვის ციფრული ხელმოწერების გამოყენება წარმოადგენს, [35]. ასეთი აუთენტიფიკაცია შესაძლოა განხორციელდეს ბიჯურ (hop-by-hop) ან გამჭოლ (end-to-end) საფუძველზე, [44]. პირველ მიდგომაში თითოეული კვანძი ხელს აწერს OLSR პაკეტებზე, სანამ მათი გადაცემა ხდება (ასეთი პაკეტები შესაძლოა შეიცავდნენ OLSR შეტყობინებებს, შექმნილს სხვადასხვა კვანძების მიერ). შემდეგი ბიჯი ამოწმებს შეტყობინების აუთენტიფიკაციას, მოაშორებს წინამორბედი კვანძის ხელმოწერას და ამატებს საკუთარს. შესაბამისად, ხელმოწერა მხოლოდ იმას ამოწმებს, რომ კვანძი, რომელმაც გადააგზავნა ტრაფიკი, ის არის, რომელმაც ხელი მოაწერა შეტყობინებას, მაგრამ არ აკეთებს ორიგინალური შეტყობინების აუთენტიფიკაციას. შეტყობინებათა აუთენტიფიკაცია ეფუძნება იმ სიმეტრიულ გასაღებებს, რომლებსაც კვანძები ინაწილებენ, ხოლო ხელმოწერის შექმნა ხორციელდება რაიმე სახის ჰეშ-ფუნქციის, როგორიც არის SHA-1, გამოყენებით. გარკვეული ავტორები განიხილავენ OLSR შეტყობინებების აუთენტიფიკაციის სქემებს გამჭოლ

საფუძველზე, როდესაც OLSR შეტყობინების მიმღებ კვანძს უკეთ შეუძლია იმ კვანძის აუთენტიფიკაცია, რომელმაც საწყისი შეტყობინება შექმნა, ვიდრე შეტყობინების გადამამისამართებელი კვანძისა, [45, 46]. ქვემოთ ორივე სქემა უფრო დეტალურად არის განხილული.

პირველი სქემის შემთხვევაში ყურადღება გამახვილებულია აუთენტიფიკაციაზე ბიჯურ საფუძველზე. სურ. 2.6 გვიჩვენებს ძირითად ხელმოწერას, რომელიც თან ერთვის ყოველ OLSR პაკეტს. სქემისა და ალგორითმის ველები, ნაჩვენები სურათზე 2.6, განსაზღვრავენ ალგორითმს, რომელიც ხელმოწერის შექმნისთვის გამოიყენება (მაგალითად SHA-1). ხელმოწერის გენერირება ხდება ჰეშ-ფუნქციის (რომელიც შეტყობინებაში განსაზღვრულ სქემასა და ალგორითმს ეფუძნება) გამოყენებით OLSR პაკეტის თავსართის (header), OLSR პაკეტში შემავალი OLSR მარშრუტიზაციის შეტყობინებების, ხელმოწერის გაფართოების ველების (იხილე სურ. 2.6) მიმართ, გარდა ხელმოწერის ველებისა და განაწილებული საიდუმლო გასაღებისა. დროის ნიშნულის ველი ასევე საჭიროა მოცემულ სქემაში იმისათვის, რომ მტრულად განწყობილმა კვანძებმა ვერ შეძლონ ხელახალი თავდასხმის წამოწყება მათ მიერ სხვა მდებარეობაში გადაადგილებისა და ადრე ჩაწერილი შეტყობინებების ხელახალი გაშვების გზით. იმისათვის, რომ აღნიშნულმა სქემამ იმუშაოს, კვანძებს სჭირდებათ მათი მეზობლების მიმდინარე დროის ცოდნა. ეს არ მოითხოვს კვანძების მიერ საათების სიქნრონიზაციას. ეს მხოლოდ იმას საჭიროებს, რომ კვანძებისთვის ცნობილი იყოს დროის სავარაუდო სხვაობა მათსა და მეზობლებს შორის. ასევე იგულისხმება, რომ საათები ერთი და იმავე სიჩქარით მუშაობს. როდესაც კვანძს A მისი მეზობლის დროის დადგენა სჭირდება, იგი იწყებს დროის ნიშნულის გაცვლის პროცესს მოთხოვნის შეტყობინების გაგზავნის გზით, როგორც ეს ნაჩვენებია სურათზე 2.7, [44].



სურ.2.6 ძირითადი ხელმოწერის გაფართოება.

დანიშნულება
შემთხვევითი მნიშვნელობის “მოთხოვნის შეტყობინება”
ხელმოწერა

სურ.2.7 მოთხოვნის შეტყობინება.

დანიშნულების ველი შეიცავს IP მისამართს კვანძისა (ვთქვათ კვანძი B), რომლის დროის გაგებას ცდილობს კვანძი A. შემთხვევითი სიდიდის ველი შეიცავს შემთხვევით რიცხვს, რათა აცილებულ იქნას ხელახალი თავდასხმები, ხოლო ხელმოწერის შექმნა ხდება ჰეშ-ფუნქციის გამოყენებით, როგორც ეს ადრე იყო აღწერილი. დანიშნულების კვანძი ანუ კვანძი B, ახდენს შეტყობინების აუთენტიფიკაციას და პასუხობს მოთხოვნაზე პასუხის შეტყობინებით, რომლის ფორმატი ნაჩვენებია სურათზე 2.8.

დანიშნულება
შემთხვევითი მნიშვნელობის “მოთხოვნის შეტყობინება”
დროის ნიშნული
საპასუხო ხელმოწერა
ხელმოწერა

სურ.2.8 მოთხოვნაზე პასუხის შეტყობინება.

მოთხოვნაზე პასუხის შეტყობინება შეიცავს A კვანძის IP მისამართს, შემთხვევით რიცხვს და მის დროით ნიშნულს. საპასუხო ხელმოწერის ველის შექმნა ხორციელდება ჰეშ-ფუნქციის გამოყენებით B კვანძის IP მისამართის, შემთხვევითი მოთხოვნისა და ერთობლივი გასაღების მიმართ. ხელმოწერის ველის წარმოქმნა ხდება ჰეშ-ფუნქციის გამოყენებით მთლიანი შეტყობინებისა და ერთობლივი გასაღების მიმართ. როდესაც A კვანძი იღებს მოთხოვნაზე პასუხს B კვანძიდან, იგი პირველ რიგში ახდენს მის აუთენტიფიკაციის შემოწმებას (ჰეშ-ფუნქციის გამოყენებით და შედეგების შედარებით იმასთან, რასაც შეტყობინება შეიცავს). A კვანძი შემდეგ დროის საკუთარ ნიშნულს უგზავნის B კვანძს, რისთვისაც საპასუხო შეტყობინებას ქმნის ისე, როგორც ნაჩვენებია სურათზე 2.9.

დანიშნულება
დროის ნიშნული
საპასუხო ხელმოწერა
ხელმოწერა

სურ.2.9 პასუხის შეტყობინება.

გარკვეული ავტორები OLSR შეტყობინებების აუთენტიფიკაციისთვის გვთავაზობენ სქემას გამჭოლ საფუძველზე. სქემის მთავარი იდეა შემდეგია: როდესაც კვანძები გზავნიან OLSR შეტყობინებას, ისინი გაუმჯობესებული ხელმოწერის (ADVSIG-Advanced signature) შეტყობინებას ურთავენ თან, როგორც ეს ნაჩვენებია სურათზე 2.10, [45,46].

ხელმოწერის მეთოდი	დაჯავშნული	MSN
გლობალური დროის ნიშნული		
გლობალური ხელმოწერა		
სერტიფიკატის ხელმოწერა #1		
……		
სერტიფიკატის ხელმოწერა #n		
დასტურის დროის ნიშნული #1		
დასტურის ხელმოწერა #1		
……		
დასტურის დროის ნიშნული #n		
დასტურის ხელმოწერა #n		

სურ.2.10 ADVSIG შეტყობინების ფორმატი.

კვანძები, რომლებიც ლინკების შეტყობინებას ახდენენ, ხელს იმგვარად აწერენ, რომ შესაძლებელი იყოს შეტყობინების წარმომავლობის აუთენტიფიკაცია. ამ სქემის კიდევ ერთ მნიშვნელოვან კონცეფციას ის წარმოადგენს, რომ როდესაც კვანძი ლინკს ატყობინებს სხვა კვანძს (მაგალითად, TC შეტყობინების მეშვეობით), იგი თან ურთავს დასტურს იმისა, რომ ლინკი ნამდვილად არსებობს, რაც ამ კვანძიდან გაგზავნილი Hello შეტყობინებების მეშვეობით ხდება. OLSR-ს დაცვის აღნიშნული მიდგომა მოითხოვს გასაღების მართვის რაიმე სქემას, რაც შეიძლება

გამოყენებულ იქნას შეტყობინებათა აუთენტიფიკაციისთვის. სქემა ასევე მოითხოვს დროის ნიშნულის არსებობას, რაც საშუალებას აძლევს კვანძებს დროის ურთიერთშეთანხმებული ნიშნულები იქონიონ, როგორც ეს უკვე იყო აღწერილი წინამდებარე თავში.

სურათზე 2.10 ხელმოწერის მეთოდი განსაზღვრავს შეტყობინებათა ხელმოწერისთვის გამოყენებულ ფუნქციებს. MSN-ის (Message sequence number) ველი განსაზღვრავს შეტყობინების რიგით ნომერს, რომელსაც ADVSIG შეტყობინება მიმართავს. გლობალური დროის ნიშნულის შეტყობინება შეიცავს დროის ნიშნულს. გლობალური ხელმოწერის ველი შეიცავს OLSR შეტყობინების ხელმოწერას და თანდართულ ADVSIG შეტყობინებას. სერტიფიკატთა ხელმოწერები (#1–#n) არის Hello შეტყობინებების ხელმოწერები, რომლებიც გენერირებულია შეტყობინების გამგზავნი კვანძის მიერ და რომლისთვისაც ჯერ დასტური არ არსებობს (ვინაიდან Hello შეტყობინება არ არის მიღებული ლინკის მეორე მხარის მიერ). დასტურის დროის ნიშნული და ხელმოწერა (#1–#n) შეიცავს დროს, როდესაც ხელმოწერილი Hello შეტყობინება იქნა მიღებული ლინკის მეორე მხარის მიერ, რომელიც შეტყობინებულია Hello ან TC შეტყობინებით, და ხელმოწერას, რომელსაც აღნიშნული შეტყობინება შეიცავს (და გენერირებულია ლინკის მეორე მხარის მიერ).

2.5 უსაფრთხო არხის მდგომარეობის მარშრუტიზაციის პროტოკოლი (SLSP)

უსაფრთხო არხის მდგომარეობის მარშრუტიზაციის პროტოკოლი (SLSP) არის სქემა, შემოთავაზებული არხის მდგომარეობის მარშრუტიზაციის უსაფრთხოებისთვის, სადაც უსაფრთხოება მიიღწევა ასიმეტრიული პრიმიტივების გამოყენების გზით, [7]. SLSP გულისხმობს, რომ ქსელის ყველა კვანძს საჯარო-კერძო გასაღებთა წყვილი გააჩნია. ყოველი კვანძი გადასცემს სერტიფიცირებულ გასაღებს მისი ზონის შიგნით არსებულ ყველა კვანძს (მაგალითად, ყველა კვანძს, რომელიც მისგან R ბიჯის ფარგლებში იმყოფება). ეს გადაცემა პერიოდულია ან დამოკიდებულია იმაზე, თუ როდის მოითხოვს ამას გარემოება (მაგალითად, როდესაც მნიშვნელოვნად იცვლება ქსელის ტოპოლოგია), რაც საშუალებას აძლევს ახალ კვანძებს შევიდნენ ზონაში და დაადგინონ გასაღები. საჯარო

გასაღებთა სერტიფიცირება შესაძლოა მიღწეულ იქნას სერტიფიცირების განაწილებული უფლებამოსილი ორგანოს მეშვეობით.

პირველ ნაბიჯს ნებისმიერ არხის მდგომარეობის მარშრუტიზაციის პროტოკოლში მეზობლის აღმოჩენა წარმოადგენს. SLSP-ში მეზობლების აღმოჩენა ხდება ხელმოწერილი Hello შეტყობინებების მეშვეობით, რომლებიც შეიცავენ კვანძის IP მისამართსა და გარემოს მისაწვდომობის მართვის MedAC მისამართს. აღნიშნული მიდგომა ცდილობს უზრუნველყოს, რომ ცალკეულმა კვანძმა ვერ შეძლოს მრავლობითი კვანძების მიბაძვა, ხოლო ორმაგი IP მისამართების აღმოჩენა იოლად მოხდეს. შეტყობინებები იმ კვანძებიდან, რომლებიც მოსალოდნელ ქცევას ეწინააღმდეგებიან, შესაძლოა უარყოფილ იქნას. მოცემული მიდგომა ასევე საშუალებას აძლევს SLSP-ს შემსრულებელ კვანძებს, რომ გამოთვალონ სიჩქარე, რომლითაც სხვა კვანძები მარშრუტიზაციის შეტყობინებების გენერირებას ახდენენ. შეტყობინებებს კვანძებიდან, რომლებიც მარშრუტიზაციის ძალიან ბევრი შეტყობინების გენერირებას ახდენენ, შესაძლოა დაბალი პრიორიტეტი მიენიჭოთ, რითაც მოხდება თავის დაზღვევა ამ კვანძების მიერ მარშრუტიზაციის ძალიან ბევრი შეტყობინების გენერირების გზით DoS თავდასხმების განხორციელებისა.

როგორც ეს არხის მდგომარეობის მარშრუტიზაციის პროტოკოლებისთვის არის დამახასიათებელი, როგორც კი კვანძები მეზობლებს აღმოაჩენენ, ისინი უშუალო მეზობლებს არხის მდგომარეობის განახლების (LSU-Link State Update) შეტყობინებებს უგზავნიან. SLSP-ში LSU შეტყობინებებს ისეთი თავსართი აქვთ, როგორც ეს ნაჩვენებია სურათზე 2.11. LSU-ს შემქმნელი საკუთარი ზონის რადიუსს R განსაზღვრავს RLSU ველში. შემდეგ იგი ირჩევს შემთხვევით რიცხვს x და ჰეშ-ფუნქციას იყენებს მის მიმართ h(x). h(x) ისმება "განვლილი ბიჯების" ველში, ხოლო $h^R(x)$– "ზონის რადიუსის" ველში. მომდევნო კვანძები, რომლებიც LSU-ს გადააგზავნიან, ჰეშ-ფუნქციას იყენებენ "განვლილი ბიჯების" ველის მიმართ და მის მნიშვნელობას ახალი სიდიდით ცვლიან. შესაბამისად, ჩვენ ვიღებთ განვლილი ბიჯები=h(განვლილი ბიჯები). SLSP_LSU_SEQ შეიცავს 32-ბიტიან რიგით ნომერს, რომელიც იზრდება კვანძის მიერ LSU-ს გენერირებასთან ერთად. კვანძი LSU-ს ხელმოწერის ველს ამატებს ხელმოწერას. IP პაკეტის TTL (Time To Live, არსებობის დრო) სიდიდეს ენიჭება R-1 და, ამდენად, მარშრუტიზაციის შეტყობინებები ზონის შიგნით რჩება. ასევე შესაძლებელია, რომ სერტიფიცირებული გასაღები თავად კვანძის მიერ იყოს დართული LSU

შეტყობინებისთან (ვიდრე გამოყენებულ იქნას გასაღების პერიოდული გადაცემა). აღნიშნული უზრუნველყოფს, რომ LSU-ს მიმღებ კვანძს შეტყობინების გადამოწმება შეეძლოს.

უფრო სავარაუდოა, რომ კვანძს, რომელიც LSU-ს შემქმნელის ზონის შიგნით იმყოფება, შემქმნელის საჯარო გასაღები გააჩნია. როდესაც ასეთი კვანძი LSU შეტყობინებას მიიღებს, მას შეუძლია შეტყობინების აუთენტიფიკაციის შემოწმება. შეტყობინების აუთენტიფიკაციის შემოწმება შემდეგნაირად ხდება: კვანძი ამოწმებს "განვლილი ბიჯების" ველს. კვანძმა იცის რაოდენობა ბიჯებისა, რომელიც შეტყობინებამ უკვე გამოიარა, რაც უდრის ზონის რადიუსს R დაკლებული მიმდინარე პაკეტის TTL სიდიდე (TTL სიდიდე ერთით მცირდება თითოეულ ბიჯზე). შესაბამისად, კვანძი ჰეშ-ფუნქციას "განვლილი ბიჯების" ველის მიმართ TTL-ჯერ იყენებს და ადარებს მას ზონის რადიუსის ველს (რაც $h^R(x)$-ის ტოლია). ორი სიდიდე ტოლი უნდა იყოს:

$$h^{TTL}(hops\_traversed) = h^{TTL}[h^{R-TTL}(x)]=h^R(x) = zone\_radius$$

თუ LSU დამოწმებულია, კვანძი ამცირებს TTL-ს, ჰეშ-ფუნქციას იყენებს "განვლილი ბიჯების" ველის მიმართ და განმეორებით გადასცემს LSU-ს. LSU მხოლოდ მანამდე ინახება, სანამ LSU ლინკის მეორე მხრიდანაც იქნება მიღებული. როდესაც LSU-ს დადასტურება მოხდება, იგი გამოიყენება კვანძზე მარშრუტების განახლებისთვის, წინააღმდეგი შემთხვევისას იგი უქმდება. მაინც არსებობს შესაძლებლობა იმისა, რომ ორი კვანძი გაერთიანდეს არარსებული ლინკის შეტყობინების გასაგზავნად. SLSP-ს ასეთი თავდასხმის აღკვეთა არ შეუძლია.

ტიპი	R_{LSU}	დაჯავშნული
ზონის რადიუსი		
SLSP_LUS_SEQ		
LSU–ხელმოწერა		
განვლილი ბიჯები		

სურ.2.11 LSU შეტყობინების თავსართი.

3. ოპტიმიზირებული არხის მდგომარეობის მარშრუტიზაციის პროტოკოლი

3.1 პროტოკოლის ფუნქციონირება

წინამდებარე ქვეთავი მიზნად ისახავს ადრე განხილული OLSR პროტოკოლის ფუნქციონირების დეტალურ განხილვას. OLSR ფუნქციონალობა რეალიზებულია ძირითად ნაწილში - "გულში", რომელიც აუცილებელია პროტოკოლის ოპერირებისთვის, და დამხმარე ფუნქციების ნაკრებში, [37].

"გული" თავად განსაზღვრავს პროტოკოლს, რომელსაც საშუალება აქვს უზრუნველყოს მარშრუტიზაცია ავტონომიურ უსადენო ქსელში. თითოეული დამხმარე ფუნქცია უზრუნველყოფს დამატებით ფუნქციონალობას, რაც შესაძლოა გამოყენებულ იქნას სპეციფიურ სცენარებში, მაგალითად, ისეთი შემთხვევისას, როდესაც კვანძი უზრუნველყოფს კავშირს უსადენო ქსელსა და მარშრუტიზაციის სხვა დომენს შორის. ყველა დამხმარე ფუნქცია თავსებადია, იმ ფარგლებში, სადაც დამხმარე ფუნქციების ნებისმიერი (ქვე)ერთობლიობა შესაძლოა "გულთან" ერთად განხორციელდეს.

გარდა ამისა, პროტოკოლისთვის დასაშვებია მრავალგვაროვანი (ჰეტეროგენური) კვანძების ანუ იმ კვანძების თანაარსებობა ქსელში, რომლებიც ახორციელებენ დამხმარე ფუნქციების განსხვავებულ ქვეერთობლიობებს.

3.1.1 ძირითადი ფუნქციონალობა

OLSR-ს ფუნქციონალობა განსაზღვრავს კვანძის ქცევას, რომელიც აღჭურვილია OLSR ინტერფეისით, მონაწილეობას იღებს უსადენო ქსელში და იყენებს OLSR-ს მარშრუტიზაციის პროტოკოლად. აღნიშნული ფუნქციონალობა მოიცავს OLSR პროტოკოლის შეტყობინებათა უნივერსალურ სპეციფიკაციას და ქსელში გადაგზავნას, ლინკის ამოცნობას, ტოპოლოგიის გავრცელებას და მარშრუტის გამოთვლას, ა.შ. [37].

"გული" ძირითადად შემდეგი კომპონენტებისგან შედგება:

პაკეტის ფორმატი და გადაგზავნა

პაკეტის ფორმატის უნივერსალური სპეციფიკაცია და გავრცელების ოპტიმიზირებული მექანიზმი წარმოადგენს OLSR კონტროლის ტრაფიკის ტრანსპორტირების მექანიზმს.

ლინკის ამოცნობა

ლინკის ამოცნობა ხორციელდება HELLO შეტყობინებების პერიოდული გაგზავნით იმ ინტერფეისების საშუალებით, რომელთა მეშვეობითაც ხდება კავშირის შემოწმება. ლინკის ამოცნობის შედეგად ვიღებთ ლინკების ადგილობრივ ერთობლიობას, რომელიც აღწერს ლინკებს "ადგილობრივ ინტერფეისებსა" და "დაშორებულ ინტერფეისებს" (ანუ მეზობელი კვანძების ინტერფეისებს) შორის. თუ ლინკის (არხის) დონის მიერ უზრუნველყოფილია საკმარისი ინფორმაცია, შესაძლოა იგი იქნას გამოყენებული ადგილობრივ ლინკთა ერთობლიობის შევსებისთვის HELLO შეტყობინების ნაცვლად.

მეზობლის აღმოჩენა

თუ მოცემულია ქსელი მხოლოდ ერთი ინტერფეისის კვანძებით, კვანძმა შესაძლოა მეზობელი ერთობლიობა გამოთვალოს უშუალოდ გაცვლილი ინფორმაციიდან, როგორც ლინკის ამოცნობის ნაწილი: ერთ-ინტერფეისიანი კვანძის "ძირითადი მისამართი", განსაზღვრების მიხედვით, არის ამ კვანძზე ერთადერთი ინტერფეისის მისამართი. მრავლობით ინტერფეისიანი კვანძებისგან შემდგარ ქსელში, ძირითად მისამართზე (და, შესაბამისად, კვანძებზე) ინტერფეისის მისამართის ასახვისთვის დამატებითი ინფორმაციაა საჭირო. დამატებითი ინფორმაციის მოპოვება მრავლობითი ინტერფეისის დეკლარაციის (MID multiple interface declaration) შეტყობინებების მეშვეობით ხდება.

MPR (Multipoint Relay – მრავალპუნქტიანი რელე) შერჩევა

MPR შერჩევის მიზანს კვანძის მიერ მეზობელთა ქვეერთობლიობების იმგვარად შერჩევა წარმოადგენს, რომ გადაცემული შეტყობინება, გადაგზავნილი ან შერჩეული მეზობლების მიერ, მიღებულ იქნას 2 ბიჯით დაშორებული ყველა კვანძის მიერ. კვანძის MPR ერთობლიობის გამოთვლა იმგვარად ხდება, რომ თითოეული ინტერფეისისთვის მოცემულ პირობას აკმაყოფილებდეს. აღნიშნული გამოთვლის შესასრულებლად აუცილებელი ინფორმაციის მოპოვება HELLO შეტყობინებების პერიოდული გაცვლის გზით ხდება.

ტოპოლოგიის კონტროლის TC (Topology Control) შეტყობინების გავრცელება

ტოპოლოგიის კონტროლის შეტყობინებების გავრცელების დანიშნულებას წარმოადგენს ქსელის თითოეული კვანძის უზრუნველყოფა არხის მდგომარეობის შესახებ საკმარისი ინფორმაციით.

მარშრუტის გამოთვლა

თუ მოცემულია არხის მდგომარეობის ინფორმაცია, ისევე როგორც კვანძების ინტერფეისის კონფიგურაცია, მოპოვებული შეტყობინებათა პერიოდული გაცვლის გზით, თითოეული კვანძისთვის შესაძლებელია მარშრუტიზაციის ცხრილის გამოთვლა. ამ მექანიზმის საკვანძო იდეას MPR ურთიერთობა წარმოადგენს.

3.1.2 დამხმარე ფუნქციონალობა

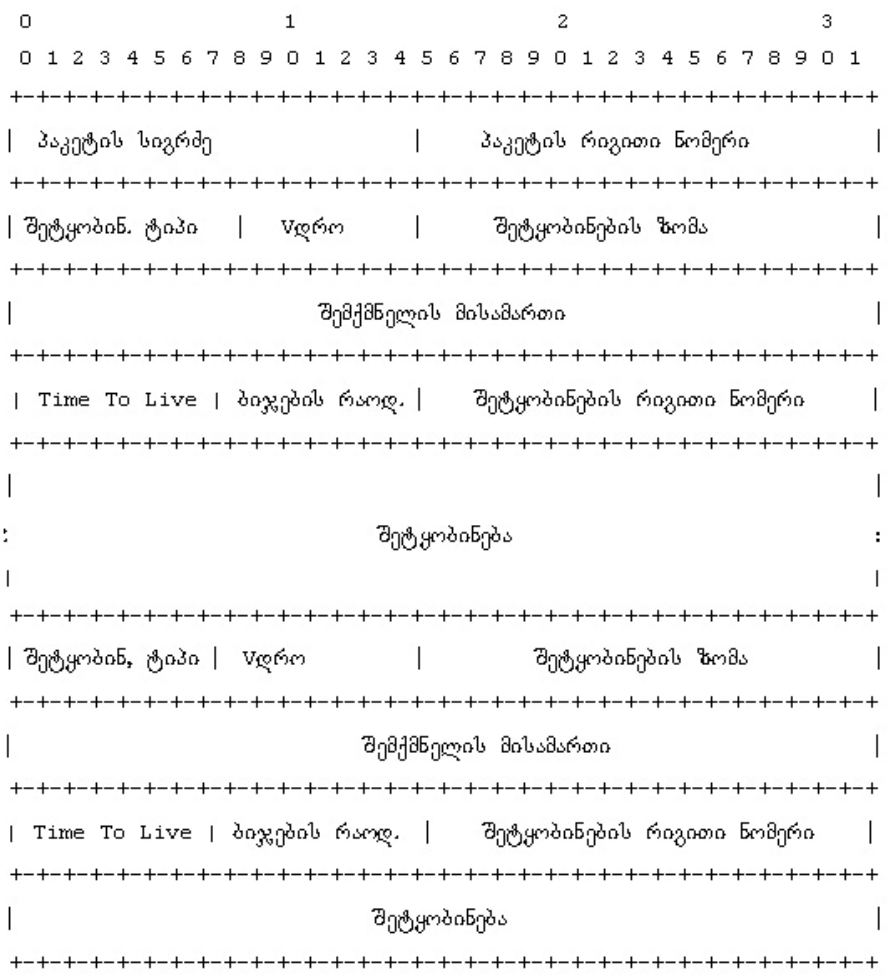
OLSR-ს საკვანძო ფუნქციონალობის გარდა, არსებობს შემთხვევები, როდესაც სასურველია დამატებითი ფუნქციონალობა. აღნიშნული მოიცავს სიტუაციებს, როდესაც კვანძს მრავლობითი ინტერფეისი გააჩნია, რომელთა ნაწილი მარშრუტიზაციის სხვა დომენში მონაწილეობს, სადაც ინტერფეისის პროგრამირება ქსელის მოწყობილობაში დამატებით ინფორმაციას უზრუნველყოფს ლინკის დონის შეტყობინებების ფორმით და სადაც სასურველია ქსელისთვის ჭარბი ინფორმაციის მიწოდება პროტოკოლის მიმდინარე ხარჯების ანგარიშით, [37]

3.1.3 პაკეტების ფორმატი და გადაგზავნა

OLSR პროტოკოლი იყენებს პაკეტის უნიფიცირებული ფორმატს, რომელიც საერთოა პროტოკოლთან დაკავშირებული ყველა მონაცემისთვის. იგი აგრეთვე აიოლებს სხვადასხვა "ტიპის" ინფორმაციის გაერთიანებას ერთ გზავნილად და, შესაბამისად, ზემოაღნიშნულის რეალიზაციისთვის ქსელის მიერ დაშვებული ფრეიმის მაქსიმალური ზომის გამოყენების ოპტიმიზაციას ემსახურება. მოცემული პაკეტები UDP დატაგრამებში თავსდება ქსელით გადაცემისთვის..

ყოველი პაკეტი ახდენს ერთი ან მეტი შეტყობინების ინკაფსულირებას. შეტყობინებები ინაწილებენ თავსართის საერთო ფორმატს, რაც საშუალებას აძლევს კვანძებს, სწორად მიიღონ და (თუ გამოიყენება) გადაგზავნონ უცნობი ტიპის შეტყობინებები. შეტყობინებები შესაძლოა გავრცელდეს მთელ ქსელში ან გავრცელება შეტყობინების შემქმნელის მიერ შეიზღუდოს კვანძებით დიამეტრის ფარგლებში (ბიჯების რაოდენობის ერთეულით). ნებისმიერი საკონტროლო

შეტყობინების გავრცელებისას დუბლირებული გადაგზავნის აღმოფხვრა ადგილობრივად მოხდება (ანუ თითოეული კვანძი შეიცავს დუბლირებულ ერთობლიობას, რათა აცილებულ იქნას OLSR კონტროლის შეტყობინების ორჯერ გაგზავნა) და მინიმიზებული იქნება მთელს ქსელში MPR-ს გამოყენების მეშვეობით, როგორც ეს ქვემოთ არის აღწერილი.



[37].

სურ.3.1 OLSR-ს ნებისმიერი პაკეტის ძირითადი ფორმატი.

პაკეტის სიგრძე

პაკეტის სიგრძე (ბაიტებში).

პაკეტის რიგითი ნომერი (PSN Packet Sequence Number)

(PSN) ყოველ ჯერზე უნდა იქნას ერთით გაზრდილი, როდესაც OLSR პაკეტის გადაცემა ხდება. თითოეული ინტერფეისისთვის ცალკე ინახება პაკეტის რიგითი ნომერი იმგვარად, რომ ინტერფეისით გადაცემული პაკეტები თანმიმდევრულად გადათვლილი იყოს. IP მისამართი ინტერფეისისა, რომლითაც

მოხდა პაკეტის გაგზავნა, შესაძლოა მიღებულ იქნას პაკეტის IP თავსართიდან. თუ პაკეტი არანაირ შეტყობინებას არ შეიცავს (მაგალითად, პაკეტის სიგრძე ნაკლები ან ტოლია პაკეტის თავსართის ზომაზე), პაკეტი ჩუმად უნდა იქნას გაუქმებული.

შეტყობინების ტიპი

მოცემული ველი გვიჩვენებს შეტყობინების ტიპს, რომელიც მოთავსებულია "შეტყობინების" ნაწილში.

Vდრო

აღნიშნული ველი გვიჩვენებს მიღებიდან რამდენი ხნის განმავლობისას უნდა ჩათვალოს კვანძმა შეტყობინებაში შემავალი ინფორმაცია ძალმოსილად, სანამ ინფორმაციის მორიგი განახლება იქნება მიღებული.

შეტყობინების ზომა

აღნიშნული გვაძლევს შეტყობინების ზომას, რაც გამოთვლილია ბაიტებით და იზომება "შეტყობინების ტიპის" ველიდან მომდევნო "შეტყობინების ტიპის" ველის დაწყებამდე (ან – თუ არ არსებობს მომდევნო შეტყობინება, პაკეტის დასასრულამდე).

შემქმნელის მისამართი

აღნიშნული ველი შეიცავს ძირითად მისამართს კვანძისა, რომელმაც თავდაპირველად მოახდინა შეტყობინების გენერირება. მოცემული ველი არ უნდა აგვერიოს წყაროს მისამართთან, რომელიც მოთავსებულია IP-ს თავსართში, და რომელიც ყოველ ჯერზე იცვლება შუალედური კვანძის ინტერფეისის შესაბამისად, რომელმაც მოახდინა შეტყობინების გადაგზავნა. შემქმნელის მისამართი არასოდეს არ უნდა შეიცვალოს გადაგზავნისას.

Time To Live

აღნიშნული ველი შეიცავს მაქსიმალურ რაოდენობას ბიჯებისა, რომლებსაც გაგზავნილი შეტყობინება გაივლის. სანამ შეტყობინება გადაიგზავნება, Time To Live 1-ით უნდა შემცირდეს. როდესაც კვანძი იღებს შეტყობინებას, რომლის Time To Live 0-ს ან 1-ს უდრის, შეტყობინება არ უნდა გადაიგზავნოს არანაირ გარემოებებში. ჩვეულებრივ, კვანძი არ მიიღებს შეტყობინებას, რომლის TTL 0-ის ტოლია. ამგვარად, აღნიშნული ველის განსაზღვრით შეტყობინების შემქმნელს შეუძლია გავრცელების რადიუსის შეზღუდვა.

ბიჯების რაოდენობა

აღნიშნული ველი შეიცავს რაოდენობას ბიჯებისა, რომლებიც შეტყობინებამ უკვე გაიარა. შეტყობინების გადაგზავნამდე ბიჯების რაოდენობა 1-ით უნდა გაიზარდოს. თავდაპირველად, შეტყობინების შემქმნელი მას 0-ად განსაზღვრავს.

შეტყობინების რიგითი ნომერი

შეტყობინების გენერირებისას "შემქმნელი" კვანძი თითოეულ შეტყობინებას უნიკალურ საიდენტიფიკაციო ნომერს ანიჭებს. აღნიშნული ნომერი იწერება შეტყობინების რიგითი ნომრის ველში. რიგითი ნომერი ერთით იზრდება თითოეული შეტყობინებისთვის, რომლის შექმნა ერთი კვანძის მიერ ხდება. შეტყობინების რიგითი ნომერი გამოიყენება იმის უზრუნველსაყოფად, რომ მოცემული შეტყობინება კვანძის მიერ ერთზე მეტად არ იქნას გადაგზავნილი.

უნდა აღინიშნოს, რომ შეტყობინებების დამუშავება და გადამისამართება ორი განსხვავებული ქმედებაა, რაც განსხვავებული წესებით არის განპირობებული. დამუშავება უკავშირდება შეტყობინების შიგთავსის გამოყენებას, როდესაც გადამისამართება დაკავშირებულია იმავე შეტყობინების გადაგზავნასთან ქსელის სხვა კვანძებისთვის. აცნობი ტიპის შეტყობინებები ამ ალგორითმის მიერ "ბრმად" არ უნდა გადამისამართდეს. გადამისამართება (და შეტყობინების მართებული თავსართის განსაზღვრა გადასამისამართებელ, ცნობილ შეტყობინებაში) წარმოადგენს ალგორითმის პასუხისმგებლობას, რომელიც განსაზღვრავს, როგორ უნდა მოხდეს შეტყობინების მართვა და, აუცილებლობისას, გადაგზავნა. აღნიშნული საშუალებას იძლევა შეტყობინების ტიპი იმგვარად განისაზღვროს, რომ გადაცემის განმავლობისას მოხდეს მისი მოდიფიცირება (მაგალითად, ასახვა მარშრუტისა, რომელიც შეტყობინებამ აირჩია). აღნიშნული ასევე MPR გავრცელების მექანიზმისთვის გვერდის ავლის საშუალებას იძლევა, თუ რაიმე მიზეზით შეტყობინების სახეობის კლასიკური გავრცელებაა აუცილებელი. ალგორითმი, რომელიც იმას განსაზღვრავს, თუ როგორ უნდა იმართოს მოცემული შეტყობინებები, უბრალოდ მოახდენს შეტყობინების გადაგზავნას, მიუხედავად MPR-სა.

შეტყობინებათა ტიპების ერთობლიობის განსაზღვრით, რაც OLSR-ს ყველა რეალიზაციამ უნდა ამოიცნოს, შესაძლებელი იქნება პროტოკოლის გაფართოვება შეტყობინებათა დამატებითი სახეობების შემოტანით, იმავდროულად

კი ძველ რეალიზაციებთან თავსებადობა შენარჩუნებული იქნება. OLSR-ს საკვანძო ფუნქციონალობისთვის აუცილებელ შეტყობინებათა სახეებია:

- HELLO შეტყობინება, რომელიც ასრულებს ლინკის ამოცნობის, მეზობლის განსაზღვრისა და MPR-ს შერჩევის ამოცანას;
- TC (Topology Control) შეტყობინებები, რომლებიც ტოპოლოგიის დეკლარაციის (არხის მდგომარეობის შეტყობინება) ამოცანას ასრულებენ;
- MID (Multiple Interface Declaration) შეტყობინებები, რომლებიც კვანძებზე მრავლობითი ინტერფეისების არსებობის დეკლარირების ამოცანას ასრულებენ.

3.2 ინფორმაციის საცავები

OLSR-ს კონტროლის შეტყობინებათა გაცვლის გზით თითოეული კვანძი ახდენს ქსელის შესახებ შემდეგი ინფორმაციის აკუმულირებას, [37]:

მრავლობითი ინტერფეისის მქონე კვანძების საინფორმაციო ბაზა. არხის ინფორმაციის ადგილობრივი ბაზა ინახავს ინფორმაციას მეზობლებთან ლინკების შესახებ.

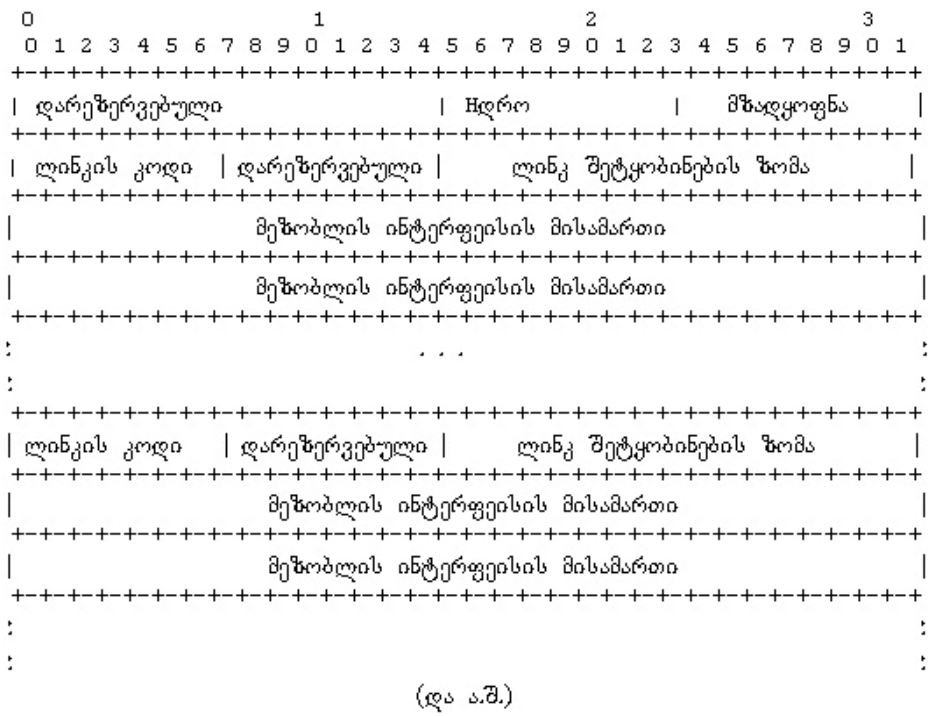
მეზობლების საინფორმაციო ბაზა - ინახავს ინფორმაციას მეზობლების, 2-ბიჯიანი მეზობლების, MPR-სა და MPR-ს შემრჩევების შესახებ.

ქსელის თითოეული კვანძი ინახავს ინფორმაციას ქსელის ტოპოლოგიის შესახებ. აღნიშნული ინფორმაციის მოპოვება ხდება TC-შეტყობინებებიდან და გამოიყენება მარშრუტიზაციის ცხრილების გამოთვლისთვის. კვანძში ტოპოლოგიის კორტეჟების ერთობლიობას "ტოპოლოგიის ერთობლიობა" ეწოდება.

3.3 Hello შეტყობინების ფორმატი და გენერირება

ადგილობრივი ლინკებისა და მეზობლების შესახებ ინფორმაციული ბაზის შევსებისთვის საერთო მექანიზმი გამოიყენება, რომელსაც სახელად HELLO შეტყობინებების პერიოდული გაცვლა ეწოდება. ამდენად, წინამდებარე თავი აღწერს HELLO შეტყობინების ზოგად მექანიზმს, რასაც თანმიმდევრობით მოყვება ლინკის ამოცნობა და ტოპოლოგიის დადგენა, [37]..

HELLO შეტყობინების ფორმატი შემდეგია:



სურ.3.2. HELLO შეტყობინების ფორმატი

აღნიშნული იგზავნება, როგორც ძირითადი პაკეტის ფორმატის მონაცემთა ნაწილი. მას თან ახლავს "მესიჯის სახეობა", რომელიც განსაზღვრულია, როგორც HELLO_MESSAGE, TTL ველი განსაზღვრულია 1-ის (ერთი) ტოლად, ხოლო Vდრო, შესაბამისად, NEIGHB_HOLD_TIME-ის სიდიდედ, რომლის განმარტებაც ქვემოთ არის მოცემული.

დარეზერვებული

აღნიშნული ველი ცარიელია და უნდა განისაზღვროს, როგორც "0000000000000".

Hდრო

აღნიშნული ველი განსაზღვრავს HELLO-ს გაშვების ინტერვალს, რასაც კვანძი იყენებს ამ კონკრეტულ ინტერფეისზე ანუ ეს არის დრო მომდევნო HELLO-ს გადაცემამდე. HELLO-ს გაშვების ინტერვალი წარმოდგენილია მანტისათი (Hდრო ველის ოთხი უდიდესი ბიტი) და ექსპონენტით (Hდრო ველის ოთხი უმცირესი ბიტი). სხვა სიტყვებით:

HELLO-ს გაშვების ინტერვალი = C*(1+a/16)*2^b

სადაც *a* არის მთელი რიცხვი, წარმოდგენილი Hდროის ველის ოთხი უდიდესი ბიტით, ხოლო b – მთელი რიცხვი, წარმოდეგნილი Hდროის ველის ოთხი უმცირესი ბიტით.

მზადყოფნა

აღნიშნული ველი განსაზღვრავს კვანძის მზადყოფნას სხვა კვანძებისთვის გადაიტანოს და გადაამისამართოს ტრაფიკი. კვანძი, რომლის მზადყოფნაა WILL_NEVER, არასოდეს არ უნდა იქნას არჩეული MPR-ად რომელიმე კვანძის მიერ. გაუცხადებლად, კვანძმა უნდა გადასცეს მზადყოფნა WILL_DEFAULT.

ლინკის კოდი

აღნიშნული ველი განსაზღვრავს ინფორმაციას ლინკის შესახებ გამგზავნის ინტერფეისსა და მეზობელი ინტერფეისებს შორის. იგი ასევე განსაზღვრავს ინფორმაციას მეზობლის სტატუსის შესახებ. ლინკის კოდები, რომლებიც კვანძისთვის უცნობია, უგულებელყოფილ არიან.

ლინკის შეტყობინების ზომა

ლინკის შეტყობინების ზომა, რომელიც ითვლება ბაიტებით და იზომება "ლინკის კოდის" ველის დასაწყისიდან "ლინკის კოდის" მომდევნო ველამდე (ან – თუ სხვა სახის ლინკები არ არსებობს – შეტყობინების ბოლომდე).

მეზობლის ინტერფეისის მისამართი

მეზობელი კვანძის ინტერფეისის მისამართი.

HELLO შეტყობინების გენერირება მოითხოვს ლინკის ერთობლიობის, მეზობელთა ერთობლიობისა და MPR ერთობლიობის გადაცემას. პრინციპში, HELLO შეტყობინება სამ დამოუკიდებელ ამოცანას ემსახურება. ესენია:

- ლინკის ამოცნობა;
- მეზობლის დადგენა;
- MPR შერჩევა.

სამივე ამოცანა ერთ გარემოში არსებულ კვანძებს შორის ინფორმაციის პერიოდულ გაცვლას ეფუძნება და ემსახურება "ადგილობრივი ტოპოლოგიის დადგენის" ერთიან მიზანს. შესაბამისად, HELLO შეტყობინების გენერირება ხდება ინფორმაციის საფუძველზე, რომელიც შენახულია ადგილობრივი ლინკების ერთობლიობაში, მეზობელთა ერთობლიობასა და MPR ერთობლიობაში.

კვანძმა ლინკის ამოცნობა ყოველ ინტერფეისზე უნდა შეასრულოს, რათა აღმოაჩინოს ლინკები ინტერფეისსა და მეზობელ ინტერფეისებს შორის.

გარდა ამისა, კვანძმა უნდა გადასცეს მისი სრული 1-ბიჯიანი გარემოცვა ყოველ ინტერფეისზე, რათა განხორციელდეს მეზობლების აღმოჩენა. ამგვარად, მოცემული ინტერფეისისთვის HELLO შეტყობინება შეიცავს აღნიშნულ ინტერფეისზე ლინკების სიას (ლინკთა ასოცირებული სახეობებით), ისევე, როგორც მთლიანი გარემოცვის სიას (მეზობელთა ასოცირებული სახეობებით).

Vდროის ველი იმგვარად არის განსაზღვრული, რომ კვანძის NEIGHB_HOLD_TIME პარამეტრის სიდიდეს შეესაბამებოდეს. Hდროის ველი იმგვარად არის განსაზღვრული, რომ კვანძის HELLO_INTERVAL პარამეტრის სიდიდეს შეესაბამებოდეს. მზადყოფნის ველი იმგვარად არის განსაზღვრული, რომ შეესაბამებოდეს კვანძის მზადყოფნას სხვა კვანძების სახელით გადაამისამართოს ტრაფიკი. კვანძმა ერთი და იგივე მზადყოფნა უნდა გადასცეს ყველა ინტერფეისს.

ერთი OLSR ინტერფეისის მქონე კვანძისთვის ძირითადი მისამართი არის უბრალოდ OLSR ინტერფეისის მისამართი ანუ კვანძისთვის ერთი OLSR ინტერფეისის ძირითადი მისამართი, რომელიც L_neighbor_iface_addr-ს შეესაბამება, არის L_neighbor_iface_addr.

3.4 მეზობლების დადგენა

მეზობლის დადგენა ავსებს მეზობლის ინფორმაციულ ბაზას და საქმე აქვს კვანძებთან და კვანძების მთავარ მისამართებთან. მეზობლის დადგენის მექანიზმს HELLO შეტყობინებების პერიოდული გაცვლა წარმოადგენს.

3.4.1 მეზობლების ერთობლიობის შევსება

კვანძი ინახავს მეზობელთა კორტეჟების ერთობლიობას, რომელიც ლინკთა კორტეჟებს ეფუძნება. აღნიშნული ინფორმაციის განახლება ლინკის ერთობლიობის ცვლილების შესაბამისად ხდება. ლინკის ერთობლიობაში ინახება ინფორმაცია ლინკების შესახებ, როდესაც მეზობელთა ერთობლიობაში ინახება ინფორმაცია მეზობლების შესახებ. ამ ორ ერთობლიობას შორის ნათელი კავშირი არსებობს, რადგან კვანძი არის მეორე კვანძის მეზობელი მხოლოდ ისეთი შემთხვევისას, როდესაც ორ კვანძს შორის სულ ცოტა ერთი ლინკი არსებობს.

3.4.2 MPR ერთობლიობის შევსება

MPR კვანძიდან ქსელში საკონტროლო შეტყობინებების გავრცელებისთვის გამოიყენება, და ამავე დროს ქსელში წარმოშობილი გადაგზავნების რაოდენობა მცირდება. ამდენად, MPR-ს ძირითად კონცეფციას წარმოადგენს შეტყობინებების გავრცელების კლასიკური მექანიზმის ოპტიმიზაცია. ქსელის თითოეული კვანძი სიმეტრიული 1-ბიჯიანი მეზობლებიდან დამოუკიდებლად ირჩევს MPR-ების საკუთარ ერთობლიობას.

MPR-ს ერთობლიობა კვანძის მიერ ისე უნდა იქნას გამოთვლილი, რომ მან MPR ერთობლიობის მეზობლების გავლით ყველა სიმეტრიულ 2-ბიჯიან მეზობელს მიაღწიოს.

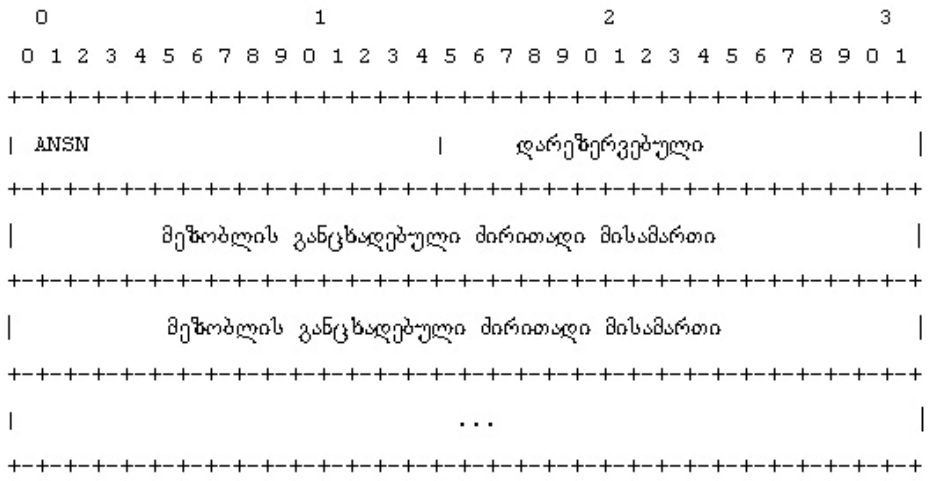
MPR ერთობლიობის ხელახალ განსაზღვრას ადგილი უნდა ჰქონდეს როდესაც ცვლილებებია აღმოჩენილი გარემოცვაში. MPR-ს განსაზღვრა ხდება ყოველი ინტერფეისისთვის, ხოლო კვანძის ყველა ინტერფეისის MPR ერთობლიობა ქმნის ამ კვანძის MPR ერთობლიობას. თუმცა არ არის არსებითი, რომ MPR-ის ერთობლიობა მინიმალური იყოს, მთავარია, რომ ყველა ზუსტი, 2-ბიჯიანი მეზობლის მიღწევა MPR-დ არჩეული კვანძებით ხდებოდეს. კვანძმა უნდა აირჩიოს MPR ერთობლიობა ისე, რომ ნებისმიერი ზუსტი, 2-ბიჯიანი მეზობელი მიღწეულ იქნას სულ ცოტა ერთი MPR კვანძით. MPR ერთობლიობის მცირე რაოდენობის შენარჩუნება უზრუნველყოფს პროტოკოლის მიმდინარე ხარჯების მინიმიზებას.

3.5 ტოპოლოგიის დადგენა

წინამდებარე თავი აღწერს, თუ ლინკის ამოცნობისა და მეზობლის განსაზღვრის მიერ მოწოდებული ინფორმაციის რა ნაწილი ვრცელდება მთელ ქსელში და როგორ გამოიყენება იგი მარშრუტის შედგენისთვის. მარშრუტიზაციის განსახორციელებლად საჭირო ინფორმაციის უზრუნველსაყოფად კვანძმა, სულ ცოტა, ლინკები უნდა გაავრცელოს საკუთარ თავსა და MPR-დ შერჩეული ერთობლიობის კვანძებს შორის.

3.5.1 TC შეტყობინების ფორმატი

TC შეტყობინების ფორმატი შემდეგია, [37]:



სურ.3.3 TC შეტყობინების ფორმატი.

აღნიშნულის გაგზავნა ხდება, როგორც ძირითადი შეტყობინების ფორმატის მონაცემთა ნაწილისა, სადაც "შეტყობინების სახეობა" განსაზღვრულია, როგორც TC_MESSAGE. time to live უნდა განისაზღვროს, როგორც 255 (მაქსიმალური სიდიდე), რათა შეტყობინების გავრცელება მთელ ქსელში მოხდეს, ხოლო Vდრო განსაზღვრული TOP_HOLD_TIME სიდიდის შესაბამისად.

შეტყობინებული მეზობლის რიგითი ნომერი Advertised Neighbor Sequence Number (ANSN)

რიგითი ნომერი ასოცირდება მეზობელთა შეტყობინებულ ერთობლიობასთან. ყოველ ჯერზე, როდესაც კვანძი ცვლილებას აღმოაჩენს შეტყობინებულ მეზობელთა ერთობლიობაში, იგი ზრდის რიგით ნომერს. ეს ნომერი იგზავნება TC შეტყიბინების ANSN ველში უახლესი ინფორმაციის ჩანაწერის შენახვის მიზნით. როდესაც კვანძი TC შეტყიბინებას იღებს, მას შეტყობინებული მეზობლის რიგითი ნომრის საფუძველზე შეუძლია გადაწყვიტოს, არის თუ არა შემქმნელი კვანძის შეტყობინებული მეზობლის შესახებ მიღებული ინფორმაცია უფრო ახალი, ვიდრე უკვე არსებული.

შეტყობინებული მეზობლის ძირითადი მისამართი

აღნიშნული ველი შეიცავს მეზობელი კვანძის მთავარ მისამართს. შემქმნელი კვანძის შეტყობინებული მეზობლების ყველა ძირითადი მისამართი ჩართულია TC შეტყობინებაში. თუ შეტყობინების დასაშვები მაქსიმალური ზომა (განსაზღვრული

ქსელის მიერ) მიღწეულია და კიდევ არსებობს შეტყობინებულ მეზობელთა მისამართები, რომელთა ჩართვა TC შეტყობინებაში ვერ მოხერხდა, მოხდება მეტი TC შეტყიბინების გენერირება, სანამ სრული შეტყობინებული მეზობლების ერთობლიობა გაიგზავნება.

დარეზერვებული

ველი ცარიელია და განისაზღვრება, როგორც "0000000000000000".

TC შეტყობინება ქსელში კვანძის მიერ იგზავნება, რათა გაცხადებულ იქნას ერთობლიობა ლინკებისა, რასაც გაცხადებულ ლინკთა ერთობლიობა ეწოდება და უნდა შეიცავდეს, სულ ცოტა, ლინკებს MPR შემრჩევთა ერთობლიობის ყველა კვანძთან ანუ მეზობლებისა, რომლებმაც გამგზავნი კვანძი MPR-ად აირჩიეს.

რიგითი ნომერი (ANSN), ასოცირებული შეტყობინებულ მეზობელთა ერთობლიობასთან, ასევე იგზავნება სიასთან ერთად. ANSN ნომერი უნდა შემცირდეს, როდესაც ლინკების მოშორება ხდება შეტყობინებულ მეზობელთა ერთობლიობიდან. შემდეგ ANSN ნომერი უნდა გაიზარდოს, როდესაც ლინკების დამატება მოხდება შეტყობინებულ მეზობელთა ერთობლიობისთვის.

ტოპოლოგიის საინფორმაციო ბაზის შესაქმნელად MPR-დ არჩეული ყოველი კვანძი აგზავნის ტოპოლოგიის კონტროლის (TC) შეტყობინებებს. TC შეტყობინებები ვრცელდება ქსელის ყველა კვანძზე.

მისამართების სია შესაძლოა ნაწილობრივი იყოს ყოველ TC შეტყობინებაში (მაგალითად, ქსელის მიერ განსაზღვრული შეტყობინების ზომის შეზღუდვების გამო), მაგრამ ანალიზი ყველა TC შეტყობინებისა, რომელიც აღწერს კვანძის შეტყობინებული ლინკების ერთობლიობას, უნდა დასრულდეს განახლების კონკრეტული პერიოდის (TC_INTERVAL) განმავლობისას. ამ TC შეტყობინებების მიერ ქსელში გავრცელებული ინფორმაცია ხელს უწყობს თითოეულ კვანძს გამოთვალოს საკუთარი მარშრუტიზაციის ცხრილი.

როდესაც კვანძის გაცხადებული ლინკების ერთობლიობა ცარიელდება, ამ კვანძმა მაინც უნდა გააგზავნოს (ცარიელი) TC შეტყობინებები t ხანგრძლივობის განმავლობისას, რაც მანამდე გაშვებული TC შეტყობინების "ძალმოსილების დროის" ტოლია (ჩვეულებრივ, იგი ტოლი იქნება TOP_HOLD_TIME-ისა), რათა მოხდეს წინამორბედი TC შეტყობინებების არაძალმოსილად ცნობა. შემდეგ მან უნდა შეწყვიტოს TC შეტყობინებების გაგზავნა, სანამ რომელიმე კვანძი არ ჩაჯდება მის შეტყობინებულ ლინკთა ერთობლიობაში.

კვანძმა შესაძლოა გადასცეს დამატებითი TC შეტყობინებები, რათა გაზარდოს საკუთარი რეაგირებადობა ლინკის ხარვეზებისადმი. როდესაც იცვლება MPR შემრჩევთა ერთობლიობა და ეს ცვლილება შესაძლოა ლინკის ხარვეზს მივაწეროთ, TC შეტყობინება უნდა გაიგზავნოს ინტერვალის შემდეგ, რომელიც ნაკლებია TC_INTERVAL-ზე.

TC შეტყობინებების გაგზავნა და გადაგზავნა MPR-ების მიერ ხდება, რათა შეტყობინება მთელ ქსელში გავრცელდეს. TC შეტყობინებების გადაგზავნა უნდა მოხდეს "გადაგზავნის სტანდარტული ალგორითმის" შესაბამისად. TC შეტყობინების მიღების შემდეგ "ძალმოსილების დრო" უნდა იქნას გამოთვლილი შეტყობინების თავსართის V დროის ველიდან.

3.6 უსაფრთხოების მოსაზრებები

დღესდღეობით OLSR არ განსაზღვრავს უსაფრთხოების რაიმე სპეციფიურ ღონისძიებებს. როგორც პროაქტიული მარშრუტიზაციის პროტოკოლი, OLSR სხვადასხვა თავდასხმების თავიდან აცილებას ისახავს მიზნად. წინამდებარე თავში სხვადასხვა შესაძლო ნაკლოვანებებია განხილული.

3.6.1 კონფიდენციალობა

OLSR პროაქტიული პროტოკოლი პერიოდულად ავრცელებს ტოპოლოგიის ინფორმაციას. ამიტომ თუ გამოიყენება დაუცველ უსადენო ქსელში, ქსელის ტოპოლოგიის დანახვა შეუძლია ყველას ვინც კი უსმენს OLSR-ის მაკონტროლებელ შეტყობინებებს.

ისეთ სიტუაციებში, როდესაც ქსელის ტოპოლოგიას დიდი მნიშვლელობა ენიჭება, გამოიყენება კრიპტოგრაფიული მექანიზმები, როგორიცაა OLSR-ის მაკონტროლებელ შეტყობინებების PGP-ით ან რომელიმე სხვა საზოგადო გასაღებით კოდირება, რათა უზრუნველყოფილ იქნას მაკონტროლებელი ტრაფიკის კონფიდენციალობა.

3.6.2 მთლიანობა

OLSR-ში თითოეულ კვანძს ტოპოლოგიის ინფორმაცია ქსელში შეაქვს HELLO შეტყობინებებისა და, ზოგი კვანძისთვის, TC შეტყობინების გაგზავნის გზით. თუ ზოგი კვანძი რაიმე მიზეზით, მტრული განწყობის ან ხარვეზით

ფუნქციონირების გამო, შეიტანს არასწორ კონტროლირებად ტრაფიკს, ქსელის მთლიანობა შესაძლოა დარღვეულ იქნას.

შესაბამისად, რეკომენდებულია შეტყობინების აუთენტიფიკაცია. შესაძლოა ადგილი ჰქონდეს სხვადასხვა სიტუაციებს, როგორიცაა მაგალითად:

1. კვანძი ახდენს TC (ან HNA) შეტყობინების გენერირებას, აცხადებს რა ლინკებს არამეზობელ კვანძებთან;

2. კვანძი ახდენს TC (ან HNA) შეტყობინების გენერირებას და ემსგავსება სხვა კვანძს;

3. კვანძი ახდენს HELLO შეტყობინების გენერირებას, აცხადებს რა ლინკებს არამეზობელ კვანძებთან;

4. კვანძი ახდენს HELLO შეტყობინების გენერირებას და ემსგავსება სხვა კვანძს;

5. კვანძი გადაგზავნის კონტროლის შეცვლილ შეტყობინებას;

6. კვანძი არ გადასცემს კონტროლის შეტყობინებებს;

7. კვანძი სწორად არ ირჩევს მრავალპუნქტიან რელეებს;

8. კვანძი კონტროლის შეტყობინებებს შეუცვლელად გადაგზავნის, მაგრამ არ გზავნის ცალმხრივ მონაცემთა ტრაფიკს;

9. კვანძი ხელახლა "კითხულობს" კონტროლის ტრაფიკის ჩაწერილ შეტყობინებას სხვა კვანძიდან. კონტრლონისძიების სახით შესაძლოა გამოყენებულ იქნას საკონტროლო შეტყობინებების შემქმნელი კვანძის აუთენტიფიკაცია (2,4 და 5 სიტუაციებისთვის) და ინდივიდუალური ლინკების გაცხადება საკონტროლო შეტყობინებებში (1 და 3 სიტუაციებისთვის). მიუხედავად ამისა, კვანძების დასაცავად ძველი (სწორად აუთენტიზირებული) ინფორმაციის განმეორებისგან (სიტაუცია 9), აუცილებელია დროებითი ინფორმაცია, რაც საშუალებას მისცემს კვანძს პოზიტიურად მოახდინოს ასეთი დაგვიანებული შეტყობინებების იდენტიფიცირება.

ზოგადად, ციფრული ხელმოწერა და უსაფრთხოების სხვა აუცილებელი ინფორმაცია შესაძლოა ცალკე OLSR შეტყობინების სახით იქნას გაგზავნილი, რათა, სურვილის შემთხვევაში, "დაცულ" და "დაუცველ" კვანძებს ერთსა და იმავე ქსელში თანაარსებობის საშუალება ჰქონდეთ.

უფრო კონკრეტულად, შესაძლოა ჩამოყალიბებულ იქნას სრული OLSR საკონტროლო შეტყობინებების აუთენტიფიკაცია IPsec აუთენტიფიკაციის

თავსართის გამოყენებით, სადაც ინდივიდუალური ლინკების აუთენტიფიკაცია (სიტუაციები 1 და 3) მოითხოვს უსაფრთხოების დამატებითი ინფორმაციის განაწილებას.

მნიშვნელოვანია ის, რომ ყველა საკონტროლო შეტყობინება OLSR-ში გადაეცემა ან გარემოცვის ყველა კვანძს (HELLO შეტყობინება) ან ქსელის ყველა კვანძს (მაგალითად, TC შეტყობინება).

მაგალითად, საკონტროლო შეტყობინება OLSR-ში ყოველთვის წარმოადგენს Point-to-Multipoint (ერთიდან მრავალს) გადაცემას. აქედან გამომდინარე, მნიშვნელოვანია ის, რომ აუთენტიფიკაციის გამოყენებული მექანიზმი საშუალებას იძლეოდეს, რომ ნებისმიერმა მიმღებმა კვანძმა შეძლოს შეტყობინების აუთენტიფიკაციის რატიფიცირება. ანალოგიის სახით, თუ გვაქვს ტექსტის ბლოკი, ხელმოწერილი PGP კერძო გასაღებით, ნებისმიერს შეუძლია ტექსტის აუთენტიფიკაციის დამოწმება შესაბამისი საჯარო გასაღებით.

4. უსადენო ქსელებში მარშრუტიზაციის უსაფრთხოების ამაღლება

4.1 უსადენო ქსელებისა და მათი მარშრუტიზაციის უსაფრთხოების მდგომარეობის მოკლე დახასიათება

უსადენო ქსელებში, ჩვეულებრივი ქსელებისგან განსხვავებით, თავდასხმებთან დაკავშირებით ადგილი აქვს მომატებული რისკის ფაქტორს, რაც გამოწვეულია შემდეგი ძირითადი მიზეზებით: უსადენო ქსელებში არ არსებობს ფილტრი, რომელიც შეიძლება იყოს გამოყენებული თავდასხმებისაგან დასაცავად; არ არსებობს სერვერი, რომელიც მომატებული ნდობის ფაქტორით ხასიათდება; უსადენო ქსელები ხასიათდება ობიექტების მუდმივი მოძრაობით და ამასთან ერთად არ არსებობს ფიზიკური არხები; ამ არხების არ არსებობის გამო ინფორმაცია გადაიცემა ეთერის საშუალებით, რაც თავისთავად აგრეთვე საშიშროებას წარმოადგენს, ვინაიდან თავდასხმები იწყება ზუსტად არხის მოსმენიდან.

ყველაფერი ზემოთნახსენები დამატებითი ნაკლოვანებების წყაროს წარმოადგენს და, შესაბამისად, უსადენო ქსელებში ინფრასტრუქტურულ/საკაბელო სტრუქტურას მიღმა უსაფრთხოების არასტანდარტულ გადაწყვეტილებებს მოითხოვს. ფიქსირებული ინფრასტრუქტურის არარსებობის პირობებში, რაც არასანდო კვანძების იდენტიფიცირებისა და იზოლირების გზით დაცვის ხაზს აყალიბებს, შესაძლებელია, რომ მარშრუტიზაციის პროტოკოლების მიერ გენერირებული საკონტროლო შეტყობინებები, მაგალითად, მეზობლის შეტყობინება ან არხის მდგომარეობის მონაცემი, დაზიანებულ და კომპრომეტირებულ იქნას და ამგვარად, საფრთხის ქვეშ დააყენოს კომუნიკაცია ქსელში.

რაც შეეხება მარშრუტიზაციას უსადენო ქსელებში – როგორც ჩვენ აღვნიშნეთ აქ არსებობს მთელი რიგი მარშრუტიზაციის პროტოკოლებისა, შემუშავებული უსადენო ქსელებისათვის, მაგრამ მათში პრაქტიკულად არ არის გათვალისწინებული უსაფრთხოების საკითხები. თუ ჩვენ გადავხედავთ არსებულ სამუშაოებს ამ მიმართულებით, აღმოვაჩენთ, რომ მათი უმრავლესობა ეყრდნობა კრიპტოგრაფიის მეთოდებს, გასაღების გამოყენებასა და გასაღების მენეჯმენტს.

უსადენო ქსელების მარშრუტიზაციის პროტოკოლების მრავალ შემოთავაზებას შორის ოპტიმიზირებული არხის მდგომარეობის მარშრუტიზაციის (OLSR) პროტოკოლი, საიმედო შესრულებას გვთავაზობს ქსელის სიხშირის,

აუცილებელი მიმდინარე ხარჯებისა და მიწოდებული ტრაფიკის კუთხით, თუმცა ამას აკეთებს უსაფრთხოების ამოცანათა ფართო სპექტრის უგულებელყოფის ხარჯზე, რასაც უმეტესად კავშირი აქვს ტოპოლოგიის ინფორმაციის აუცილებელ გაცვლასთან და საბაზისო დაშვებასთან, რომ ყველა კვანძი კეთილგანწყობილია. ამიტომაც აუცილებელი ხდება დამატებითი ზომების მიღება უსადენო ქსელში უსაფრთხოების უზრუნველსაყოფად.

4.2. OLSR-ს მარშრუტიზაციის პროტოკოლის ფუნქციონირების ძირითადი პრინციპები და უსაფრთხოების ნაკლოვანებები

განვიხილოთ OLSR პროტოკოლის ძირითადი ფუნქციონირება. როგორც პროაქტიული პროტოკოლი, იგი პერიოდულად ახდენს მარშრუტიზაციის ინფორმაციის გაცვლას და საჭიროების შემთხვევაში მარშრუტები დაუყოვნებლივ ხელმისაწვდომი აქვს. როგორც არხის მდგომარეობის პროტოკოლი, იგი ინახავს ქსელის ტოპოლოგიის ინფორმაციას, მოპოვებულს მარშრუტიზაციის კონტროლის ტრაფიკიდან, რაც გამოიყენება საუკეთესო მარშრუტის განსაზღვრისათვის ქსელის დანიშნულების ყოველ პუნქტამდე.

OLSR, რეალურად, გვთავაზობს უფრო მეტს, ვიდრე მხოლოდ არხის მდგომარეობის პროტოკოლი, რადგან იგი შემდეგ მახასიათებლებს მოიცავს:

- *ქსელში ნაკადის მინიმიზება* მხოლოდ შერჩეული კვანძების ერთობლიობის გამოყენებით, რომლებსაც *მრავალპუნქტიანი რელეები* (MPR) ეწოდებათ, რათა მათი მეშვეობით მოხდეს შეტყობინებების ქსელში გავრცელება;
- *საკონტროლო პაკეტების ზომის შემცირება* კვანძის მხოლოდ იმ მეზობლებთან არსებული ლინკების ქვეერთობლიობის გაცხადებით, რომლებიც მისი *მრავალპუნქტიან რელეებად შემრჩევს* (MPR შემრჩევი) წარმოადგენენ.

პროტოკოლი იყენებს არხის მდგომარეობის პაკეტების გადაგზავნის ეფექტურ მექანიზმს, რასაც *მრავალპუნქტიანი გადაცემა* ეწოდება. აღნიშნული მექანიზმი დაფუძნებულია იმაზე, რომ ყოველ კვანძს მეზობელი კვანძების ქვეერთობლიობა იმგვარად ჰქონდეს არჩეული, რომ ამ ქვეერთობლიობამ უზრუნველყოს კავშირი ყველა ორბიჯიან მეზობელთან. ამ ქვეერთობლიობის კვანძებს *მრავალპუნქტიანი რელეები* (MPR) ეწოდებათ, ხოლო ქვეერთობლიობა არის *მრავალპუნქტიან რელეთა ერთობლიობა* (MPR ერთობლიობა). იმ მეზობლებს,

რომლებიც მოცემულ კვანძს MPR-დ ირჩევენ, მოცემული კვანძის *MPR შემრჩევთა ერთობლიობა* ეწოდებათ. კონტროლის ტრაფიკის გაგზავნისთვის MPR-ის გამოყენება შედეგად გვაძლევს შეზღუდულ ნაკადს, ნაცვლად სრული კვანძიდან-კვანძამდე ნაკადისა და ამგვარად იწვევს კონტროლის გასაცვლელი ტრაფიკის რაოდენობისა და მოცულობის შემცირებას.

OLSR-ში საკონტროლო შეტყობინებათა ორი ძირითადი სახეობაა. ესენია HELLO და TC (ტოპოლოგიის კონტროლი) შეტყობინებები.

1) HELLO შეტყობინებების გაგზავნა პერიოდულად ხდება ყოველი კვანძის მიერ და იგი შეიცავს გამგზავნის იდენტიფიკატორს და სამ სიას: სიას მეზობლებისა, რომლებისგანაც გაგონილ იქნა საკონტროლო ტრაფიკი (პროტოკოლის მიერ განსაზღვრული დროის ინტერვალის განმავლობისას), მაგრამ ორ-მიმართულებიანობა არ იქნა დადასტურებული; სიას მეზობლებისა, რომლებთანაც ორ-მიმართულებიანობა უკვე დადასტურებულია; და შემქმნელი კვანძის MPR ერთობლიობას. მოცემული შეტყობინებების გაცვლა მხოლოდ მეზობელ კვანძებს შორის ხდება, მაგრამ ისინი საშუალებას აძლევენ თითოეულ კვანძს მოიპოვოს ინფორმაცია ერთ- და ორბიჯიანი მეზობლების შესახებ. აღნიშნული ინფორმაცია მოგვიანებით MPR ერთობლიობის შესარჩევად გამოიყენება;
2) TC შეტყობინებების გაგზავნაც პერიოდულად ხდება ქსელის ზოგიერთი კვანძის მიერ. აღნიშნული შეტყობინებები გამოიყენება მთლიან ქსელში ტოპოლოგიის ინფორმაციის გავრცელებისთვის. TC შეტყობინება შეიცავს MPR შემრჩევთა ერთობლიობასა და რიგით ნომერს, ასოცირებულს ამ MPR შემრჩევთა ერთობლიობასთან. ტიპიურად, ქსელის ყველა კვანძი არ აირჩევა MPR-დ, მაგრამ კომუნიკაციისთვის ყველა კვანძს უნდა გააჩნდეს არაცარიელი MPR ერთობლიობა. ამგვარად, გადაწყვეტილება MPR ერთობლიობის ნაცვლად MPR შემრჩევთა ერთობლიობის გაგზავნისა შედეგად გვაძლევს ქსელში გაგზავნილი TC შეტყობინებების რაოდენობის შემცირებას. აღნიშნული TC შეტყობინებები თითოეულ კვანძს უზრუნველყოფენ ქსელის ტოპოლოგიის შესახებ გლობალური ხედვით, რაც მოგვიანებით მარშრუტების გამოთვლისთვის იქნება გამოყენებული.

თითოეული OLSR საკონტროლო შეტყობინება შესაძლებელია ცალსახად იქნას იდენტიფიცირებული კორტეჟის მეშვეობით, რაც შედგება *შემქმნელის იდენტიფიკატორისა* და *შეტყობინების რიგითი ნომრისგან.* კვანძმა ერთი და იგივე შეტყობინება შესაძლოა რამდენჯერმე მიიღოს. შესაბამისად, განმეორებითი გადაცემების თავიდან ასაცილებლად და საკონტროლო ტრაფიკის დამუშავებისთვის თითოეულ კვანძს გააჩნია *დუბლირებული ერთობლიობა,* სადაც პროტოკოლის მიერ განსაზღვრული დროის განმავლობაში ინახება თითოეული მიღებული შეტყობინების უნიკალური იდენტიფიკატორი და ლოგიკური სიდიდე, რომელიც გვიჩვენებს, იყო თუ არა შეტყობინება უკვე გადაგზავნილი. აღნიშნულ მექანიზმს განმეორებითი გადაცემის თავიდან აცილების მექანიზმი ეწოდება.

OLSR საკონტროლო შეტყობინებების გაცვლის გზით თითოეული კვანძი ქსელის შესახებ შემდეგ ინფორმაციას ინახავს: მეზობელ კვანძებამდე არსებული ლინკები ინახება *ლინკების ერთობლიობაში,* ხოლო თავად მეზობელი კვანძები, მათი ბუნების მიხედვით, შენახულია ოთხ ერთობლიობაში. ერთბიჯიანი მეზობლები – *მეზობელთა ერთობლიობაში,* ორბიჯიანი მეზობლები და კვანძები, რომლებიც მათდამი მისაწვდომობას უზრუნველყოფენ - *მეზობელთა ორბიჯიან ერთობლიობაში,* შერჩეული MPR-ები - MPR ერთობლიობაში და კვანძები, რომლებმაც მიმდინარე კვანძი თავიანთ MPR-დ აირჩიეს - *MPR შემრჩევთა ერთობლიობაში.* კვანძები ასევე ინახავენ ინფორმაციას ქსელის ტოპოლოგიის შესახებ, რასაც ისინი TC შეტყობინებებით იღებენ. მისი შენახვა ხდება *ტოპოლოგიის ერთობლიობაში* და აქვს კორტეჟის ფორმა, რომელიც ძირითადად შედგება დანიშნულების კვანძის იდენტიფიკატორისა და ამ დანიშნულებამდე უკანასკნელი ბიჯის იდენტიფიკატორისგან.

მრავალპუნქტიანი რელეს მიზანს გადაცემული პაკეტების ქსელში გავრცელების მინიმიზაცია წარმოადგენს, რაც იმავე რეგიონში დუბლირებული გადაგზავნების შემცირების გზით ხორციელდება. თითოეული კვანძი ირჩევს ერთობლიობას მეზობელი კვანძებისა, რომლებმაც მისი პაკეტები უნდა გადაგზავნონ.

თითოეული კვანძი MPR ერთობლიობას იმგვარად ირჩევს, რომ იგი შეიცავდეს ერთბიჯიანი მეზობლების ისეთ ერთობლიობას, რომელიც ფარავს ყველა ორბიჯიან მეზობელს. გარდა ამისა, ყველა ორბიჯიან მეზობელს ორმიმართულებიანი ლინკი უნდა ჰქონდეს შერჩეულ MPR ერთობლიობასთან. რაც

უფრო მცირეა მრავალპუნქტიანი რელეს ერთობლიობა, მით უფრო ეფექტურია მარშრუტიზაციის პროტოკოლი.

OLSR ამ კვანძების მეშვეობით განსაზღვრავს მარშრუტებს დანიშნულების ყველა პუნქტამდე. აქ MPR კვანძები მარშრუტის შუალედურ კვანძებად არის არჩეული. სქემის განხორციელება ხდება თითოეული კვანძის მიერ კონტროლის ტრაფიკის ინფორმაციის პერიოდული გადაცემით ერთბიჯიანი მეზობლების შესახებ, რომლებმაც იგი მრავალპუნქტიან რელედ აირჩიეს (ან, შესაბამისად, მრავალპუნქტიანი რელეს შემრჩევები). MPR შემრჩევთა შესახებ ინფორმაციის მიღების შემდეგ თითოეული კვანძი ითვლის და ანახლებს მარშრუტებს დანიშნულების ყოველ ცნობილ პუნქტამდე. შესაბამისად, მარშრუტი არის წყაროდან დანიშნულების პუნქტამდე ბიჯების თანმიმდევრობა მრავალპუნქტიანი რელეების გავლით. ნებისმიერი კვანძის მეზობლები, რომლებიც არ შედიან მის MPR ერთობლიობაში, იღებენ და ამუშავებენ კონტროლის ტრაფიკს, მაგრამ მას არ გადაგზავნიან.

OLSR პროტოკოლი შესაძლოა განსაზღვრულ იქნას, როგორც ეს ნაჩვენებია ცხრილში 1.4.

შეიძლება აღვნიშნოთ ის, რომ მარშრუტიზაციის პროაქტიულ პროტოკოლში თითოეულ კვანძს ორი დავალება აქვს შესასრულებელი [47]: (1) კორექტულად მოახდინოს მარშრუტიზაციის პროტოკოლის კონტროლის ტრაფიკის გენერირება (ამ გზით ქსელის სხვა კვანძებს სწორი ინფორმაცია გადაეცემათ) და (2) მართებულად გადასცეს მარშრუტიზაციის პროტოკოლის კონტროლის ტრაფიკი სხვა კვანძების სახელით (ამ გზით კონტროლის ტრაფიკს საშუალება ეძლევა ქსელის ყველა კვანძს მიაღწიოს). საწყისი სპეციფიკაციით OLSR პროტოკოლს გააჩნია ნაგულისხმევი დაშვება, რომ ყველა კვანძი ექვემდებარება მნიშვნელოვანი ტოპოლოგიური ინფორმაციის გაცვლას კონტროლის ტრაფიკის მეშვეობით, რაც დაუცველს ხდის მათ სხვადასხვა თავდასხმებისადმი.

ცხრილი 1.4. (OLSR) ოპტიმიზებული არხის მდგომარეობის მარშრუტიზაციის პროტოკოლის მუშაობა

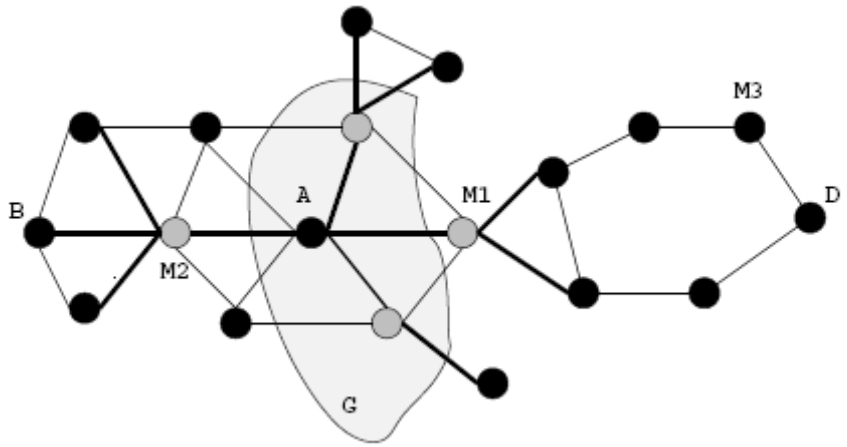
1) თითოეული კვანძი პერიოდულად გადასცემს მის HELLO შეტყობინებებს;

2) მათი მიღება ხდება ყველა ერთბიჯიანი მეზობლის მიერ, მაგრამ არ ხდება გადაცემა;
3) HELLO შეტყობინებები თითოეულ კვანძს უზრუნველყოფს მონაცემებით ერთი და ორბიჯიანი მეზობლების შესახებ;
4) HELLO შეტყობინებების ინფორმაციის გამოყენებით თითოეული კვანძი ახდენს MPR ერთობლიობის შერჩევას;
5) შერჩეული MPR-ების გაცხადება ხდება მომდევნო HELLO შეტყობინებებში;
6) აღნიშნული ინფორმაციის გამოყენებით თითოეულ კვანძს შეუძლია MPR შემრჩევთა ცხრილის შედგენა იმ კვანძების მითითებით, რომლებმაც იგი მრავალპუნქტიან რელედ აირჩიეს;
7) თითოეული კვანძის მიერ პერიოდულად ხდება TC შეტყობინების გაგზავნა და ქსელში გავრცელება, რითაც ხდება MPR შემრჩევთა ერთობლიობის გაცხადება;
8) სხვადასხვა მიღებული TC შეტყობინებების ინფორმაციის გამოყენებით თითოეული კვანძი ადგენს ტოპოლოგიის ცხრილს, რომელიც შედგება ჩანაწერებისგან: შესაძლო დანიშნულების იდენტიფიკატორი (TC შეტყობინებაში მოცემული MPR შემრჩევი), ამ დანიშნულებამდე უკანასკნელი ბიჯის იდენტიფიკატორი (TC შეტყობინების შემქმნელი) და MPR შემრჩევთა ერთობლიობის რიგითი ნომერი.
9) ტოპოლოგიის ცხრილი შემდეგ გამოიყენება მარშრუტიზაციის ცხრილის გამოთვლის ალგორითმის მიერ თითოეული კვანძისთვის მარშრუტიზაციის ცხრილის გამოთვლისათვის.

ამ მიზნით განვიხილოთ ის სიტუაციები, რომლებიც შეიძლება შეიქმნას უსადენო ქსელში სხვადასხვა სახის თავდასხმების დროს. ცხრილში 1.5 ნაჩვენებია OLSR უსაფრთხოებათა ნაკლოვანებები და თავდასხმათა ქმედებების მაგალითები ქსელში, რომელიც ნაჩვენებია სურათზე 4.1.

სურათზე 4.1 მოცემულია ქსელის ტოპოლოგიის მაგალითი ოპტიმიზირებული არხის მდგომარეობის მარშრუტიზაციისთვის. ნაცრისფერი კვანძები A კვანძის მრავალპუნქტიანი რელეებია. ღია ფერის კონტურები

წარმოადგენს კვანძებს შორის კავშირს. მუქი კონტურები გვიჩვენებს ლინკებს A-სა და ყველა ორბიჯიან მეზობელს შორის შერჩეული მრავალპუნქტიანი რელეს ერთობლიობის საშუალებით. *Mi* აღნიშნავს მტრულად განწყობილ კვანძს, *D* არის დანიშნულების კვანძი, ხოლო *G* განსაზღვრავს კვანძთა ჯგუფს.



სურ.4.1 ქსელის ტოპოლოგიის მაგალითი OLSR პროტოკოლისთვის.

უსაფრთოხების ალგორითმების დამუშავებამდე აგრეთვე უნდა განვიხილოთ შესაძლო საფრთხეები და თავდასხმები, რომლებსაც ადგილი შეიძლება ჰქონდეთ უსადენო ქსელში.

დავუშვათ, რომ მონაცემთა ლინკის დონეს საშუალება აქვს მეზობელ კვანძებს შორის სანდო კავშირი უზრუნველყოს ანუ თუ შეტყობინება გაიგზავნა და კოლიზიას ადგილი არ ჰქონია, შეტყობინება განსაზღვრულმა მიმღებმა მიიღო. რეალურად, აღნიშნულ დაშვებას არ მივყავართ სანდო გამჭოლ კომუნიკაციამდე, რადგან ერთი ან რამდენიმე კვანძი შესაძლოა საბაზისო პროტოკოლის მოლოდინის შესაბამისად არ იქცეოდეს.

ცხრილი 1.5 OLSR უსაფრთხოების ნაკლოვანებები, დაფუძნებული სურ. 4.1-ის მაგალითზე

თავდასხმა	მეთოდი	მაგალითი	სამიზნე	შედეგი
იდენტიფიკატო-რის გაყალბება	ყალბი HELLO	M_3 ახდენს HELLO-ს გენერირებას, და თავს აჩვენებს, თითქოს A კვანძი იყოს	ყველა კვანძი	M_3-ის MPR კვანძები საკუთარ თავს წარადგენენ, როგორც უკანასკნელ ბიჯს A კვანძისთვის, რაც შედეგად იწვევს A–სკენ

				მიმავალი მარშრუტების კონფლიქტს
ლინკის გაყალბება	ყალბი HELLO	M_1 ახდენს HELLO-ს გენერირებას, ატყობინებს რა ორმიმართულებიან ლინკებს A–ს ორბიჯიან მეზობელთა უმრავლესობას	კონკრეტული კვანძი	A ირჩევს M_1-ს, როგორც მის მთავარ MPR-ს, რაც საშუალებას აძლევს M_1-ს შეიპყროს და შეცვალოს A–ს ტრაფიკის დიდი ნაწილი
	ყალბი TC	M_1 ახდენს TC-ს გენერირებას და G-ს ატყობინებს, რომ D წარმოადგენს მის MPR-ად შემრჩევს	კვანძთა ჯგუფი	მანძილი M_1-სა და D-ს შორის ჩაითვლება ერთ ბიჯად და, ამდენად, M_1 მთავარ ხიდად იქცევა G-ს და D-ს შორის.
	მარშრუტი-ზაციის ცხრილის გადავსება	M_1 ახდენს მრავალი TC-ს გენერირებას, რომლებიც არარსებულ კვანძებს შეიცავს MPR ერთობლიობაში	ყველა კვანძი	მარშრუტიზაციის ცხრილის ალგორითმი ბევრ დროს დაკარგავს ყალბი მარშრუტის გამოთვლისთვის
ტრაფიკის გადაცემის/გენე-რირების უარყოფა	პაკეტების დაკარგვა	M_1 გადაიქცევა უპირატესობის მქონე კვანძად A–ს ან G-სთვის და ახდენს მათგან მიღებული პაკეტების მოშორებას	კონკრეტული კვანძი კვანძთა ჯგუფი	კავშირუნარიანობის დაკარგვა/კავშირის გაუარესება
	კონტრო-ლის ტრაფიკის გენერირება -ზე უარის თქმა	M_1 MPR-დ არის შერჩეული A–სთვის და ქსელს ამის შესახებ არ ატყობინებს	კონკრეტული კვანძი	A კვანძის მიუწვდომლობა, კავშირის გაუარესება
განმეორებითი თავდასხმები	ტრაფიკის განმეორება	M_1 სხვა კვანძებს “ძველ”, ადრე გაგზავნილ HELLO და TC შეტყობინებებს	ყველა	მოძველებული, ურთიერთსაწინააღმდეგო და/ან მცდარი ინფორმაცია შედის ქსელში, რამაც

		უგზავნის		შესაძლოა ხარვეზებიანი მარშრუტიზაცია გამოიწვიოს
ჭიის ხვრელი	პროტოკო-ლის დაუმორჩილებლობა	M_2 აგვირაბებს ტრაფიკს A-სა და B-ს შორის მარშრუტიზაციის პროტოკოლის მიერ გათვალისწინებული მოდიფიკაციის გარეშე	კონკრეტული კვანძები	გარე, არარსებული ლინკი A-სა და B-ს შორის სრულად კონტროლდება M_2-ის მიერ

როგორც უკვე აღვნიშნეთ ადრე, მარშრუტიზაციის პროაქტიულ პროტოკოლებში თითოეულმა კვანძმა ორი ამოცანა უნდა გადაჭრას. (1) სწორად შეასრულოს მარშრუტიზაციის პროტოკოლის საკონტროლო ტრაფიკის გენერირება (ამ გზით სწორი ინფორმაცია გადასცეს ქსელის სხვა კვანძებს) და (2) სწორად გადასცეს მარშრუტიზაციის პროტოკოლის ტრაფიკი სხვა კვანძების სახელით (ამ გზით საშუალება მისცეს კონტროლის ტრაფიკს მიაღწიოს ყველა კვანძს). ამდენად, თავდასხმა მარშრუტიზაციის პროტოკოლზე შედეგად იგივეს გვაძლევს, რასაც რომელიმე ამ ამოცანის დამახინჯება რომელიმე კვანძის მიერ. აღნიშნულის განხორციელება ოთხი ძირითადი ქმედების მეშვეობით შეიძლება:

1. *მარშრუტიზაციის ყალბი შეტყობინებების ფაბრიკაცია.* კვანძი ახდენს მარშრუტიზაციის კონტროლის ტრაფიკის რეგულარული შეტყობინებების გენერირებას, რომლებიც ყალბ ინფორმაციას შეიცავს ან რომელშიც გამოტოვებულია ინფორმაცია ქსელის მიმდინარე მდგომარეობის შესახებ;
2. *კონტროლის ტრაფიკის გენერირების/გადაცემის უარყოფა.* კვანძი უარს ამბობს მარშრუტიზაციის კონტროლის საკუთარი ტრაფიკის გენერირებაზე ან უარყოფს სხვა კვანძების კონტროლის ტრაფიკის გადამისამართებას (როგორც ეს მოსალოდნელი იყო).
3. *მარშრუტიზაციის კონტროლის ტრაფიკის მოდიფიცირება.* კვანძი გადასცემს სხვა კვანძების ტრაფიკს, მაგრამ ცვლის მას მცდარი ინფორმაციის შეტანით ან ქსელის შესახებ ინფორმაციის გამოტოვებით.

4. *განმეორებითი თავდასხმები.* კვანძი უსმენს მარშრუტიზაციის კონტროლის ტრაფიკის გადაცემას ქსელში და მოგვიანებით ქსელში შეაქვს სავარაუდოდ მცდარი ან მოძველებული ინფორმაცია.

4.3. OLSR-ში უსაფრთხოების უზრუნველყოფის არსებული მეთოდების მიმოხილვა

უკანასკნელ დროს მრავალი ნაშრომი გამოჩნდა, მიძღვნილი OLSR უსაფრთხოების საკითხების ნაწილობრივი გადაწყვეტისადმი [47], [48], [49], [50]. განვიხილოთ მათი მთავარი მახასიათებლები.

შრომების ერთი ჯგუფი წარმოადგენს ტექნიკას [47] OLSR-ზე განხორციელებული თავდასხმების დათვლისა, რაც ეფუძნება გასაღების განაწილების მექანიზმს. მარშრუტიზაციის კონტროლის თითოეული შეტყობინება ხელმოწერილია და აქვს დროითი ნიშნული: ხელმოწერა ახდენს სანდო კვანძებიდან მიღებული შეტყობინებების იდენტიფიცირებას, ხოლო დროითი ნიშნული თავიდან გვაცილებს ძველი შეტყობინებების ხელახლა გაგზავნას. მიდგომა არ ეხება შემდეგ საკითხებს: (ა) სანდო კვანძები შესაძლოა არასწორად იქცეოდნენ ფუნქციონირების დარღვევის გამო და განუზრახველად მოახდინონ მარშრუტიზაციის პროტოკოლის დაზიანება; (ბ) კვანძები უსადენო ქსელში, როგორც წესი, ძალიან ხშირად შედიან და გადიან, რის გამოც რთულია კვანძების დაყოფა სანდო და არასანდო ჯგუფებად; (გ) ხელმოწერის მექანიზმი დეტალიზებული არ არის.

შრომების მეორე ჯგუფი განიხილავს სანდო კვანძების რისკის ქვეშ დაყენებას და კომპრომეტირებას [50]. ავტორები გულისხმობენ, რომ საჯარო გასაღების ინფრასტრუქტურა (PKI) და გამოიყენება დროითი ნიშნულის ალგორითმი. დამატებითი შეტყობინების (ADVSIG) გაგზავნა მარშრუტიზაციის კონტროლის ტრაფიკთან ერთად ხდება. აღნიშნული შეტყობინება შეიცავს დროით ნიშნულსა და ხელმოწერის ინფორმაციას. თითოეულ კვანძს გააჩნია ე. წ. *Certiproof* ცხრილი, სადაც ხდება ADVSIG-ით მიღებული ინფორმაციის შენახვა. შემდეგ ეს ინფორმაცია ხელახლა გამოიყენება მომდევნო შეტყობინებებში არხის მდგომარეობის ინფორმაციის სისწორის დასადასტურებლად. პროცედურა უზრუნველყოფს, რომ ერთადერთმა თავდამსხმელმა კვანძმა ვერ შეძლოს ქსელისთვის არხის მდგომარეობის მცდარი ინფორმაციის გაგზავნა. აღნიშნულის

ნაკლოვანებებია: (ა) მას არ გააჩნია დაცვა სერვისის უარყოფის ან ჭიის ხვრელის თავდასხმისაგან და (ბ) იწვევს ქსელის დამატებით ხარჯებს დამატებითი ტრაფიკისა და ხელმოწერის გამოთვლის კუთხით.

ზემოთხსენებულ სქემაზე დაყრდნობით ავტორთა ჯგუფი გვთავაზობს მექანიზმს [49] გადაცემის თავდასხმების დათვლისა, რაც ეფუძნება კვანძების გეოგრაფიულ მდებარეობას და სქემას, რომელიც მუშაობს რისკის ქვეშ დაყენებულ კვანძებთან, რაც, თავის მხრივ, *ქსელის ნაკადის კონსერვაციას* ეფუძნება, სადაც არასათანადო ქცევის აღმოჩენა ხდება ტრაფიკის გადაცემისას თითოეული კვანძის მიერ მიღებული და გადაცემული პაკეტების რაოდენობის მიხედვით. აღნიშნული წინადადების ნაკლოვანებები შემდეგია: (ა) სისუსტე იმის დაშვებისა, რომ კვანძის მიერ პაკეტების მართებული რაოდენობის გადაგზავნა ადასტურებს იმას, რომ პაკეტები მართებულად იყო გადაგზავნილი; და (ბ) უსაფრთხოების ცენტრალიზებული უფლებამოსილი ორგანოს შემოღება, რაც არასათანადო ქცევის აღმოჩენას და შესაბამის ღონისძიებებს მართავს, უსადენო ქსელში გართულებულია, თუ შეუძლებელი არ არის.

სხვა ავტორთა ჯგუფი შრომას განაგრძობს გასაღების მენეჯმენტის ტექნიკაზე [49] ფოკუსირებით, სადაც გვთავაზობენ მოკლე განხილვას ჭიის ხვრელისა და შეტყობინებათა განმეორების თავდასხმების თავიდან ასაცილებლად. ჭიის ხვრელის თავდასხმის თავიდან ასაცილებელი ტექნიკა ეფუძნება თვლის ტექნიკის [49] სახესხვაობას, სადაც კვანძები გადასცემენ პაკეტების უსარგებლო ინფორმაციას, მიღებულს თითოეული უკანასკნელი k ინტერვალის შემდეგ. ამ გზით შესაძლებელია შემოწმება, მიაღწია თუ არა პაკეტების დაკარგვამ კონკრეტულ ზღვრამდე. ამ შემთხვევაში ყოველი კვანძი რისკის ქვეშ დაყენებულად მიიჩნევა.

არის ნაშრომები, რომლებიც გვთავაზობენ სრულად განაწილებული სერტიფიცირების ორგანოს შემოღებას (DCA), რაც ზღვრული კრიპტოგრაფიის მცნებას [51] ეფუძნება. კვანძი სერტიფიკატს მოითხოვს ქსელის k კვანძების (მონაწილეების) ნებისმიერი გაერთიანებისგან. ყოველი მონაწილე განსაზღვრავს, სურს თუ არა მოთხოვნის მომსახურება, რაც ეფუძნება იმას, ჩაითვლება თუ არა მომთხოვნი კვანძი სათანადო ქცევისად. k რაოდენობის "ნაწილობრივი სერტიფიკატების" მიღების შემდეგ ხდება მათი გაერთიანება უფლებამოსილი

სერტიფიკატის გენერირებისთვის. ამ მეთოდში არ არის ჩართული მონიტორინგის სისტემა ქსელის კვანძების ქცევის განსაზღვრისათვის.

ზემოთ მოყვანილ მასალაში განხილულ კრიპტოგრაფიულ სქემებს გარდა, უსაფრთხო მარშრუტიზაციის არსებული წინადადებები მოიცავს თანამშრომლობის იძულების მექანიზმებს, რომლებიც შესაძლოა ორ კატეგორიად დაიყოს: ვალუტაზე დაფუძნებული მექანიზმები და რეპუტაციაზე დაფუძნებული მექანიზმები. ვალუტაზე დაფუძნებული მექანიზმები ემყარება ან კვანძებს შორის ვირტუალური ვალუტის გაცვლას [52] ან სერვისის არსებობას, რომელიც კრედიტებს ანაწილებს იმ ქვითრებზე დაფუძნებით, რომელთა მიღება ქსელში გადაცემადი შეტყობინებებიდან ხდება [53]. რეპუტაციაზე დაფუძნებული გადაწყვეტილებები, ტიპიურად, შედგება სამი ცალკეული მექანიზმისგან: (1) ადგილობრივი მონიტორინგის მექანიზმი ქსელის კვანძების ქცევაზე დაკვირვებისა და მათი სანდოობის განსაზღვრისათვის; (2) რეპუტაციის გავრცელების მექანიზმი სხვა კვანძებისათვის ინფორმაციის მისაწოდებლად წინამორბედი მექანიზმებით შესრულებული დაკვირვებების შედეგების შესახებ; და (3) დასჯის/იზოლაციის მექანიზმი არასათანადო ქცევისგან ქსელის დასაცავად.

შეიძლება იყოს შემოღებული ე.წ. ვირტუალური ვალუტა, რომელიც გამოიყენება პაკეტების გადაგზავნის სერვისის საფასურის გადასახდელად [52]. *პაკეტის ჩანთურ მოდელში* საწყისი კვანძი ვალუტას პაკეტში გაგზავნამდე ტვირთავს და თითოეული გადამგზავნი კვანძი მოიპოვებს ამ თანხის ნაწილს, როგორც ანგარიშსწორებას. *პაკეტების გაცვლის მოდელში* თითოეული გადამგზავნი კვანძი ყიდულობს პაკეტს წინამორბედი კვანძისგან ვალუტის რაღაც რაოდენობის სანაცვლოდ და მიჰყიდის მათ მომდევნო კვანძებს უფრო ძვირად. ორივე მიდგომა ეყრდნობა გაყალბებამედეგი უსაფრთხოების მოდულს. უნდა ითქვას, რომ რთულია შეფასება პაკეტით გადასაგზავნი ვალუტის რაოდენობისა, რათა მან დანიშნულების პუნქტს მიაღწიოს პაკეტის ჩანთურ მოდელში, ხოლო პაკეტის გაცვლის მოდელი იძლევა ქსელის გადატვირთვის შესაძლებლობას, რადგან წყარო არ არის ვალდებული გადაიხადოს პაკეტების გაგზავნისთვის. ერთ-ერთ სტატიაში [54] ავტორები გვერდს უვლიან შეფასებას ვირტუალური ვალუტის რაოდენობის საკითხს, რომელიც უნდა გაიზგავნოს თვლის ტექნიკის გამოყენებით, სადაც თითოეულ კვანძს გააჩნია თანხის მრიცხველი, რომელიც მცირდება, როდესაც

კვანძი საკუთარ პაკეტს გზავნის და იზრდება, როდესაც იგი პაკეტს გადაგზავნის სხვა კვანძის სახელით.

"მეთვალყურე" და "მარშრუტის შემფასებელი" [55] წყაროს დინამიკური მარშრუტიზაციის (DSR) პროტოკოლის ორი გაფართოებაა, რომელიც ცდილობს აღმოაჩინოს და შეამციროს მარშრუტიზაციის არასათანადო ქცევის ეფექტი. *"მეთვალყურე"* არის არასათანადო ქცევის აღმოჩენის მექანიზმი, დაფუძნებული მარშრუტში კვანძის მონიტორინგზე, რათა გაირკვეს, გადაგზავნის თუ არა იგი მისთვის გაგზავნილ პაკეტებს. თუკი კვანძი, რომელიც ვალდებულია გადაგზავნოს პაკეტი, ამას დროის გარკვეული მონაკვეთის განმავლობაში ვერ ახერხებს, *"მეთვალყურე"* თან ურთავს წარუმატებლობის რეიტინგს კონკრეტული კვანძისათვის და როდესაც წარუმატებლობა კონკრეტულ ზღვარს გადააჭარბებს, ეს კვანძი მიიჩნევა არასათანადო ქცევის მქონედ. შემდეგ *"მარშრუტის შემფასებელი"* მოგროვილ ინფორმაციას იყენებს საუკეთესო შესაძლო მარშრუტების განსაზღვრისათვის, რათა აცილებულ იქნან არასათანადო ქცევის კვანძები. აღნიშნული მექანიზმი არ სჯის მოცემულ კვანძებს, რეალურად იგი ათავისუფლებს მათ გადაგზავნის ოპერაციებისგან.

CONFIDANT ნიშნავს კვანძების თანამშრომლობას და სამართლიანობას დინამიურ უსადენო ქსელებში Cooperation Of Nodes, Fairness in Dynamic Ad-hoc NeTworks [56]. ეს გახლავთ DSR-ს გაფართოვება, რაც ოთხი ცალკეული მექანიზმისგან შედგება. მონიტორინგის *მექანიზმი* გადახრებს აღმოაჩენს მარშრუტის მომდევნო კვანძის მიერ გადაცემაზე დაკვირვებით, რათა დადგენილ იქნას გადაცემის უარყოფის თავდასხმები. *ნდობის მენეჯერი* პასუხისმგებელია განგაშის სიგნალების მიღება/გადაცემაზე და მიღებული საგანგაშო სიგნალებისათვის მინიჭებული ნდობის მართვაზე საწყისი კვანძის ნდობის ხარისხის შესაბამისად. *რეპუტაციის სისტემა* მართავს ქსელის კვანძების რეიტინგებს; მათი მოდიფიცირება ხდება ხარისხის ფუნქციის შესაბამისად, რომელიც სხვადასხვა წონებს ანიჭებს სხვადასხვა არასათანადო ქცევას. *მარშრუტის მენეჯერი* მონაწილეობას იღებს მარშრუტის შერჩევის მექანიზმში, რისთვისაც აუქმებს მარშრუტებს, რომლებიც შეიცავენ დაუშვებელი რეიტინგის მქონე კვანძებს და ატარებს ღონისძიებებს არასათანადო ქცევის კვანძების იზოლირებისათვის.

CORE (Collaborative Reputation Mechanism) არის თანამშრომლობის რეპუტაციის მექანიზმი [57], რომელიც უსადენო ქსელებში აიძულებს კვანძებს

ითანამშრომლონ. იგი შედგება შემოწმების მექანიზმისა და რეპუტაციის რთული მექანიზმისგან, რაც გულისხმობს რეპუტაციის სამ სახეობას, რომლებიც კომბინირებულია რეპუტაციის გლობალურ სიდიდედ. შემოწმების მექანიზმი ახდენს მეზობელი კვანძების ზოგიერთი ოპერაციის შესრულების მონიტორინგს. *სუბიექტური რეპუტაცია* ეფუძნება განხორციელებულ დაკვირვებას და თავიდან იცილებს ცალკეულ არასათანადო ქცევას, რისთვისაც გამოთვლებში წარსულ დაკვირვებებს მნიშვნელობას ანიჭებს. *არაპირდაპირი რეპუტაცია* დაფუძნებულია მხოლოდ პოზიტიური ინფორმაციის გაცვლაზე, რომლის მოწოდება ქსელის სხვა კვანძების მიერ ხდება. *ფუნქციონალური რეპუტაცია* ეფუძნება სხვადასხვა შესრულებად ფუნქციებზე დაკვირვებას (მაგალითად, მარშრუტებისა და პაკეტების გადაგზავნა), რის გაერთიანებაც გვაძლევს რეპუტაციის გლობალურ მნიშვნელობას. მოცემული მნიშვნელობა განსაზღვრავს თითოეული კვანძის სურვილს შეასრულოს ქსელის ოპერაცია მათი სახელით.

ჯამურად, OLSR-ს უსაფრთხოებათა არსებული გაფართოებები მოიცავს ცალკეულ პრობლემათა მნიშვნელოვან ნაწილს. ერთი შეხედვით კონსენსუსი მიღწეულია ხელმოწერისა და გასაღების მენეჯმენტის სისტემების გამოყენებასთან დაკავშირებით, რათა უზრუნველყოფილ იქნას გამგზავნის მარშრუტიზაციის კონტროლის ტრაფიკის აუთენტიფიკაცია. მსგავსად ამისა, დროითი ნიშნულების მეთოდიკამ აღიარება ჰპოვა ძველი შეტყობინებების განმეორების წინააღმდეგ ბრძოლაში. გარდა კრიპტოგრაფიული უსაფრთხოების გადაწყვეტილებებისა, რომლებიც აუცილებელია მთლიანობისა და აუთენტიფიკაციის გარანტირებისათვის, არსებითია მექანიზმების არსებობა მომხმარებლების თანამშრომლობის იძულებისთვის, რისთვისაც ხდება სათანამშრომლო ინიციატივის დაჯილდოვების და/ან თანამშრომლობაზე უარის თქმის შემთხვევაში დასჯის ღონისძიებების გატარება. დღემდე შემუშავებული გადაწყვეტილებები ორი სახისაა: ვალუტაზე დაფუძნებული, რომლებიც დამოკიდებულია გაყალბებამედეგ კომპონენტებზე, რომლებმაც, თავის მხრივ, შესაძლოა შეამცირონ მათი ფართო გამოყენება; და რეპუტაციაზე დაფუძნებული გადაწყვეტილებები, რომლებიც ეყრდნობა ქსელში კვანძების იდენტიფიცირების უნარს.

4.4. უსადენო ქსელებში რეპუტაციის საფუძველზე უსაფრთხოების უზრუნველყოფის თეორიული ასპექტები

4.4.1. რეპუტაციის კონცეფცია

უსადენო ქსელებში კვანძები შესაძლოა წარმოვიდგინოთ ერთობის წევრებად (ან სუბიექტებად), რომლებიც საერთო რესურსს ინაწილებენ. კვანძის არასათანადო ქცევასთან დაკავშირებული პრობლემების გასაღები დევს საერთო რესურსის გამოყენებისა და ერთობის წევრთა კოოპერატიულ ქცევას შორის მჭიდრო კავშირში. ამდენად, ერთობის ყველა წევრი, რომელიც რესურსს ინაწილებს, ვალდებულია წვლილი შეიტანოს ერთობის ცხოვრებაში, რათა აღნიშნული რესურსების გამოყენების უფლება მიენიჭოს. მიუხედავად ამისა, ერთობის წევრები ხშირად ერთმანეთთან დაკავშირებული არ არიან და ერთმანეთის ქცევის შესახებ არანაირი ინფორმაცია არ გააჩნიათ.

ჩვენ ვიზიარებთ იმ მოსაზრებას, რომ რეპუტაცია სათანადო საზომია ქსელის საყოველთაო ოპერაციებში ვინმეს წვლილისა. მართლაც, რეპუტაცია, ჩვეულებრივ, განისაზღვრება, როგორც ოდენობა რწმენისა, რაც ერთობის ცალკეული წევრის მიერ არის ჩანერგილი კონკრეტულ გარემოსა ან ინტერესის დომენში. კარგი რეპუტაციის მქონე წევრებს, გამომდინარე მათი წვლილიდან ერთობის ცხოვრებაში, შეუძლიათ რესურსების გამოყენება, იმ დროს, როდესაც ცუდი რეპუტაციის წევრებს თანდათანობით გარიცხავენ ერთობიდან, რადგან მათ უარი თქვეს თანამშრომლობაზე.

შემდგომ მოყვანილია მიდგომა, რომელიც შეიძლება იყოს გამოყენებული საფუძვლად უსაფრთხოების მექანიზმისა, რომელიც გადაჭრის არასათანადო ქცევის კვანძებთან დაკავშირებულ პრობლემებს უსადენო ქსელებში, კერძოდ კი იმ შემთხვევაში, როდესაც გამოიყენება მარშრუტიზაციის პროტოკოლი OLSR.

ამ მიზნით ჩვენ ვთავაზობთ რეპუტაციის მექანიზმის გამოყენებას. გარდა ამისა, წარმოდგენილია გადაწყვეტილებები, რათა მინიმიზებულ იქნას კვანძების არასათანადო ქცევის შეცდომით დადგენასთან დაკავშირებული პრობლემები. მაგალითად, არასახარბიელო მდგომარეობაში არსებული კვანძი, ქსელიდან იმავე საფუძვლით არ უნდა იქნას გარიცხული, როგორც არასათანადო ქცევის კვანძი: აღნიშნული უნდა გაკეთდეს რეპუტაციის სიდიდის აკურატული შეფასებით, რაც ცალკეულ არასათანადო ქცევასაც ითვალისწინებს.

განვიხილოთ რეპუტაციის ცნება. ჩვეულებრივ გამოყოფენ სამი სახის რეპუტაციას: სუბიექტურ, არაპირდაპირ და ფუნქციონალურ რეპუტაციას.

სუბიექტური რეპუტაციის ცნება შეიძლება იყოს გამოყენებული რეპუტაციის დასახასიათებლად, რომლის გამოთვლა უშუალოდ სუბიექტის დაკვირვების საფუძველზე ხდება.

რაც შეეხება არაპირდაპირ რეპუტაციას - მისი შემოღებით რეპუტაციის დადგენის დროს ემატება რთულ ერთობათა მახასიათებლების ასახვის შესაძლებლობა: სუბიექტის რეპუტაციისთვის მიკუთვნებულ საბოლოო სიდიდეზე ასევე გავლენას ახდენს ერთობის სხვა წევრების მიერ მოწოდებული ინფორმაცია.

ცნებას - ფუნქციონალური რეპუტაცია - გამოიყენებენ სუბიექტური და არაპირდაპირი რეპუტაციის დასადგენად სხვადასხვა ფუნქციების მიმართ. ამ უკანასკნელი სახის რეპუტაციის შემოღებით მოდელს ემატება სუბიექტის გლობალური რეპუტაციის გამოთვლის შესაძლებლობა, რაც სხვადასხვა დაკვირვება-შეფასებების კრიტერიუმებს ითვალისწინებს.

ქვემოთ მოყვანილია აღნიშნული რეპუტაციების დახასიათება, რომელიც წარმოდგენს ერთ-ერთ არსებულ მოდელს.

არსებობს მიდგომა, რომლის თანახმად სუბიექტური რეპუტაცია t დროს s_i სუბიექტის აზრით გამოითვლება s_j-სუბიექტის დაკვირვების რეიტინგული ფაქტორების საშუალო სიდიდის გამოყენებით, რომელიც მეტ მნიშვნელობას ანიჭებს წარსულ დაკვირვებებს. მიზეზი იმისა, თუ რატომ ენიჭება წარსულ დაკვირვებებს მეტი მნიშვნელობა, იმაში მდგომარეობს, რომ ცალკეულ არასათანადო ქცევას უახლოეს დაკვირვებებში მინიმალური გავლენა უნდა ჰქონდეს რეპუტაციის საბოლოო სიდიდის შეფასებაზე. შედეგად, აღნიშნული მოდელის ავტორები თვლიან, რომ შესაძლებელია მცდარი რეპუტაციის დადგენის თავიდან აცილება ლინკის წყვეტის გამო, და არასახარბიელო მდგომარეობაში მყოფი კვანძებით გამოწვეული არასათანადო ქცევის ლოკალიზებით.

ამ შემთხვევაში ძირითადი ფორმულა, რომლითაც გამოითვლება სუბიექტური რეპუტაცია არის:

$$r_{s_i}^t(s_j \mid f) = \sum p(t,t_k) * \sigma_k ,$$

სადაც $r_{s_i}^t(s_j \mid f)$ - არის სუბიექტური რეპუტაციის სიდიდე, გამოთვლილი t დროს s_i სუბიექტის მიერ s_j სუბიექტისათვის f ფუნქციასთან მიმართებაში;

$p(t,t_k)$ - არის დროზე დამოკიდებული ფუნქცია, რომელიც უმაღლეს მნიშვნელობას ანიჭებს σ_k წარსულ სიდიდეებს;

σ_k - წარმოადგენს რეიტინგის ფაქტორს, რომელიც k დაკვირვებას მიენიჭა: ამ შემთხვევაში გამოიყენება სკალა, რომელიც იწყება -1-ით უარყოფითი გამოსახულებისათვის (რაც იმას ნიშნავს, რომ დაკვირვების შედეგი არ შეესაბამება მოსალოდნელს) და გრძელდება +1-მდე დადებითი გამოსახულებისათვის (როდესაც დაკვირვების და მოსალოდნელი შედეგი თანხვდება). როდესაც დაკვირვებათა რაოდენობა ან ხარისხი, შეკრებილი t დროის შემდეგ, არასაკმარისია, სუბიექტური რეპუტაციის საბოლოო მნიშვნელობა იძენს ნულოვან სიდიდეს, რაც ნეიტრალური გამოსახულებისათვის გამოიყენება. საბოლოოდ, თუ მოცემულია, რომ $\sigma_k \in [-1,1]$ და $p(t,t_k)$ ნორმალიზებული სიდიდეა, აგრეთვე $r_{s_i}^t(s_j \mid f) \in [-1,1]$

ასევე გათვალისწინებული უნდა იყოს ის გარემოება, რომ ერთობლიობა $\{s_j\}$ შეზღუდულია s_i სუბიექტის მეზობელთა ერთობლიობით. ცნება მეზობელი აქ გამოიყენება უსადენო გადაცემის ფარგლებში მყოფი სუბიექტის სხვა სუბიექტის აღსანიშნავად. მოყვანილ სქემაში სუბიექტური რეპუტაციის შეფასება ხდება მხოლოდ სუბიექტსა და მის მეზობელს შორის უშუალო ურთიერთქმედების გათვალისწინებით.

მოყვანილ მოდელში $ir_{s_i}^t(s_j \mid f)$ აღნიშნავს s_j-ის არაპირდაპირ რეპუტაციას, შეკრებილს s_i-ს მიერ t დროს f ფუნქციისთვის. განხილულ მიდგომაში არაპირდაპირი რეპუტაციის მეშვეობით მოპოვებულ ინფორმაციას შესაძლოა მხოლოდ დადებითი მნიშვნელობა გააჩნდეს: ამდენად პრევენცირებულია სერვისის იერიშებზე უარის თქმა, რაც ეფუძნება ლეგიტიმური კვანძებისთვის უარყოფითი რეიტინგის ყალბ გადაცემას.

რაც შეეხება ფუნქციონალურ რეპუტაციას განხილულ მოდელში შესაძლებელია შემდეგი მაგალითის მოყვანა: სუბიექტს s_i შეუძლია s_j სუბიექტის სუბიექტური რეპუტაციის გამოთვლა $r_{s_i}^t(s_j \mid$პაკეტების გადაცემა) პაკეტების გადაცემის ფუნქციის მიმართ და სუბიექტური რეპუტაციის ფუნქციის გამოთვლა $r_{s_i}^t(s_j \mid$*მარშრუტიზაცია)* მარშრუტიზაციის ფუნქციის მიმართ და მათი კომბინირება

სხვადასხვა წონების გამოყენებით, რათა მიღებულ იქნას s_j სუბიექტის გლობალური რეპუტაციის სიდიდე.

აღწერილ მოდელში რეპუტაციის ინფორმაციის კომბინირება შემდეგი ფორმულის გამოყენებით ხდება:

$$r_{s_i}^t(s_j) = \sum_k w_k \{r_{s_i}^t(s_j | f_k) + ir_{s_i}^t(s_j | f_k)\}$$

სადაც w_k წარმოადგენს წონას, ასოცირებულს ფუნქციური რეპუტაციის სიდიდესთან.

ფაქტიურად, აქ $r_{s_i}^t(s_j)$ წარმოადგენს გლობალური რეპუტაციის სიდიდეს, რომლის შეფასება ყველა კვანძზე ხდება: იგი ჯამური რეპუტაციის განსაზღვრებაა. გლობალური რეპუტაციის შეფასებისთვის გამოყენებული წონის w_k არჩევა აკურატულად უნდა მოხდეს, რადგან მან შესაძლოა გავლენა იქონიოს სისტემის მთლიან სიცოცხლისუნარიანობაზე. იმ გამოცდილებიდან გამომდინარე, რომ იმ შემთხვევაშიც კი, თუ პაკეტების გადაცემის ფუნქციის, ისევე, როგორც მარშრუტიზაციის ფუნქციის შესრულების იძულება სავალდებულოა, პირველს უფრო დიდი გავლენა აქვს გლობალურ შესრულებაზე, ვიდრე მეორეს. სწორედ ამიტომ ახდენს w_k-ს სათანადოდ არჩევა ხაზგასმას პაკეტის გადაცემის ფუნქციის სისწორისა, როდესაც კვანძის საყოველთაო რეპუტაციის შეფასება ხდება.

ყოველივე აღნშნულის რეალიზაციისათვის გამოიყენება რეპუტაციის ცხრილები, სადარაჯო მექანიზმი და შემუშავებულია რთული პროტოკოლები.

რეიტინგების ცხრილი (RT) განისაზღვრება, როგორც მონაცემთა სტრუქტურა, დაცული ქსელის ყოველ ერთეულში. ცხრილის თითოეული სტრიქონი შეიცავს კვანძის კუთვნილი რეპუტაციის მონაცემს.

მოცემულ მოდელში თითოეული სტრიქონი ოთხი ჩანაწერისგან შედგება: ერთეულის უნიკალური იდენტიფიკატორი, უკანასკნელი სუბიექტური დაკვირვებების ერთობლიობა, განხორციელებული ამ კვანძის ქცევაზე, სია უკანასკნელი არაპირდაპირი რეპუტაციის სიდიდეებისა, რომლებიც სხვა ერთეულიების მიერ არის მოწოდებული და სიდიდე რეპუტაციისა, რაც გამოთვლილია წინასწარ განსაზღვრული ფუნქციისთვის.

4.4.2. შემუშავებული უსაფრთხოების უზრუნველყოფის რეპუტაციის კონცეფცია

ჩვენ ვთვლით, რომ ზევით მოყვანილ მოდელს ახასიათებს გარკვეული სუსტი მხარეები და ნაკლოვანებები. კერძოდ, სუბიექტური რეპუტაციის დადგენის დროს გამოიყენება დროზე დამოკიდებული ფუნქცია, რითაც უპირატესობა ენიჭება წარსულ დაკვირვებებს. ჩვენი აზრით, რა თქმა უნდა, კვანძის მდგომარეობა დროთა განმავლობაში შეიძლება იცვლებოდეს და ხდებოდეს არასახარბიელო მისი რეპუტაციის დასადგენად, მაგრამ რეპუტაციის ასეთი დამოკიდებულება გაზომვის დროზე მიგვაჩნია ნაკლოვანებად, ვინაიდან კვანძს უნდა გააჩნდეს შესაძლებლობა ნებისმიერ დროს დაადგინოს სხვა კვანძის რეპუტაცია.

შემდეგ შეიძლება აღინიშნოს რომ, როგორც ჩანს მოყვანილი მოდელიდან, ამა თუ იმ კვანძის რეპუტაციის დასადგენად საჭირო ხდება მოსალოდნელი და დაკვირვების შედეგების შედარება.

გარდა ამისა, სუბიექტური რეპუტაციის დადგენის დროს აღწერილ მოდელში ნაგულისხმევია, რომ ნებისმიერი კვანძი ადგენს აღნიშნულ რეპუტაციას ყველა მეზობლისათვის, რაც საკმაოდ მნიშვნელოვან რესურსებს მოითხოვს.

რაც შეეხება არაპირდაპირ რეპუტაციას, აქ აუცილებელია ერთობის წევრებს შორის ინფორმაციის გაცვლა კვანძების რეპუტაციის შესახებ. ჩვენ ვთვლით, რომ აღნიშნული ოპერაცია მოიცავს გარკვეულ საშიშროებას, ვინაიდან ქსელში შეიძლება გადაიცეს მცდარი ინფორმაცია, და შესაძლებელი გახდეს სწორად მომუშავე კვანძების დადანაშაულება არასწორ ქმედებაში, და პირიქით.

ამასთან ერთად უნდა გავითვალისწინოთ ის გარემოება, რომ ჩვენ ვამუშავებთ უსაფრთხოების უზრუნველყოფის მოდელს OLSR-მარშრუტიზაციის პროტოკოლისთვის, რომელიც ხასიათდება გარკვეული შეზღუდვებით ინფორმაციის გადაცემაზე, კერძოდ, აღნიშნულ პროტოკოლში ტრაფიკის კონტროლის ინფორმაცია გადაიცემა მხოლოდ MPR-ად შერჩეულ კვანძებზე, და არა ყველა კვანძზე (MPR-კვანძებად ირჩევა ის კვანძები, რომელთა საშუალებით შესაძლებელია მოცემული კვანძიდან ყველა დანარჩენი კვანძის მიღწევა).

გარდა ამისა, OLSR-პროტოკოლს ახასიათებს სამსახურებრივი ინფორმაციის გადაცემა, რომლის საფუძველზე შემდგომ წარმოებს ქსელში მარშრუტის გაკვალვა, ანუ პროტოკოლის და კვანძების ძირითადი ამოცანის – მარშრუტიზაციის - შესრულება. ამიტომაც, მარშრუტიზაცია სრულად

დამოკიდებულია კვანძების სწორად მუშაობაზე. აქაც შეიძლება ითქვას, რომ ქსელის კვანძებს მიერ უნდა იყოს შესრულებული ორი მოთხოვნა: კვანძების მიერ ინფორმაციის გადაცემა სხვა კვანძებზე და მარშრუტიზაციის პროცედურის უზრუნველყოფა.

გარდა ზემოთ აღნიშნული ნაკლოვანებების გასწორებისა და თავისებურებების გათვალისწინებისა, ჩვენ მიზნად ვისახავთ ზოგადი რეპუტაციის გამოსახულების მიღებას, რომელიც გააერთიანებს თავის თავში ზემოთ აღნიშნული სამი ტიპის რეპუტაციას: სუბიექტურს, არაპირდაპირს და ფუნქციონალურს.

ყოველივე ზემოთქმულის გათვალისწინებით ჩვენს მიერ შემუშავებულია გამოსახულება, რომელიც ჩვენის აზრით, უნდა ასახავდეს j-სუბიექტის რეპუტაციას, დადგენილს i-სუბიექტის მიერ:

$$r^{i}{}_{j} = f(R1_{j}, R2_{j}),$$

სადაც R1j – არის პირველადი რეიტინგი,

R2j – მეორადი რეიტინგი.

ჩავთვალოთ, რომ მეორადი რეიტინგი ასახავს კვანძის სურვილს გადასცეს მასთან მოსული ინფორმაცია (გადაცემის ფუნქცია), მის დასადგენად შეიძლება იყოს გამოყენებული უშუალო დაკვირვება, ხოლო პირველადი რეიტინგის საშუალებით შესაძლებელია შეფასდეს კვანძების მიერ მარშრუტიზაციის ამოცანის შესრულება (მარშრუტიზაციის ფუნქცია), პირველადი რეიტინგი შეიძლება შეფასდეს ქსელში გადაცემული ტრაფიკის კონტროლის ინფორმაციის შემოწმებითა და ამ მონაცემების კორელაციით მეორადი რეიტინგის მონაცემებთან.

აღნიშნული რეიტინგები შეიძლება იღებდნენ მნიშვნელობებს {0, 100} ფარგლებში. შესაბამისად, რეპუტაცია შეიძლება იყოს გამოსახული პროცენტული ოდენობით. მის მნიშვნელობაზე დამოკიდებული იქნება, თუ რა ალბათობით გადაიცემა შესაბამისი კვანძისგან მიღებული ინფორმაცია.

რეპუტაციის (ანუ რეიტინგების) ცხრილის თითოეული ჩანაწერი მოიცავს მხოლოდ სამ ელემენტს – კვანძის იდენტიფიკატორს, მისი პიველადი და მეორადი რეიტინგების მნიშვნელობებს.

შემუშავებულია მთელი რიგი ალგორითმებისა, რომელთა საშუალებითაც დგინდება პირველადი და მეორადი რეიტინგების მნიშვნელობა. გათვალისწინებულია შეცდომის აცილების აუცილებლობა კვანძის არასახარბიელო მდგომარეობის

შემთხვევაში. ამ მიზნით სრულდება HELLO და TC შეტყობინებების (აღნიშნული შეტყობინებები დამახასიათებელია OLSR-პროტოკოლისათვის) საშუალებით მოძიებული ინფორმაციის შედარება უკუკავშირით მიღებულ ინფორმაციასთან. საბოლოოდ კი, აცილებულია დროზე დამოკიდებულია რეპუტაციის შეფასების საჭიროება.

დასკვნის სახით, პირველ რიგში უნდა ითქვას, რომ რეპუტაციის დადგენის შემოთავაზებულ მოდელში თითოეული კვანძი ადგენს რეიტინგებს მხოლოდ მის მიერ არჩეული MPR-ებისათვის, და არა ყველა კვანძისათვის, რაც მოითხოვს ნაკლები რესურსების გამოყენებას.

ინფორმაციის გადაცემის სისწორე, ანუ რეპუტაცია დგინდება დროზე ყველანაირი დამოკიდებულების გარეშე. შემოთავაზებულ მოდელში არ არის საჭირო ინფორმაციის გავრცელება ქსელში (არც რეპუტაციის შესახებ და არც რეიტინგების ცხრილისა). აღნიშნული მიდგომა თავიდან გვაცილებს ქსელში მცდარი ინფორმაციის გავრცელებას კვანძების მუშაობის სისწორის შესახებ.

და ბოლოს შეიძლება აღინიშნოს, რომ რეპუტაციის შეფასების მოყვანილი გამოსახულება, ფაქტობრივად, მოიცავს თავის თავში ყველა სახის რეპუტაციებს - სუბიექტურ და არაპირდაპირ რეპუტაციებს სხვადასხვა ფუნქციებისათვის, ვინაიდან აფასებს ძირითადი ამოცანების შესრულებას დანარჩენი კვანძების მუშაობს შესახებ ინფორმაციის საფუძველზე.

4.5 OLSR პროტოკოლის გაფართოვება არასაიმედო და არასათანადო ქცევის კვანძის დასადგენად

4.5.1. OLSR-ის გაფართოვება უსაფრთხოების უზრუნველსაყოფად

OLSR-სთვის უკვე შემუშავებულია გარკვეული მიდგომა მისი უსაფრთხოების უზრუნველყოფის მიზნით. ქვემოთ მოგვყავს ამ მეთოდის განხილვა, ვინაიდან ჩვენი ნაშრომის მიზანს წარმოადგენს აღნიშნული მეთოდის განვითარება და გაუმჯობესება.

OLSR-სთვის ერთ-ერთი არსებული უსაფრთხოების სქემის ფუნდამენტურ მოსაზრებას წარმოადგენს უზრუნველყოფა იმისა, რომ კვანძებმა ზუსტად მოახდინონ OLSR ტრაფიკის გენერირება და გადაცემა. აღნიშნული მიზნის მისაღწევად გზამკვლევი პრინციპი იქნება სათანადო ქცევის კვანძების დაჯილდოვება და დამაზიანებელთა დასჯა. ავღნიშნოთ, რომ სათანადო ქცევის

კვანძი ისეთი კვანძია, რომელიც: (1) სწორად ახდენს მარშრუტიზაციის პროტოკოლის კონტროლის ტრაფიკის გენერირებას და (2) სწორად გადასცემს მარშრუტიზაციის პროტოკოლის ტრაფიკს სხვა კვანძების სახელით. ამდენად, ჩვენს მიზანს იმ კვანძების დავჯილდოვება წარმოადგენს, რომლებიც სათანადო ქცევის აღნიშნულ გასანზღვრებას შეესაბამებიან. რსებობს მეთოდი, სადაც ამ მიზნით, OLSR-ს ჩვეულ ოპერაციებს სამი ელემენტი ემატება:

- *სრული გზის შეტყობინება* (CPM): CPM გამოიყენება გზის გასაკვალად, რომელიც განვლილია სხვა შეტყობინების მიერ ქსელში. TC შეტყობინების მიღების შემდეგ, ქვემოთ განსაზღვრული წესების შესაბამისად, თითოეული კვანძი CPM-ს უკან უგზავნის შემქმნელ კვანძს მარშრუტით, რომელიც გაკვალულია ორიგინალური TC შეტყობინების მიერ, რომელმაც, შესაბამისად, უნდა შეინახოს მის მიერ გაკვალული მარშრუტი (მაგალითად, ჩანაწერის მარშრუტის ალმის დაფიქსირებით IP სათაურში ან ინფორმაციის შენახვით TC შეტყობინების ძირითად ნაწილში);
- *რეიტინგის ცხრილი:* ქსელის თითოეული კვანძი ინახავს რეიტინგის ცხრილს, სადაც ჩაწერილია ინფორმაცია მისი ერთ- და ორბიჯიანი მეზობლების შესახებ. რეიტინგის ცხრილის თითოეულ ჩანაწერს გააჩნია კვანძის ID, და პირველადი და მეორადი რეიტინგი. კვანძის ID ახდენს კვანძის უნიკალურ იდენტიფიცირებას, მეორადი რეიტინგი არის კვანძის კლასიფიკაცია უშუალო დაკვირვების მიხედვით, ხოლო პირველადი რეიტინგი არის კვანძის უფრო ჩამოყალიბებული კლასიფიკაცია, რაც ეფუძნება მეორად რეიტინგსა და CPM-ების მიერ მოწოდებული ინფორმაციის შესაბამისობას კვანძის მიერ გაცხადებულ ქსელურ ინფორმაციასთან. აღნიშნულ ცხრილში დაცული ინფორმაცია საშუალებას აძლევს კვანძს გადაწყვიტოს, როგორ მოექცეს არასათანადო ქცევის კვანძებს;
- *გამფრთხილებელი შეტყობინება:* კიდევ ერთი სახეობა შეტყობინებებისა, რომელსაც პოტენციური არასათანადო ქცევის გამაფრთხილებელი შეტყობინებები ეწოდება, გამოიყენება მეზობელი კვანძების შესატყობინებლად კვანძების სავარაუდო არასათანადო ქცევის შესახებ.

მოყვანილი უსაფრთხოების ალგორითმი მოითხოვს თითოეული კვანძისა და ყოველი პაკეტის ზუსტი წარმომავლობის იდენტიფიცირების უნარს, ამიტომაც იგი ემყარება განაწილებული CA-ს გამოყენებას, რაც უსადენო ქსელების არსს შეესაბამება.

4.5.2. OLSR პროტოკოლის სპეციფიკაცია უსაფრთხოების გაფართოვების გათვალისწინებით

OLSR პროტოკოლის უსაფრთხოების გაფართოვება, რაც ზემოთაღნიშნულ სქემას იყენებს, შეიძლება შემდეგის მიხედვით იქნას განსაზღვრული.

(i) ქსელის ფორმირებისას გამოიყენება განაწილებული მასერტიფიცირებელი ორგანო CA (Certificate Authority), რაც უზრუნველყოფს თითოეული კვანძის სათანადო აუთენტიფიკაციას;

(ii) ყოველ ჯერზე, როდესაც ახალი კვანძი შეუერთდება ქსელს, განწილებული CA გამოიყენება კვანძის აუთენტიფიკაციის უზრუნველსაყოფად;

(iii) HELLO შეტყობინებების გადაცემისას ერთ- და ორბიჯიანი მეზობლების ცნობის უზრუნველსაყოფად გათვალისწინებულია მხოლოდ სათანადოდ აუთენტიზირებული კვანძები;

(iv) ყოველი ნაპოვნი აუთენტიზირებული კვანძისთვის რეიტინგის ცხრილს ახალი ჩანაწერი ემატება სიდიდით 100 – მეორადი რეიტინგისა და სიდიდით 50 – პირველადი რეიტინგისთვის;

(v) იგივე, რაც პუნქტები 4, 5, 6 და 7 თავდაპირველი OLSR პროტოკოლისა

(vi) TC შეტყობინების მიღების შემდეგ CPM, რომელიც შეიცავს TC შეტყობინების მიერ გაკვალულ მარშრუტს, უკან იგზავნება წყაროსკენ β ალბათობით;

(vii) იგივე, რაც პუნქტები 8 და 9 თავდაპირველი OLSR პროტოკოლისა

4.5.3. არასათანადო ქცევის კვანძის დადგენა უშუალო დაკვირვებით

არასათანადო ქცევის დადგენა უშუალო დაკვირვებით ხორციელდება შემდეგნაირად: თითოეული კვანძი უსმენს მის MPR გადაცემას. თუ კავშირის საწყისი კვანძი S აღმოაჩენს, რომ MPR არ გადასცემს მის შეტყობინებას, იგი MPR-ის მეორად რეიტინგს ორით ამცირებს და ყველა ერთბიჯიან მეზობელს პოტენციური არასათანადო ქცევის შეტყობინებას უგზავნის. აღნიშნული შეტყობინების მიღების შემდეგ S-ის ყოველი მეზობელი MPR რეიტინგს ერთით ამცირებს. საწინააღმდეგო შემთხვევისას, თუ დადგენილია, რომ MPR გზავნის შეტყობინებას, მისი მეორადი რეიტინგი ერთით იზრდება, მაგრამ მხოლოდ S კვანძის მიერ.

უნდა აღინიშნოს, რომ სასჯელი უფრო დიდია, ვიდრე ჯილდო. გარდა ამისა, ის ფაქტი, რომ მხოლოდ S კვანძი ზრდის მეორად რეიტინგს უშუალო დაკვირვების გზით და ყველა ერთბიჯიანი მეზობელი ამცირებს მას, თუ კვანძი არასათანადოდ იქცევა, კვანძისათვის ართულებს კარგი რეპუტაციის შენარჩუნებას და ზღუდავს ხშირ არასათანადო ქცევას.

კვანძების სათანადო ქცევის მოტივირებისთვის შემდეგნაირად ჩავთვალოთ: პირველადი რეიტინგის შესაბამისად ყოველი კვანძი 5-დან თითოეულ კატეგორიას ეკუთვნის. კვანძის კატეგორია განსაზღვრავს ალბათობას მისი პაკეტების გადაცემისა სხვა კვანძების მიერ, როგორც ეს განსაზღვრულია ცხრილში 1.6.

ცხრილი 1.6 კვანძების კლასიფიკაცია და პაკეტის გადაცემის ალბათობა

კატეგორია	პირველადი რეიტინგის დიაპაზონი	გადაცემის ალბათობა
A	80-100	100%
B	60-80	80%
C	40-60	60%
D	20-40	40%
E	0-20	20%

4.5.4 კვანძის არასაიმედო ქცევის დადგენა CPM-ების ანალიზის მეშვეობით

მიუხედავად იმისა, რომ OLSR კვანძსა და მის MPR-ებს შორის ორმიმართულებიან კავშირს ითვალისწინებს, შემდგომ ჩამოთვლილი სცენარებისას არასათანადო ქცევის დადგენა მეზობელთა უშუალო დაკვირვების გზით შესაძლოა ვერ მოხერხდეს. ეს არის სიტუაციები, როდესაც ადგილი აქვს პაკეტების კოლიზიას, გადაცემის შეზღუდულ ძალმოსილებას, კვანძების შეთქმულებას და პაკეტების ნაწილობრივ დაკარგვას. შესაბამისად, მეორადი რეიტინგი (მოპოვებული სხვა კვანძთა პაკეტების გადაგზავნაზე უშუალო დაკვირვებით) გამოიყენება მხოლოდ არასათანადო კვანძის სტატუსის აღსანიშნად. კვანძის კლასიფიკაციისთვის, როგორც არასაიმედო ქცევის მქონე, გამოიყენება პირველადი რეიტინგი. პირველადი რეიტინგის მოპოვება ხდება მეორადი რეიტინგისა და CPM-ებიდან მიღებული ინფორმაციის კორელაციის გზით.

ჭარბი ინფორმაციის გამოყენებისგან თავის დასაცავად კვანძის, ვთქვათ A კვანძის მიერ CPM-ის მიღების შემდეგ, თუ CPM-ს აქვს მარშრუტი, რომელიც A-მ გაუგზავნა მეზობლებს დროის კონკრეტული მონაკვეთის β განმავლობისას ან იმავე კვანძის მიერ გენერირებული პაკეტი მიღებულ იქნა დროის იმავე მონაკვეთის განმავლობისას, A იშორებს მას. საწინააღმდეგო შემთხვევისას დამუშავება ისევე ხორციელდება, როგორც განსაზღვრულია ალგორითმით (სურ. 4.1.1).

ქვემოთ განხილულია OLSR-ის გაფართოვების არსებული მეთოდის ალგორითმი.

არსებითად, ალგორითმი ადგენს, რომ თუ კვანძი A არის CPM-ის სავარაუდო მიმღები და მან TC შეტყობინება გაგზავნა დროის δ მონაკვეთის განმავლობისას, A პოულობს MPR-ს, რომელსაც მან პაკეტი გადაუგზავნა, ვთქვათ M_1, და ამოწმებს: (ა) M_1–ის შემდეგი ბიჯი მარშრუტში, რომელიც გაწერილია CPM-ში, ეკუთვნის თუ არა M_1-ის MPR ერთობლიობას და (ბ) არის თუ არა ეს კვანძი, ნავარაუდევი A-ს მიმდინარე მარშრუტიზაციის ცხრილით.

თუ ასეა და თუ M_1-ის მეორადი რეიტინგი მეტია, ვიდრე მისი პირველადი რეიტინგი (რაც შეესაბამება კვანძის სათანადო ქცევას), M_1-ის პირველადი რეიტინგი იზრდება მთელი რიცხვით (2/3 x (მეორადი რეიტინგი – პირველადი რეიტინგი)) (ნაბიჯი 6) და მეორად რეიტინგს მიენიჭება პირველადი რეიტინგის მნიშვნელობა.

ალგორითმი CPM-ის დამუშავება

1: SR_{MPR} ←MPR-ის მეორადი რეიტინგი A-ს რეიტინგის ცხრილში

2: PR_{MPR} ← MPR- ის პირველადი რეიტინგი A-ს რეიტინგის ცხრილში

3: ***თუ*** A არის CPM-ის სავარაუდო მიმღები და A-მ TC შეტყობინება გაგზავნა დროის δ მოკლე მონაკვეთის განმავლობისას, ***მაშინ***

4: ***თუ*** CPM-ის ინფორმაცია შესაბამისია A-ს მიერ MPR-დან მოპოვებული ინფორმაციისა, ***მაშინ***

5: ***თუ*** $SR_{MPR} > PR_{MPR,}$ ***მაშინ***

6: $PR_{MPR} \leftarrow PR_{MPR}$ + მთელი რიცხვი ($2/3 \times (SR_{MPR} - PR_{MPR})$)

7: $SR_{MPR} \leftarrow PR_{MPR}$

8: ***საწინააღმდეგო შემთხვევაში***

9: $SR_{MPR} \leftarrow PR_{MPR} + 1$

10: ***პირობითი ოპერატორის დასასრული***

11: ***საწინააღმდეგო შემთხვევაში***

12: ***თუ*** $SR_{MPR} < PR_{MPR}$ ***მაშინ***

13: $PR_{MPR} \leftarrow SR_{MPR}$

14: ***საწინააღმდეგო შემთხვევაში, თუ*** $SR_{MPR} > PR_{MPR}$ ***მაშინ***

15: $SR_{MPR} \leftarrow SR_{MPR} - 2$

16: ***პირობითი ოპერატორის დასასრული***

17: ***პირობითი ოპერატორის დასასრული***

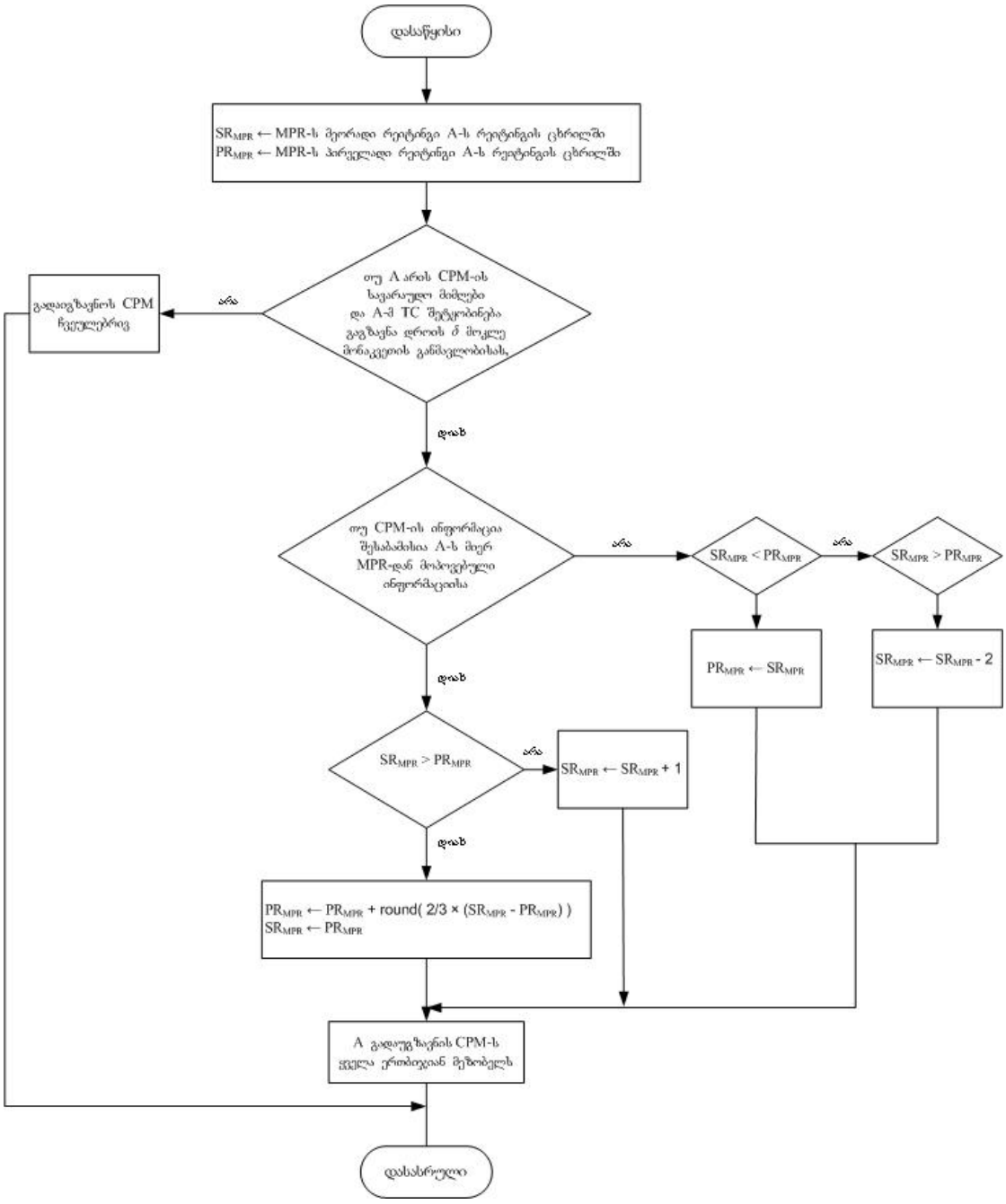
18: A გადაუგზავნის CPM-ს ყველა ერთბიჯიან მეზობელს

19: ***საწინააღმდეგო შემთხვევაში***

20: CPM-ის ჩვეულებრივ გადაგზავნა

21: ***პირობითი ოპერატორის დასასრული***

სურ. 4.1.1 CPM-ის დამუშავება



სურ. 4.1.2. ბლოკსქემა - CPM-ის დამუშავება

(ნაბიჯი 7, ალგორითმი, სურ. 4.1.1). თუ მეორადი რეიტინგი პირველად რეიტინგზე დაბალია (კვანძი ცნობილ იქნა, როგორც არასათანადო ქცევის მქონე), მეორადი რეიტინგის ინფორმაცია შესაძლოა არასწორი იყოს (ვინაიდან კვანძების გადაგზავნის უშუალო დაკვირვება შეცდომებს ექვემდებარება) და მეორადი რეიტინგი ერთით იზრდება (ნაბიჯი 9, ალგორითმი, სურ. 4.1.1).

საწინააღმდეგო შემთხვევაში, თუ CPM-ის ინფორმაცია არ შეესაბამება M_1–ის მიერ გადაცემულს (ნაბიჯი 11, ალგორითმი, სურ. 4.1.1) და M_1–ის მეორადი რეიტინგი პირველადზე დაბალია (არასათანადო ქცევის კვანძი), M_1–ის პირველადი რეიტინგი განისაზღვრება, როგორც მეორადი რეიტინგის სიდიდე (ნაბიჯი 13, ალგორითმი, სურ. 4.1.1). თუ მეორადი რეიტინგი პირველადზე მაღალია, მიიჩნევა, რომ M_1 სათანადოდ იქცევა, მაგრამ რადგან (რაც უფრო მნიშვნელოვანია) CPM საპირისპიროს უჩვენებს, M_1–ის მეორადი რეიტინგი ორით მცირდება (ნაბიჯი 15, ალგორითმი, სურ. 4.1.1). შემდგომ ამისა, A პაკეტს ყველა ერთბიჯიან მეზობელს იმავე დამუშავებისთვის გადაუგზავნის.

4.5.5. ალგორითმის დახასიათება

რადგანაც უსაფრთხოების ალგორითმს მემკვიდრეობით ერგო განაწილებული სერტიფიცირების ორგანოს უპირატესობები, ეს საშუალებას აძლევს მას ყოველი კვანძისა და ყოველი პაკეტის ზუსტი წარმომავლობის იდენტიფიცირება ცენტრალიზებული მიდგომის გარეშე განახორციელოს. აქედან, "იდენტურობის გაყალბების თავდასხმები" აღკვეთილია, ხოლო განმეორებითი თავდასხმებისგან თავდაცვისთვის შესაძლოა დროის ნიშნულის ტრადიციულ მექანიზმებს დავეყრდნოთ. ამ ასპექტებს მიღმა, ეს ალგორითმი, რომელიც კვანძის გადაცემაზე უშუალო დაკვირვებით მოპოვებულ ინფორმაციას (ინფორმაციას, რომელსაც ბოლომდე ვერ დავეყრდნობით, რადგან შეიძლება არაზუსტი ან მცდარი იყოს), უკავშირებს ინფორმაციას, მოპოვებულს წარმატებით მიწოდებული პაკეტების მიერ გაკვალული მარშრუტებით, შემდეგი საკითხების გადაჭრის საშუალებას იძლევა:

- *არხის გაყალბება* გამოიწვევს მტრულად განწყობილი კვანძის დაჯარიმებას. არხის მდგომარეობის შესახებ არასწორი ინფორმაციის გაგზავნისას (HELLO ან TC შეტყობინებების მეშვეობით) CPM-ებში მიღებული მარშრუტები შეუსაბამო იქნება მტრულად განწყობილი კვანძის მიერ მოწოდებული ინფორმაციისა, რაც შეამცირებს მის პირველად რეიტინგს და, შესაბამისად, შეამცირებს მისი კომუნიკაციის შესაძლებლობას;

- *ტრაფიკის გადაცემის უარყოფა* შესაძლოა აღმოჩენილ იქნას მიღებული CPM-ების რაოდენობის და გამგზავნი კვანძის მიერ CPM-ის გაგზავნის ალბათობისა და ქსელის ინტენსივობის კორელაციით;
- *ჭიის ხვრელის* თავდასხმა შესაძლოა ნაწილობრივ იქნას აღმოჩენილი *ტრაფიკის გადაცემის უარყოფის* იმავე ტექნიკის მეშვეობით, თუ მტრულად განწყობილი კვანძი გადაწყვეტს პაკეტების მოშორებას, თუმცა შეტყობინებათა მოდიფიკაციის პრობლემის გადაჭრა უფრო რთულია.

უსაფრთხოების ალგორითმი საკმაოდ რეკონფიგურირებადია უსაფრთხოების მოთხოვნებისა და ტრაფიკის მიმდინარე ხარჯების კუთხით. ქვემოთ ჩამოთვლილი ცვლადები საშუალებას იძლევა პროტოკოლი ზუსტად დარეგულირდეს უსაფრთხოების სასურველი დონის შესაბამისად:

- *CPM გადაცემის ალბათობა*: რაც უფრო მაღალია ალბათობა, მით უსაფრთხოა პროტოკოლი, თუმცა გაზრდილი ტრაფიკის ხარჯზე;
- *ინტერვალი TC-ს გაგზავნასა და CPM-ის მიღებას შორის:* კონფიგურირებულ უნდა იქნას ქსელის სიხშირისა და ზომების შესაბამისად (მსხვილ ქსელებში იგი უნდა გაიზარდოს, საწინააღმდეგო შემთხვევისას ახლო გარემოცვაში გენერირებული CPM-ების მხოლოდ მცირე რაოდენობა იქნება გათვალისწინებული);
- *თაიმაუთი ერთი და იმავე წყაროს მიერ CPM-ების გენერირებას შორის:* ამ სიდიდის კონფიგურირება შეიძლება მოხდეს ქსელის კვანძების სანდოობის ხარისხის შესაბამისად (თუ მტრულად განწყობილი კვანძების სავარაუდო რაოდენობა დიდია, თაიმაუთს უფრო მაღალი მნიშვნელობა უნდა ჰქონდეს, რაც აგვაცილებს განმეორებით გენერირებულ მტრულად განწყობილ CPM-ებს);
- *კვანძების საწყისი პირველადი და მეორადი რეიტინგი:* აღნიშნული სიდიდეები შესაძლოა შეიცვალოს კვანძების სანდოობის საფუძველზე. თუკი ისინი ზოგადად მტრულად განწყობილად მიიჩნევიან, დაბალი პირველადი რეიტინგი აიძულებს მათ სათანადოდ მოიქცნენ, საწინააღმდეგო შემთხვევისას კომუნიკაცია შეუძლებელი იქნება.

ყველაფრის მიუხედავად აღნიშნული OLSR-ის სპეციფიკაცია ხასიათდება ქვემოთ მოყვანილი გარკვეული ნაკლოვანებებით:

1. არ არის გათვალისწინებული აღდგენა არასათანადო ქცევის მდგომარეობიდან;
2. კვანძის ქცევის შემოწმება ხდება მხოლოდ პირდაპირი დაკვირვებითა და CPM შეტყობინების გამოყენებით;
3. გამოიყენება არასათანადო ქცევის შესახებ გამაფრთხილებელი შეტყობინების/სიგნალი;
4. მეთოდს ახასიათებს გარკვეული შეცდომები კვანძების მოძრაობის დროს.

4.6. უსაფრთხოების მოდიფიცირებული ალგორითმის შემუშავება გადაცემადი ინფორმაციის დამახინჯების შემთხვევისათვის

4.6.1. OLSR პროტოკოლზე თავდასხმის ზოგადი განხილვა

წინამორბედი განხილვიდან ჩვენ დავადგინეთ, რომ მარშრუტიზაციის პროტოკოლების უსაფრთხოებისა და იძულების მექანიზმებს უსადენო ქსელების მუშაობისათვის უაღრესი მნიშვნელობა ენიჭებათ. ნაკლოვანებათა ანალიზსა და OLSR პროტოკოლების უსაფრთხოებასთან დაკავშირებულ წინამორბედ შრომაზე დაყრდნობით ჩვენ მოვახდინეთ ორი სახის თავდასხმის იდენტიფიცირება, რომელთათვის არსებობს საყოველთაოდ მიღებული გადაწყვეტილებები: (1) იდენტურობის იმიტირების თავდასხმებთან ბრძოლა შესაძლებელია ხელმოწერისა და გასაღების მენეჯმენტის სისტემებით და (2) განმეორებით თავდასხმებზე რეაგირება შესაძლებელია დროითი ნიშნულის მექანიზმით.

ჩვენი შემდგომი სამუშაო მიმართულია იმ თავდასხმებზე, სადაც კვანძი ყალბ ინფორმაციას აცხადებს არარსებული ლინკების შესახებ იმ კვანძებამდე, რომელთა მიღწევა არ შეუძლია. აღნიშნულ თავდასხმას გააჩნია პოტენციალი გამოიწვიოს მარშრუტის სიგრძის გაზრდა და გააჩინოს კრიტიკული კვანძები, რომლებიც მოგვიანებით შესაძლოა გამოყენებულ იქნან შავი ხვრელის თავდასხმის შესრულების ან ქსელის დაყოფისათვის.

აღნიშნულ საკითხზე რეაგირებისთვის შეგვიძლია გამოვიყენოთ სქემა (მექანიზმი), რომელიც არასათანადო ქცევის კვანძების აღმოჩენისა და დასჯის გზით აიძულებს მარშრუტიზაციის კონტროლის სათანადო ტრაფიკის გენერირებას. მიუხედავად იმისა, რომ რეპუტაციის მექანიზმები უკვე იქნა განხილული, პრაქტიკულად ყველა შემთხვევა, რომელიც რეაქტიული მარშრუტიზაციის პროტოკოლის მიმართ იქნა გამოყენებული, ეყრდნობოდა მხოლოდ "მეთვალყურის", როგორც მონიტორინგის მექანიზმს.

როგორც სხვადასხვა ავტორების ნამუშევრებშია მოყვანილი, "მეთვალყურის", როგორც მონიტორინგის იარაღის, გამოყენებამ შესაძლოა ვერ აღმოაჩინოს არასათანადო ქცევის კვანძები შემდეგ შემთხვევებში: (1) კოლიზიები, (2) გადაცემის შეზღუდული შესაძლებლობა, (3) შეთქმულება და (4) პაკეტების ნაწილობრივი მოშორება. უფრო მეტიც, იგი იძლევა არასათანადო ქცევის მხოლოდ ადგილობრივად დადგენის შესაძლებლობას და, შესაბამისად, არასათანადო კვანძების

გაცხადებისთვის საგანგაშო სიგნალების გავრცელებაზეა დამოკიდებული. აღნიშნული სიგნალები კი შესაძლოა არასწორად გამოყენებულ იქნას უფლებამოსილი კვანძების მცდარად მუშაობაში ბრალდებისათვის.

დავახასიათოთ აქტიური თავდამსხმა. აქტიური თავდამსხმელი არის ქსელის ჩვეულებრივი კვანძი და, შესაბამისად, მისთვის მარშრუტიზაციის იგივე ინფორმაციაა მისაწვდომი, რაც ქსელის ყველა კვანძისთვის. მას, ისევე, როგორც ყველა სხვა კვანძს, შეუძლია ქსელში მარშრუტიზაციის ინფორმაციის შეტანა, რომელიც მეზობელ კვანძებს მიაღწევს (გავრცელების მექანიზმების გამოყენებით). თავდამსხმელის განზრახვას სურვილის მიხედვით მარშრუტიზაციის პროტოკოლის დაზიანება ან შეცვლა წარმოადგენს.

დავუშვათ, რომ კვანძების აუთენტიფიკაცია კავშირის დროს ხდება (მაგალითად, გასაღებთა განაწილებით კავშირის დამყარებამდე, როგორც რეკომენდებულია [48]-ში) და, ამდენად, არ შეუძლიათ სხვა კვანძების იმიტირება ან კომუნიკაციისთვის რამდენიმე იდენტიფიკატორის გამოყენება (Sybil თავდასხმა). უფრო მეტიც, განმეორებითი თავდასხმების პრევენცია დროითი ნიშნულის მექანიზმებით ხდება, როგორიც [47]-ში და [49]-შია.

4.6.2. მოდიფიცირებული OLSR პროტოკოლის გაფართოვება და სპეციფიკაცია

ფუნდამენტური საკითხი, რომელიც წინამდებარე სქემის საფუძვლად დევს, არის ის, რომ კვანძებმა კორექტულად მოახდინონ OLSR კონტროლის ტრაფიკის გენერირება. აღნიშნული მიზნის მისაღწევად ძირითად პრინციპს წარმოადგენს დაჯილდოვება იმ კვანძებისა, რომლებიც შეესაბამებიან მარშრუტიზაციის პროტოკოლს, და დასჯა დამაზიანებელი ქმედების კვანძებისა ქსელის მისაწვდომობის კუთხით [56], [57], მაგალითად, არასათანადო ქცევის კვანძებისთვის ქსელით კომუნიკაციის შესაძლებლობის შეზღუდვა.

ამ მიზნით ზემოთმოყვანილი მეთოდიდან ჩვენ ვიყენებთ მხოლოდ ორი ელემენტს:

- *CPM შეტყობინება: CPM* შეტყობინება გამოიყენება კონტროლის ტრაფიკის შეტყობინების მიერ გაკვალული მარშრუტის ქსელში გადასატანად. TC შეტყობინების მიღების შემდეგ, ქვემოთ აღწერილი წესების შესაბამისად, თითოეული MPR კვანძი *CPM* შეტყობინებას უკან უგზავნის TC

შეტყობინების გამგზავნს. იგი შეიცავს TC შეტყობინების მიერ გაკვალულ მარშრუტს და, შესაბამისად, ახდენს მის მიერ გაკვალული მარშრუტის ჩაწერას, როდესაც იგი ქსელში იკვალავს გზას;

- *რეიტინგის ცხრილი:* ქსელის თითოეული კვანძი აწარმოებს რეიტინგის ცხრილს, რომელშიც ინახება ინფორმაცია ქსელის კვანძების ქცევის შესახებ. რეიტინგის ცხრილის ყოველ ჩანაწერს აქვს კვანძის ID, პირველადი და მეორადი რეიტინგები. კვანძის ID ახდენს ქსელის კვანძის უნიკალურ იდენტიფიცირებას; მეორადი რეიტინგი არის კვანძის კლასიფიკაცია პაკეტების გადაგზავნის უშუალო დაკვირვებაზე დაყრდნობით, ხოლო პირველადი რეიტინგი გახლავთ კვანძის უფრო ზუსტი კლასიფიკაცია, დაფუძნებული მისი მეორადი რეიტინგის მნიშვნელობაზე, CPM შეტყობინებების მიერ მოწოდებული ინფორმაციის ანალიზსა და ადგილობრივი მარშრუტიზაციის ინფორმაციაზე, რომელსაც ინახავენ კვანძები. მოცემული რეიტინგები გამოიყენება კვანძების სათანადო ქცევის მოტივირებისათვის და განსაზღვრავენ კვანძების სურვილს გადასცენ თუ არა ტრაფიკი სხვათა სახელით, ანუ კვანძები ტრაფიკის დიდ ნაწილს გადასცემენ მაღალი რეიტინგის კვანძების სახელით და უარს ამბობენ აღნიშნულის განხორციელებაზე დაბალი რეიტინგის კვანძებისთვის.

OLSR პროტოკოლის უსაფრთხოების ჩვენს მიერ მოდიფიცირებული გაფართოვება, შეიძლება განსაზღვრულ იქნას, როგორც ეს ნაჩვენებია ცხრილში 1.7. როგორც ჩანს ცხრილიდან, ნაბიჯები 4-6, 9 და 11 მიეკუთვნება OLSR ჩვეულ ოპერაციებს, როდესაც დანარჩენები წარმოდგენილია, როგორც უსაფრთხოების მოდიფიცირებული სქემის ნაწილები.

ცხრილი 1.7 უსაფრთხოების მოდიფიცირებული ალგორითმის ფუნქციონირება

1) ქსელის ფორმირებისას ხელმოწერისა და გასაღების მენეჯმეტის მექანიზმების გამოყენება, რაც თითოეული კვანძის სათანადო აუთენტიკაციის გარანტიას იძლევა;
2) HELLO შეტყობინების გადაცემის განმავლობისას, რაც

უზრუნველყოფს ერთ და ორბიჯიანი მეზობლების დადგენას, გათვალისწინებულია მხოლოდ სათანადოდ აუთენტიზირებული კვანძები (ხელმოწერის მექანიზმის გამოყენებით);
3) ყველა აღმოჩენილი აუთენტიზირებული კვანძისთვის რეიტინგის ცხრილს ემატება ახალი ჩანაწერი α პირველადი რეიტინგისთვის და β – მეორადი რეიტინგისთვის;
4) HELLO შეტყობინების ინფორმაციის გამოყენებით თითოეული კვანძი ახდენს MPR ერთობლიობის შერჩევას, რომლის გაცხადება მომდევნო HELLO შეტყობინებებში ხდება;
5) აღნიშნული ინფორმაციის გამოყენებით თითოეული კვანძი აგებს მის MPR შემრჩევთა ერთობლიობას იმ კვანძების მითითებით, რომლებმაც იგი MPR-დ აირჩიეს;
6) თითოეული კვანძის მიერ ქსელში ხდება TC-ს გავრცელება და ამით MPR შემრჩევთა ერთობლიობის გაცხადება;
7) TC-ს გადაგზავნაზე უშუალო დაკვირვებით ხდება არასათანადო ქცევის კვანძის აღმოჩენა დაკვირვების მექანიზმის გამოყენებით, რომელიც აღწერილი იყო როგორც "მეთვალყურის" კონცეფცია;
8) TC შეტყობინების მიღების შემდეგ შესაძლოა უკან იქნას გაგზავნილი CPM შეტყობინება, რომელიც მოიცავს გაგზავნილი TC შეტყობინების მიერ გაკვალულ მარშრუტს, ამ CPM შეტყობინების გაგზავნის ალბათობა დამოკიდებულია CPM შეტყობინების გადაგზავნის მაჩვენებელზე λ,
9) მიღებული TC შეტყობინების გამოყენებით ყოველი კვანძი ადგენს ტოპოლოგიის ცხრილს, რომელიც შედგება დანიშნულების იდენტიფიკატორის (MPR შემრჩევი TC შეტყობინებაში), დანიშნულებამდე უკანასკნელი ბიჯის კვანძის იდენტიფიკატორისა (TC-ს ორიგინატორი) და MPR შერჩევის ერობლიობის რიგითი ნომრის ჩანაწერებისგან;
10) როდესაც CPM შეტყობინების მიღება ხდება, იგი მუშავდება CPM შეტყობინების დამუშავების ალგორითმის შესაბამისად (სურ. 4.1.3);
11) ტოპოლოგიის ცხრილი შემდეგ გამოიყენება მარშრუტიზაციის

ცხრილის გამოთვლის ალგორითმის მიერ თითოეულ კვანძამდე მარშრუტიზაციის ცხრილის გამოსათვლელად.

როგოც ჩანს მოცემული ცხრილიდან, პირველი ნაბიჯი ემსახურება კვანძის აუთენტიფიკაციას. მეორე ნაბიჯზე HELLO შეტყობინების საშუალებით წარმოებს ერთ და ორბიჯიანი მეზობლების დადგენა. მესამე ნაბიჯზე ხდება პირველადი და მეორადი რეიტინგებისათვის მნიშვნელობების მინიჭება. პირველადი და მეორადი რეიტინგები მერყეობს 0-დან 100-მდე, სადაც 100 საუკეთესო შესაძლო სიდიდეა, რომლის მიღწევა კვანძის რეიტინგებს შეუძლია. საწყისი პირველადი რეიტინგი α და მეორადი რეიტინგი β მესამე ნაბიჯზე არსებითად განსაზღვრავს თითოეული კვანძის საწყის სანდოობის დონეს. თუ განვიხილავთ ქსელს შემსრულებელი კვანძებით, თავდაპირველად მათ შეგვიძლია მაღალი სიდიდეები მივანიჭოთ. წინააღმდეგ შემთხვევისას, მათთვის დაბალი სიდიდეების მინიჭებით ვაიძულებთ არასათანადო ქცევის მდგომარეობიდან გამოვიდნენ ქსელის ფორმირების პროცესის დროს.

მეოთხე ნაბიჯზე წარმოებს MPR-ების ერთობლიობის დადგენა, ხოლო შემდეგ მეხუთე ნაბიჯზე – MPR-ის შემრჩევთა ერთობლიობის დაგენა. ამის შემდეგ მეექვსე ნაბიჯზე ხდება TC შეტყობინების გადაცემა, რომლის საშუალებითაც ქსელში ვრცელდება ტოპოლოგიური ინფორმაცია. TC შეტყობინებების მიერ ქსელში მარშრუტის გაკვალვისას მათ უნდა დააფიქსირონ მარშრუტი, რომელიც გაიარეს. აღნიშნულის განხორციელება შემდეგი წესით ხდება: თითოეული კვანძი, ჩვეული წესით შეტყობინების გადაგზავნამდე, ამატებს თავის იდენტიფიკატორს მარშრუტში, რომელიც აკუმულირდება TC შეტყობინებაში.

TC შეტყობინების გადაცემის შემდეგ მეშვიდე ნაბიჯზე მისი გამგზავნი აწარმოებს დაკვირვებას მის MPR-ზე, რათა დაადგინოს გადასცემს თუ არა ის მის მიერ გაგზავნილ TC-შეტყობინებას. ანუ გამოიყენება დაკვირვების მექანიზმი.

როგორც ჩვენ უკვე აღვნიშნეთ, "მეთვალყურის" მექანიზმი ეფუძნება თითოეული კვანძის მიერ MPR-ს გადაცემის არაერთგავროვნად მოსმენას შემდეგის მიხედვით: როდესაც კვანძი ქსელში გადასცემს TC შეტყობინებებს, იგი განაგრძობს MPR-ის მოსმენას. თუ კვანძი დაადგენს, რომ MPR არ გადასცემს მის პაკეტს, იგი MPR-ს მეორად რეიტინგს m-ით ამცირებს. საწინააღმდეგო შემთხვევისას მეორადი რეიტინგი *n*-ით იზრდება. გადაგზავნასთან დაკავშირებით

თანამშრომლობის სტიმულირებისთვის სასჯელი უფრო დიდი უნდა იყოს, ვიდრე ჯილდო.

აღნიშნული მექანიზმი შეცდომებს არის დაქვემდებარებული და, შესაბამისად, ჩვენ გამოვიყენეთ იგი, რათა მოხდეს ცვლილებები მეორად რეიტინგში, რაც, როგორც ეს მოგვიანებით იქნება ნაჩვენები, გამოიყენება მხოლოდ იმის განსაზღვრისათვის, თუ რამდენად სწრაფად აღდგება კვანძი არასათანადო ქცევის მდგომარეობიდან.

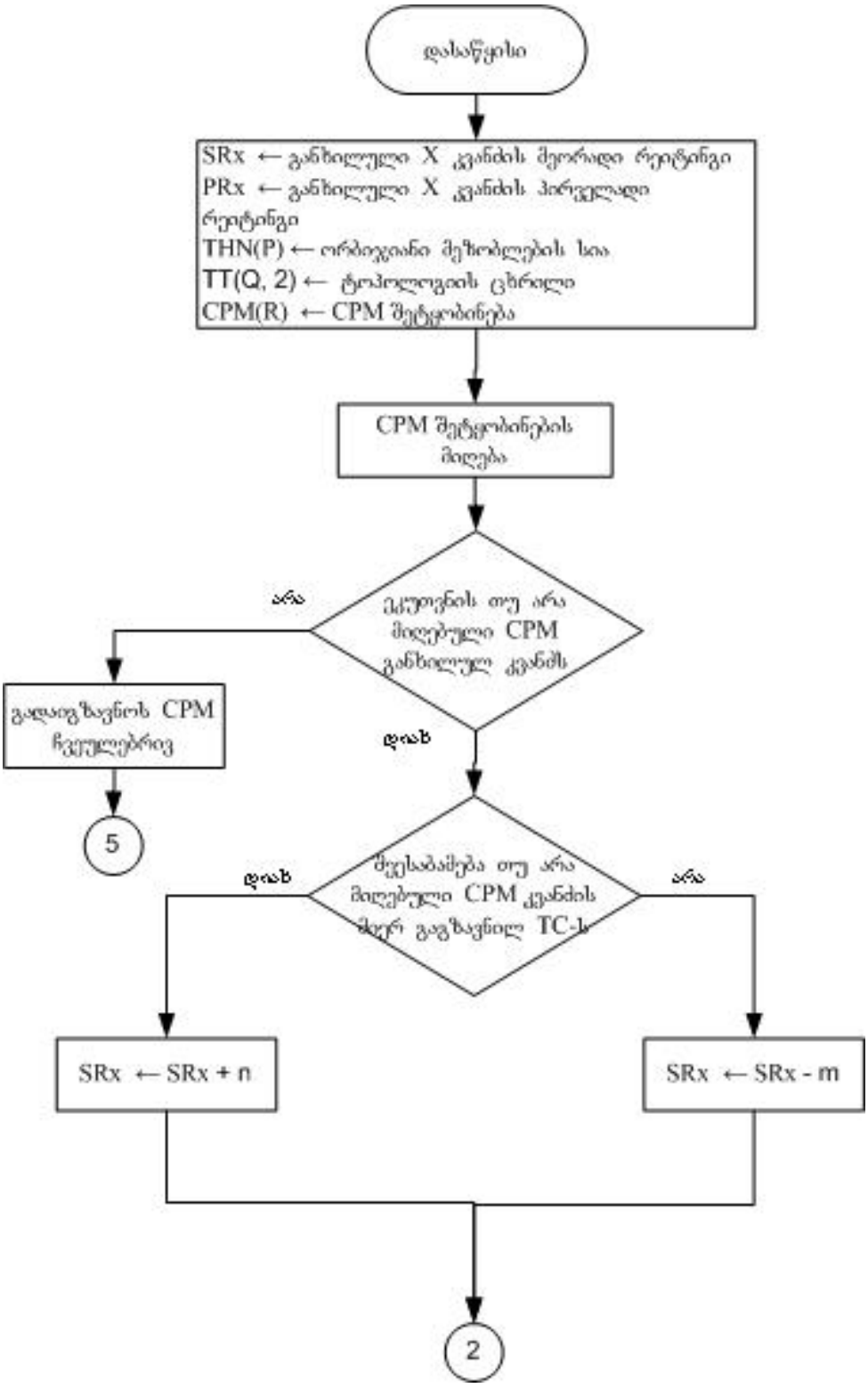
მერვე ნაბიჯზე, TC შეტყობინების მიმღების მიერ იგზავნება CPM შეტყობინება გარკვეული ალბათობით. აქ CPM შეტყობინების მაჩვენებელი λ განსაზღვრავს CPM შეტყობინებების რაოდენობას, რომელთა გენერირება ქსელის კვანძების მიერ TC შეტყობინებების მიღების პასუხად მოხდება. მეცხრე ნაბიჯზე წარმოებს ტოპოლოგიური ცხრილის აგება.

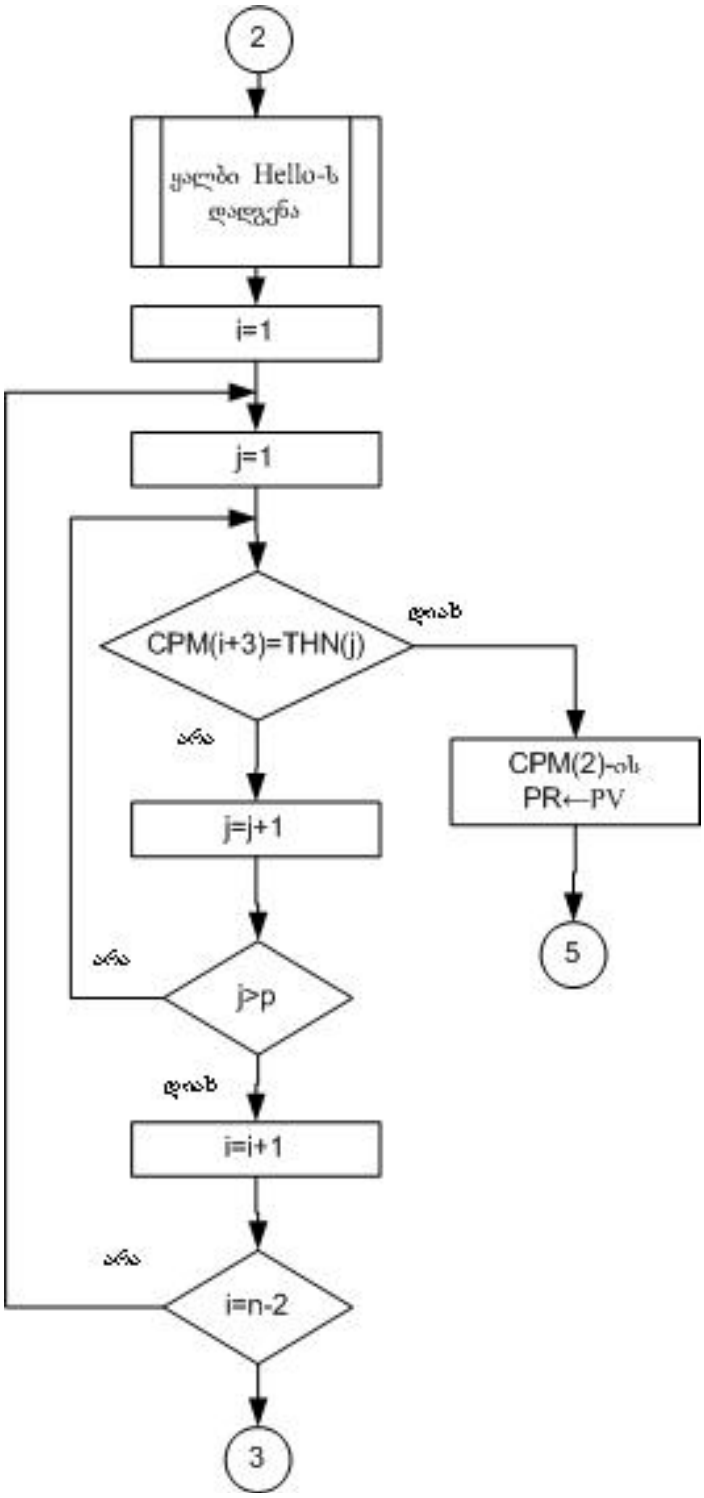
4.6.3. უსაფრთხოების მოდიფიცირებული ალგორითმი

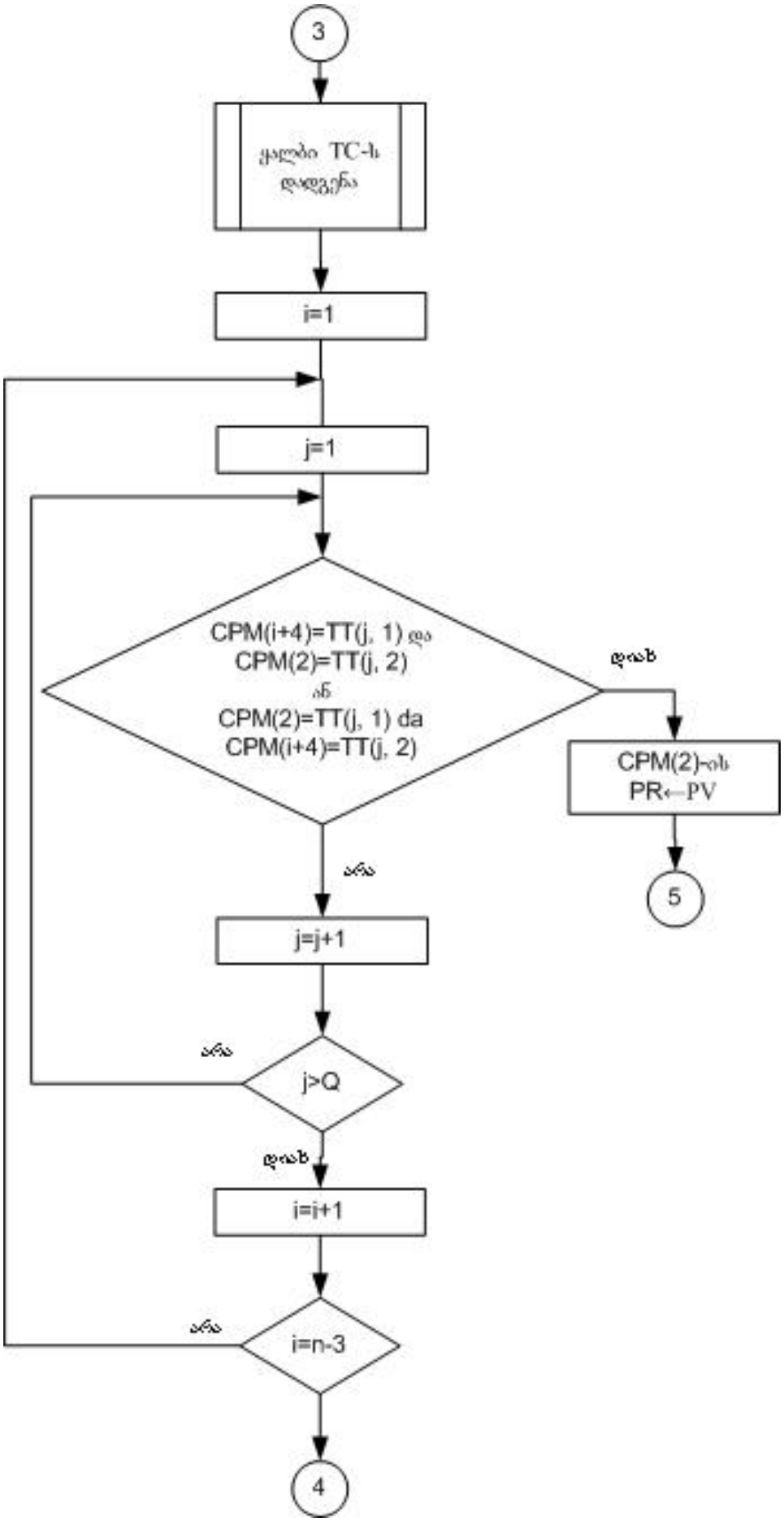
ნაშრომის ყველაზე მნიშვნელოვან ნაწილს CPM შეტყობინების დამუშავების მექანიზმი წამოადგენს. ეს გახლავთ სანდო მონიტორინგის მექანიზმი, რომელიც ეფუძნება CPM შეტყობინებებს, გენერირებულს მარშრუტიზაციის კონტროლის ტრაფიკის საპასუხოდ, რაც OLSR-ს შემთხვევაში TC-ს შეესაბამება.

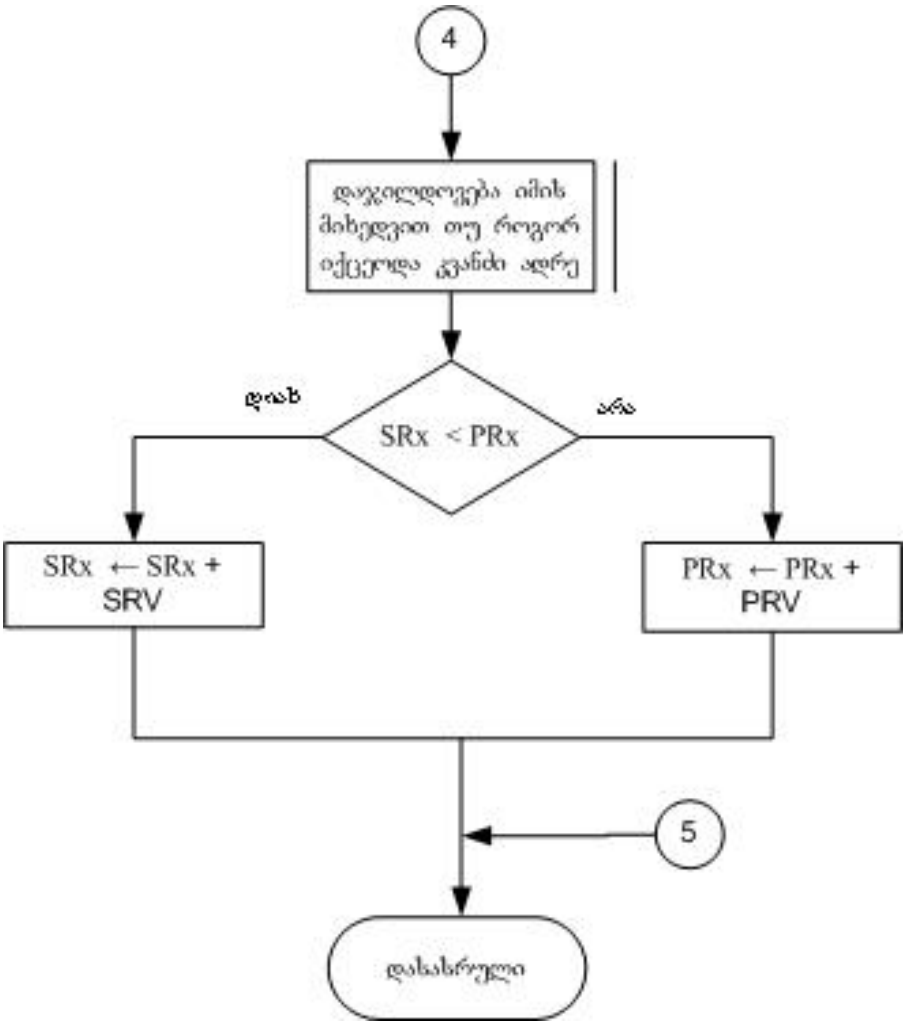
ალგორითმი CPM შეტყობინების დამუშავების მოდიფიცირებული ალგორითმი
1: SR_x – X განსახილველი კვანძის მეორადი რეიტინგი,
2: PR_x – X განსახილველი კვანძის პირველადი რეიტინგი,
3: მიღებული CPM-ის საკუთრების განსაზღვრა
4: თუ საჭიროა, S-კვანძის მეორადი რეიტინგის ცვლილებები პირდაპირი დაკვირვების მექანიზმის გამოყენებით
5: HELLO-სა და TC-ის კონტროლი
6: **თუ** ყალბი HELLO ან TC შეტყობინებების გენერირების აღმოჩენის მექანიზმმა S კვანძი განსაზღვრა, როგორც არასათანადო ქცევის კვანძი, **მაშინ**
7: $PR_x = PV$
8: **საწინააღმდეგო შემთხვევაში**
9: **თუ** $SR_x < PR_x$, **მაშინ**
10: $SR_x = SR_x + SRV$
11: **საწინააღმდეგო შემთხვევაში**
12: $PR_x = PR_x + PRV$
13: **პირობითი ოპერატორის დასასრული**
14: **პირობითი ოპერატორის დასასრული**

სურ. 4.1.3. CPM შეტყობინების დამუშავების მოდიფიცირებული ალგორითმი


დასაწყისი
SRx ← განხილული X კვანძის მეორადი რეიტინგი
PRx ← განხილული X კვანძის პირველადი რეიტინგი
THN(P) ← ორბიტალური მეზობლების სია
TT(Q, 2) ← ტოპოლოგიის ცხრილი
CPM(R) ← CPM შეტყობინება
CPM შეტყობინების მიღება
არა
ემთხვევა თუ არა მიღებული CPM განხილულ კვანძს
გადაგზავნოს CPM ჩვენსებრივ
5
დიახ
დიახ
მეზობლობა თუ არა მიღებული CPM კვანძის მიერ გაგზავნილ TC-ს
არა
SRx ← SRx + n
SRx ← SRx - m
2


2
ყალბი Hello-ს დაბეჭდვა
i=1
j=1
CPM(i+3)=THN(j)
დიახ
არა
CPM(2)-ის PR←PV
j=j+1
5
არა
j>p
დიახ
i=i+1
არა
i=n-2
3


3
ქალბი TC-ს
დადგენა
i=1
j=1
CPM(i+4)=TT(j, 1) და
CPM(2)=TT(j, 2)
ან
CPM(2)=TT(j, 1) da
CPM(i+4)=TT(j, 2)
დიახ
CPM(2)-ის
PR←PV
5
არა
j=j+1
არა
j>Q
დიახ
i=i+1
არა
i=n-3
4



სურ. 4.1.4. ბლოკსქემა - CPM შეტყობინების დამუშავების მოდიფიცირებული ალგორითმი

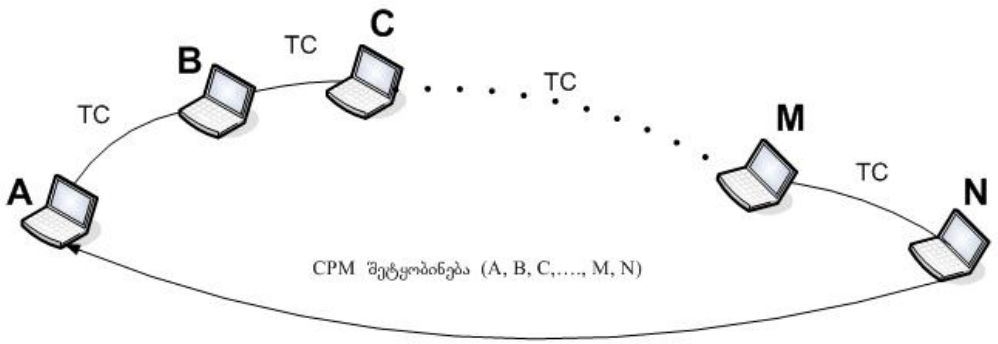
როდესაც ხდება CPM შეტყობინების მიღება, მისი დამუშავება ხდება ალგორითმის მიხედვით (სურ. 4.1.3). ალგორითმი ადგენს, რომ, თუ კონკრეტული კვანძი გაცხადებულ იქნება, როგორც მარშრუტიზაციის ყალბი ინფორმაციის გენერატორი (ნაბიჯი 6), მის პირველად რეიტინგს მიენიჭება დასჯის სიდიდე *PV (Punishment Value)* (ნაბიჯი 7). საწინააღმდეგო შემთხვევაში, თუ აღმოჩნდება, რომ კვანძმა მარშრუტიზაციის სათანადო ინფორმაციის გენერირება მოახდინა, მისი რეპუტაცია იზრდება (ნაბიჯები 9-12).

ეხლა კი განვიხილოთ, თუ როგორ ხდება ყალბი შეტყობინებების გენერირების დადგენა.

ყალბი HELLO-ს გენერირების დადგენა ეყრდნობა ინფორმაციის ორი წყაროს კორელაციას: CPM შეტყობინებებიდან მოპოვებულ მარშრუტებს და

HELLO-დან მიღებულ და *ორბიჯიანი მეზობლების ერთობლიობაში* შენახულ ადგილობრივ ინფორმაციას. ვინაიდან HELLO შეტყობინებების გაცვლა მხოლოდ უშუალო მეზობლებს შორის ხდება და მხოლოდ კვანძის MPR გადასცემს მის ტრაფიკს, ამ მექანიზმისთვის ალგორითმში (სურ. 4.1.3) განხილული დაკვირვების ქვეშ მყოფი კვანძები მიმდინარე კვანძის MPR-ს წარმოადგენენ.

განვიხილოთ სურათზე 4.2 ნაჩვენები სცენარი, სადაც კვანძმა *C* მოახდინა TC შეტყობინების გენერირება და ახლა ქსელის ერთ-ერთი კვანძისგან CPM შეტყობინებას იღებს.



სურ. 4.2. CPM შეტყობინების ილუსტრირება

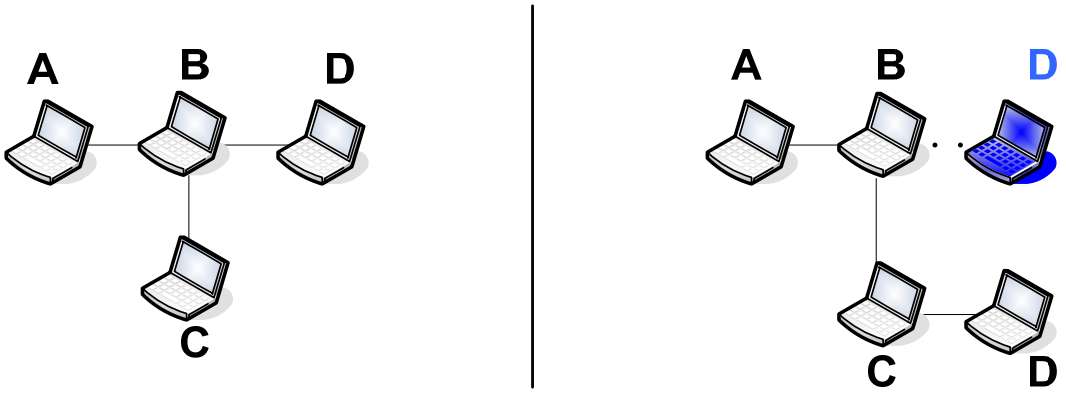
დავუშვათ, რომ *B* არის *A*-ს MPR, რომელიც CPM შეტყობინების მარშრუტში დევს (ანუ *B* იყო *A*-ს TC შეტყობინების გადამგზავნი, რომელმაც შექმნა CPM მიმდინარე შეტყობინება). ყალბი HELLO შეტყობინების აღმოჩენის პროცედურა შემდეგია:

1) *A* იღებს CPM შეტყობინებას, რომელიც შეიცავს მის მიერ ქსელისთვის გაგზავნილი TC შეტყობინების მარშრუტს;
2) *A* *B*-სგან ორი და მეტი ბიჯით დაშორებული ყველა M კვანძისთვის ამოწმებს, არის თუ არა მის 2-ბიჯიან ერთობლიობაში ჩანაწერი, რომელიც ამბობს, რომ *B*-ს პირდაპირი კავშირი აქვს M -თან;
3) თუ ასეა, *B* არასათანადო ქცევის კვანძია, რადგან მან HELLO შეტყობინების მეშვეობით პირდაპირი კავშირი განაცხადა M-თან, ხოლო M არ არის უშუალოდ მისაწვდომი *B*-სთვის;
4) საწინააღმდეგო შემთხვევაში *B* სათანადო ქცევის კვანძად ჩაითვლება;

5) იმის გათვალისწინებით, არის *B* სათანადო თუ არასათანადო ქცევის კვანძია, *B*-ს რეპუტაცია შესაბამისად იცვლება, როგორც ეს ალგორითმშია ნაჩვენები (სურ. 4.1.3).

აღნიშნულ მიდგომასთან დაკავშირებით ერთი მნიშვნელოვანი საკითხი უნდა აღინიშნოს. OLSR კვანძების მიერ შენახული ადგილობრივი ინფორმაცია ეფუძნება კონტროლის ტრაფიკის პერიოდულ გაცვლას. კვანძების მოძრაობასთან ერთად არსებობს გარდამავალი მდგომარეობა, როდესაც ქსელის რეალური მდგომარეობა და მის შესახებ არსებული ლოკალური ინფორმაცია ერთმანეთს არ შეესაბამება.

მაგალითისთვის განვიხილოთ სურათი 4.3. ჩავთვალოთ, რომ სურათის მარცხენა მხარეს ნაჩვენები B არის A-ს MPR, ხოლო C და D არიან B-ს MPR-ები. D მოძრაობს და გადის B-ს გადაცემის ფარგლებს გარეთ და ხვდება C-ს გადაცემის საზღვრებში, ამგვარად იგი ხდება C-ს MPR (სურ. 4.3 მარჯვენა მხარე). იმავდროულად ადგილი არ აქვს კონტროლის ტრაფიკის პერიოდულ გაცვლას და, შესაბამისად, A-სთვის კვლავაც უცნობია ტოპოლოგიის ეს ცვლილება. A ქსელში გზავნის TC შეტყობინებას მარშრუტით A-B-C-D და D ახდენს CPM შეტყობინების გენერირებას, რაც ამ მარშრუტს შეიცავს. ვინაიდან ადგილობრივი ინფორმაცია გვეუბნება, რომ B-ს შეუძლია D-ს მიღწევა (რადგან A-ს ადგილობრივი ინფორმაციის განახლება ჯერ არ მომხდარა), აღნიშნული შედეგად გვაძლევს არასათანადო ქცევის აღმოჩენის ყალბ შესაძლებლობას, სადაც B არასათანადო ქცევის კვანძად ჩაითვლება.



სურ 4.3. MPR-ის გარდამავალი მდგომარეობა

შეიძლება აღინიშნოს, რომ ქსელის მუშაობის პროცესში ყალბი შესაძლებლობები ნაკლებად გვხვდება და გაცილებით იშვიათია, ვიდრე

არასათანადო ქცევის სწორად აღმოჩენა. მათი რაოდენობის კვლავაც შემცირების ერთ-ერთი სავარაუდო გადაწყვეტილებაა კონტროლის ტრაფიკის გენერირების ინტერვალების შემცირება მარშრუტიზაციის კონტროლის ტრაფიკის გაზრდის ხარჯზე. აღნიშნული შედეგად გვაძლევს კონტროლის ტრაფიკის უფრო ხშირ გენერირებას, რაც გააიოლებს რეალური ქსელის უფრო განახლებულ ხედვას და, მოგვიანებით, საშუალებას მოგვცემს შემცირდეს ყალბი შესაძლებლობების რაოდენობა.

მაგრამ შეიძლება იყოს გამოყენებული სხვა გზაც, რომელიც განხილულია შემდგომ.

ყალბი TC შეტყობინების გენერირების დადგენა ეფუძნება ინფორმაციის ორ წყაროს: CPM შეტყობინებებიდან მიღებულ მარშრუტებს და TC შეტყობინებების ადგილობრივ ინფორმაციას, შენახულს *ტოპოლოგიათა ცხრილში.* ვინაიდან TC-ს გადაცემა ქსელის ყველა კვანძისთვის ხდება, აღნიშნულ მექანიზმში განსახილველი კვანძები წარმოადგენენ ყველა კვანძს C შეტყობინების მარშრუტში. აღნიშნული საშუალებას გვაძლევს შემდგომი მექანიზმის მეშვეობით აღმოვაჩინოთ კვანძები, რომლებმაც ყალბი TC შეტყობინებები შექმნეს: დავუშვათ, რომ A არის ქსელის კვანძი, რომელიც ქსელიდან CPM შეტყობინებას იღებს. ყალბი TC შეტყობინების გენერირების აღმოჩენის პროცედურა შესაძლოა შემდეგის მიხედვით იქნას აღწერილი:

1) *A* იღებს CPM შეტყობინებას, რომელიც შეიცავს რომელიმე კვანძის მიერ ქსელისთვის გაგზავნილი TC შეტყობინების მარშრუტს;

2) CPM შეტყობინების მარშრუტის ყველა B კვანძისთვის და B-სგან სამი ან მეტი ბიჯით დაშორებული ყველა M კვანძისთვის მარშრუტში, A ამოწმებს, არის თუ არა *ტოპოლოგიის ცხრილში* ჩანაწერი იმის შესახებ, რომ B-ს პირდაპირი კავშირი აქვს M -თან;

3) თუ ასეა, *B* არასათანადო ქცევის კვანძია, რადგან მან TC შეტყობინების მეშვეობით პირდაპირი კავშირი განაცხადა M -თან, ხოლო M არ არის უშუალოდ მისაწვდომი *B*-სთვის;

4)საწინააღმდეგო შემთხვევაში *B* სათანადო ქცევის კვანძად ჩაითვლება;

5) იმის გათვალისწინებით, არის *B* სათანადო თუ არასათანადო ქცევის კვანძი, *B*-ს რეპუტაცია შესაბამისად იცვლება, როგორც ეს ალგორითმშია ნაჩვენები (სურ. 4.1.3).

ყალბი TC შეტყობინების გენერირების აღმოჩენა ასევე გავლენას ახდენს ჩვენს მიერ ზემოთ ნახსენები MPR–ის გარდამავალი მდგომარეობის პრობლემაზე. ერთ-ერთ შესაძლო გადაწყვეტილებას წარმოადგენს კონტროლის ტრაფიკის გენერირების ინტერვალის შემცირების იმავე ტექნიკის გამოყენება გარკვეული გაზრდილი დანახარჯების ხარჯზე. მიუხედავად ამისა, ჩვენ განსხვავებული მიდგომა ავირჩიეთ, რაც თავიდან გვაცილებს ტრაფიკის გაზრდით გამოწვეული ხარჯების მატებას. ჩვენი მიდგომა მოცემული საკითხის გადასაწყვეტად უკვე იყო შემოთავაზებული პროცედურის მეორე ნაბიჯზე, სადაც ყალბი TC გენერირების აღმოჩენა ხდებოდა და ა.შ. ნაცვლად კვანძებს შორის კავშირის ანალიზისა, რომლებიც ორი ან სამი ბიჯით არიან დაშორებული სავარაუდო არასათანადო ქცევის კვანძისგან, ჩვენ ვაანალიზებთ ამას სამი ან მეტი ბიჯით დაშორებული კვანძებისთვის. აღნიშნული ვარიანტი MPR–ის გარდამავალი მდგომარეობის შემთხვევათა რაოდენობის შემცირებით წარმატებით ზღუდავს ყალბი შესაძლებლობების რაოდენობას, მაგრამ საშუალებას აძლევს არასათანადო ქცევის კვანძს გააყალბოს ორი ბიჯით დაშორებულ კვანძებთან არსებული კავშირები. მიუხედავად ამისა, აღნიშნული ჩვენ დასაბუთებულ კომპრომისად მიგვაჩნია, რადგან მიღებული ყალბი შესაძლებლობების რაოდენობა ძალიან დაბალია და რადგან კვანძებამდე კავშირის გაყალბებით არასათანადო ქცევის კვანძს ორი ბიჯით დაშორებულ კვანძებთან არსებული კავშირების გააყალბებით მხოლოდ მარშრუტის გაგრძელება შეუძლია ერთი ერთეულით.

განვიხილოთ არასათანადო ქცევის კვანძების დასჯის საკითხი. იმის დადგენის შემდეგ, არის თუ არა კვანძი არასათანადო ქცევის, გატარებულ უნდა იქნას სათანადო ღონისძიებები. როგორც ალგორითმის (სურ. 4.1.3) მეშვიდე ნაბიჯზე არის ნაჩვენები, როდესაც კვანძი არასათანადოდ იქცევა, მისი პირველადი რეიტინგი უტოლდება სასჯელის სიდიდეს (PV-Punishment Value). პირველადი რეიტინგი მერყეობს 0-100 ფარგლებში, სადაც 100 საუკეთესო შესაძლო სიდიდეა კვანძისთვის. კვანძების სათანადო ქცევის მოტივირებისთვის პირველადი რეიტინგი ქსელის კვანძების მიერ გამოიყენება მათი სურვილის განსაზღვრისათვის გადაგზავნონ სხვა კვანძების ტრაფიკი. აღნიშნული ხორციელდება სხვა კვანძების ტრაფიკის გადაცემით მათი პირველადი რეიტინგის შესაბამისად. მაგალითად, კვანძი A, რომელიც დაადგენს, რომ B-ს პირველადი რეიტინგი 40-ის ტოლია, გადაგზავნის B-ს პაკეტების მხოლოდ 40%-ს.

რაც შეხება არასათანადო ქცევის კვანძების აღდგენას: აღდგენის მექანიზმი საშუალებას აძლევს კვანძს, რომელიც წყვეტს არასათანადო ქმედებას, გამოვიდეს არასათანადო ქცევის მდგომარეობიდან. ამიტომ არის მეორადი რეიტინგი (რომელიც იცვლება "მეთვალყურე" მექანიზმის შესაბამისად) გამოყენებული. აღნიშნული მექანიზმი იწვევს იმ კვანძების *თანდათანობით აღდგენას,* რომლებმაც უარყვეს სხვა კვანძების სახელით კონტროლის ტრაფიკის გადაცემა. სრული პროცედურა, აღწერილი ალგორითმის ნაბიჯებში 9-12 (სურ. 4.1.3), შემდეგია: თუ აღსადგენი კვანძის მეორადი რეიტინგი პირველადზე დაბალია, მხოლოდ მეორადი რეიტინგი იზრდება SRV-თი (Secondary Rating Value – მეორადი რეიტინგის აღდგენის სიდიდე), სანამ პირველადი რეიტინგის სიდიდეს არ მიაღწევს. დროის ეს პერიოდი წარმოადგენს იმ კვანძების აღდგენის დროს, რომლებმაც უარყვეს კონტროლის ტრაფიკის გადაცემა. როგორც კი მეორადი რეიტინგი მიაღწევს პირველად რეიტინგზე მაღალ სიდიდეს, არასათანადო ქცევის კვანძი მაშინვე ეფექტურად იწყებს აღდგენას, მისი პირველადი რეიტინგის PRV-თი (Primary Rating Value - პირველადი რეიტინგის სიდიდე) გაზრდით.

აღნიშნულ მექანიზმს ჩვენ *პირდაპირი უერთიერთქმედების აღდგენას* ვუწოდებთ, რადგან იგი მხოლოდ მაშინ არის აქტიური, როდესაც კვანძი უშუალოდ ურთიერთქმედებს ანუ კვანძს მხოლოდ მაშინ შეუძლია არასათანადო ქცევის მდგომარეობიდან აღდგენა, როდესაც იგი სხვა კვანძის სიახლოვეს იმყოფება. აღნიშნულის აზრს ის წარმოადგენს, რომ CPM შეტყობინებების რაოდენობა, რომლებიც სათანადო ქცევას ადგენენ, გაცილებით დიდია, ვიდრე რაოდენობა CPM შეტყობინებებისა, რომლებსაც არასათანადო ქცევის აღმოჩენამდე მივყავართ და, შესაბამისად, ჩვენ აღდგენის შეზღუდვა დაგვჭირდა. საწინააღმდეგო შემთხვევისას არასათანადო ქცევის კვანძები ზედმეტად სწრაფად აღდგებოდნენ და სათანადოდ არ დაისჯებოდნენ.

თუმცა უშუალო ურთიერთქმედებაზე დაფუძნებული ეს მიდგომა შესაძლოა ყველა სახის პრობლემას არ მიესადაგოს თუ ორი კვანძი არ მოძრაობს და ბრალს სდებს ერთმანეთს, ისინი ვერასოდეს შეძლებენ რეპუტაციის აღდგენას, თუ ერთმანეთის არეალში არ მოხვდებიან.

ყოველივე ზემოთქმულიდან გამომდინარე შეგვიძლია ავღნიშნოთ, რომ ჩვენს მიერ უსაფრთხოების მოდიფიცირებული ალგორითმი წარმოადგენს მექანიზმს, რომელიც TC შეტყობინებებში შენახულ მარშრუტებს ეყრდნობა. როდესაც TC

კონკრეტულ კვანძს მიაღწევს, CPM შეტყობინება უკან იგზავნება იმ კოეფიციენტის შესაბამისად, რომელიც განსაზღვრულია TC შეტყობინებაში აკუმულირებული მარშრუტის გადასატანად საწყის კვანძამდე (ალგორითმი, სურ. 4.1.3 ბლოკი 8). აღნიშნული ინფორმაცია შემდეგ გამოიყენება არასათანადო ქცევის კვანძების განსაზღვრისათვის, როგორც ეს ადრე იყო აღწერილი.

უნდა აღინიშნოს, რომ თუ სათანადოდ დაცული არ იქნება, მარშრუტის ინფორმაცია შესაძლოა გამოყენებულ იქნას შავი სიის თავდასხმების განსახორციელებლად, სადაც უფლებამოსილ კვანძებს ბრალი ედებათ არასათანადო ქცევაში. მოცემული ინფორმაციის ერთიანობისა და ჭეშმარიტობის უზრუნველსაყოფად, შეგვიძლია შემდეგი ღონისძიებები გავატაროთ;

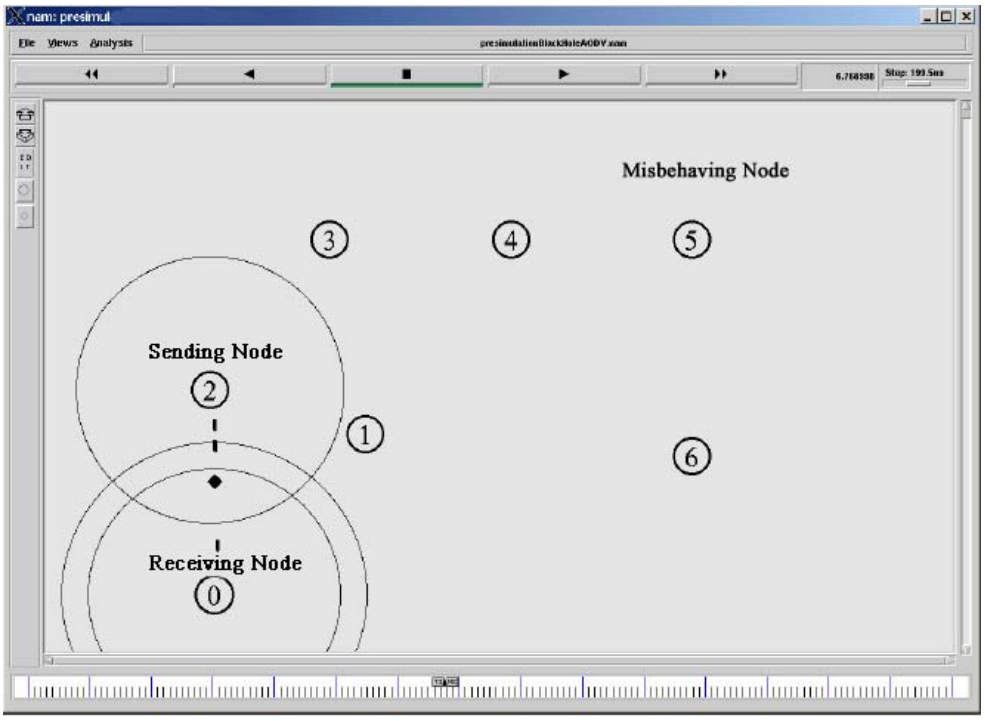
- *აუთენტიფიკაციის შემოწმება,* სადაც მარშრუტის თითოეული კვანძი ამოწმებს წინამორბედი კვანძის მიერ მოწოდებული ხელმოწერის ინფორმაციას, რათა განისაზღვროს, ჩართო თუ არა მან საკუთარი თავი TC-ში;
- *მარშრუტის გაყალბებისგან დაცვა.* ქსელში გავრცელებულ შეტყობინებებში შენახული მარშრუტების ერთიანობის დაცვა. სქემათა აღნიშნული სახეობები ფართოდ იქნა შესწავლილი მოთხოვნამდე მარშრუტიზაციის პროტოკოლების ფარგლებში და ეს სქემები ახორციელებს დაცვას ისეთი მარშრუტების გაყალბებისგან, რომლებიც [58] უკვე არსებობს;

გარდა ამისა, თუ ქსელის მარშრუტების მრავალფეროვნება დაბალია, შესაძლოა CPM შეტყობინების უკან გაგზავნა მოხდეს მარშრუტის გავლით, რომელიც თავად არასათანადო ქცევის კვანძს შეიცავს. აღნიშნული საშუალებას მისცემს არასათანადო ქცევის კვანძს მოიშოროს პაკეტი და მოცემული ინფორმაცია დაიკარგოს და, ამდენად, შედეგად სასჯელი შემცირდება. აღნიშნული საკითხის გვერდის ავლის ერთ-ერთი გზა შემდეგია:

1) შეიცვალოს OLSR-ს სიჭარბის პატამეტრი (TC_REDUNDANCY) იმგვარად, რომ იმ კვანძმა მოახდინოს მეტი მეზობლის გაცხადება, ვიდრე მხოლოდ TC შეტყობინების MPR შემრჩევთა ერთობლიობისა.

4.6.4. უსაფრთხოების მოდიფიცირებული ალგორითმის მოდელირება და მისი შედეგების განხილვა

მოცემულ ნაშრომში აღწერილი მოდელირება განხორციელდა ქსელის სიმულატორის ns2 ვერსია 2.29.2- გამოყენებით. OLSR პროტოკოლისთვის ყველა ნაგულისხმევი სიდიდეა გამოყენებული RFC3626 სპეციფიკაციიდან. მოდელირება ჩატარდა 30 კვანძისთვის, სადაც გადაცემის მანძილი იყო 250 მეტრი, 800 წამის განმავლობისას ფართობზე 2000X400 მეტრზე. გამოყენებულ იქნა შემთხვევითი კოორდინატების მობილურობის მოდელი (Random Waypoint Mobility model). შედეგების გასაშუალოებისა და სასურველი ან არასასურველი სცენარის არჩევის შესაძლებლობის შესამცირებლად გაშვებულ იქნა ხუთი დამოუკიდებელი რეპლიკაცია, თითოეული მობილურობის 10 ცალკეული სცენარით, რამაც შედეგად მოგვცა ჯამური 50 მოდელი, გაშვებული პარამეტრთა თითოეული შესაფასებელი ერთობლიობისთვის. ქსელის შემოწმებისთვის მოძრავი მობილური კვანძებით, ვითვალისწინებთ კვანძებს, რომელთა სიჩქარეა 1.4მ/წ და 2.4მ/წ. გარდა ამისა, 1 და 5-წამიანი პაუზები ასევე იქნა ტესტირებული.



სურ. 4.4 ექსპერიმენტში მონაწილე კვანძების ფრაგმენტი

ა. თავდასხმა

თავდამსხმელი ორი სახის თავდასხმას ახორციელებს: ყალბი HELLO და ყალბი TC შეტყობინებების გენერირება.

ყალბი HELLO შეტყობინებების გენერირებისთვის თავდამსხმელი კვანძი ამატებს ყალბ ინფორმაციას იმის შესახებ, რომ მას შეუძლია მისი ყველა ორბიჯიანი მეზობლის მიღწევა, რის განზრახვას მისი MPR-ად არჩევის იძულება წარმოადგენს. მოცემული თავდასხმა შესაძლოა ზიანის მომტანი იყოს ორი გზით: (ა) მან შესაძლოა გამოიწვიოს მცდარი MPR ერთობლიობის შერჩევა და (ბ) თავდასხმული კვანძის მიერ გაგზავნილმა შეტყობინებებმა შესაძლოა ვერ მიაღწიონ მის ზოგ ორბიჯიან მეზობელს.

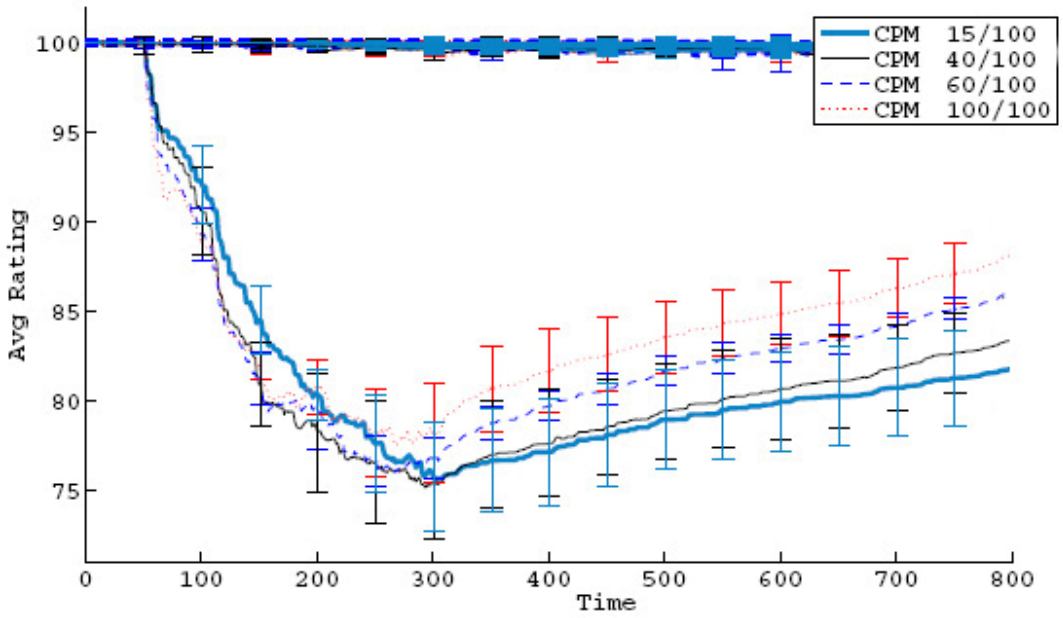
ყალბი TC შეტყობინებების გენერირებისთვის თავდამსხმელი კვანძი შემთხვევით ირჩევს კვანძს, რომელიც სამი ან მეტი ბიჯით არის დაშორებული მისგან და აცხადებს მასთან პირდაპირ კავშირს. მოცემული თავდასხმა შესაძლოა ზიანის მომტანი იყოს, რადგან მას შემოაქვს კონფლიქტური მარშრუტები და ხელს უწყობს კავშირის დაკარგვასა და ქსელის მარშრუტების დაგრძელებას.

ორივე სახის თავდასხმა და აღმოჩენის შესაბამისი მექანიზმები დამოუკიდებლად იქნა ტესტირებული. ჩვენი მოდელირებისას თავდამსხმელი კონტროლის ყალბი ტრაფიკის გენერირებას იწყებს მოდელირების დაწყებიდან 50 წამის შემდეგ, ხოლო სათანადოდ ქცევას 300 წამის შემდეგ იწყებს.

ბ. უსაფრთხოების მოდიფიცირებული ალგორითმის პარამეტრები

რამდენადაც უსაფრთხოების მოდიფიცირებული ალგორითმის მიზანს მარშრუტიზაციის კონტროლის ყალბი გენერირების დასჯა წარმოადგენს, როგორც დამოუკიდებლად, ისე იმ შემთხვევაშიც, როდესა კვანძი უარს ამბობს ტრაფიკის გადაცემაზე, ტრაფიკის გადაცემის უარყოფასთან დაკავშირებული პარამეტრები უნდა განისაზღვრის სიდიდით SRV = 1 (მეორადი რეიტინგის აღდგენის სიდიდე), ***n*** (მეორადი რეიტინგის ზრდა) ან ***m*** (მეორადი რეიტინგის შემცირება), აღნიშნული შედეგად გვაძლევს, როგორც ეს მოსალოდნელი იყო, ძალიან მაღალ მეორად რეიტინგებს, რადგან ტრაფიკის უარყოფა გამოყენებული არ იყო.

დანარჩენი პარამეტრებისთვის α პირველადი და მეორადი β სიდიდეები განისაზღვრა, როგორც მაქსიმალური სიდიდე 100. სხვა სიტყვებით რომ ვთქვათ, ჩვენ ვუშვებთ, რომ ქსელის კვანძები პატიოსანია. ვინაიდან CPM შეტყობინების მაჩვენებლისთვის λ რთულია ადექვატური სიდიდის არჩევა, ჩვენ განვახორციელეთ რიგი მოდელირებები და გავაანალიზეთ CPM სხვადასხვა შეტყობინებათა მაჩვენებელი.



სურ. 4.5. კვანძების საშუალო რეიტინგი (ყალბი HELLO, 1.4 მ/წ)

სასჯელის სიდიდე PV და პირველადი რეიტინგის აღდგენის სიდიდე PRV კორელაციურად უნდა განისაზღვროს, რათა საშუალება გვქონდეს არასათანადო ქცევის კვანძების სათანადოდ დასჯისა და იმ კვანძების დასაბუთებული აღდგენის საშუალება, რომლებმაც სათანადოდ ქცევა დაიწყეს არასათანადო ქცევის შემდეგ. ჩვენი მოდელირება უჩვენებს, რომ ყალბი დადანაშაულებები უფრო ხშირია ყალბი HELLO შეტყობინებების დადგენისას, ვიდრე ყალბი TC შეტყობინებების დადგენისას. შესაბამისად, ჩვენ გამოვიყენეთ უფრო მკაცრად დასჯის სიდიდე PV=0 ყალბი TC შეტყობინებების აღმოჩენისთვის და ნაკლებად მკაცრი PV=*პირველადი რეიტინგი/2* – ყალბი HELLO შეტყობინებების აღმოჩენისთვის. პირველადი აღდგენის სიდიდესთან დაკავშირებით სიდიდემ PRV=1 ფრიად დამაკმაყოფილებელი შედეგები მოგვცა დასჯის თვალსაზრისით, კვანძების აღდგენის საპირისპიროდ. როგორც PV-ს, ისე PRV-სთვის, მათთვის მაღალი სიდიდეების განსაზღვრა გვაძლევს უკეთესი აღდგენის, მაგრამ უარესი სასჯელის საშუალებას და პირიქით.

გ. მიღებული შედეგები

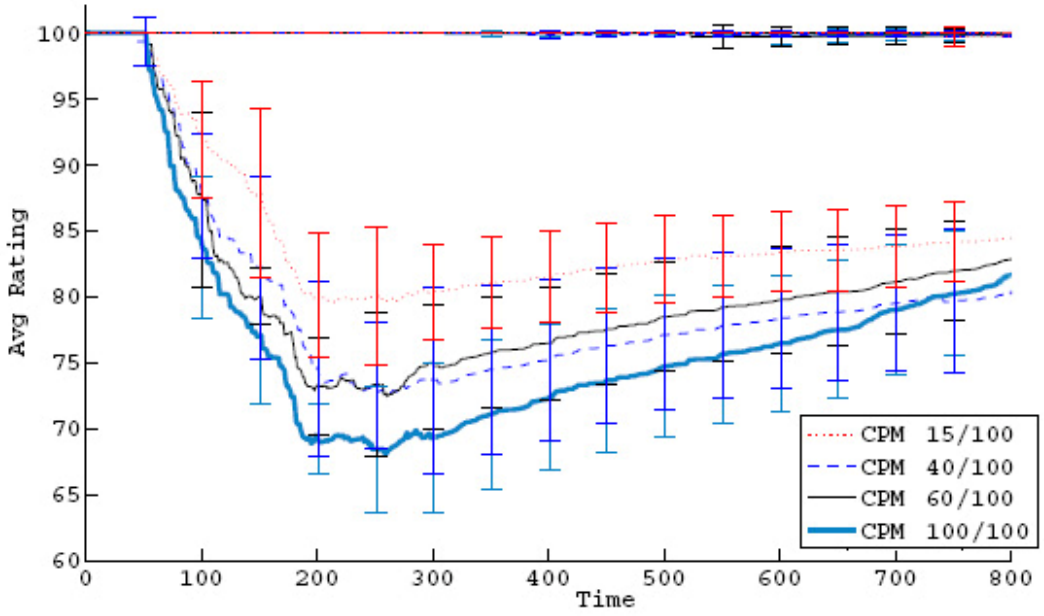
წინამდებარე თავში ჩვენ განვიხილავთ რიგ მოდელირების შედეგებს, სადაც ხაზგასმულია უსაფრთხოების ჩვენს მიერ დამუშავებული სქემის ეფექტურობა და ხარჯები ჭარბი ტრაფიკის თვალსაზრისით. ნაჩვენებია ორი სახის გრაფიკი:

გრაფიკი კვანძთა საშუალო რეიტინგით და გრაფიკი ჭარბი ხარჯებით, გამოწვეული CPM შეტყობინებებითა და OLSR ოპერაციებით.

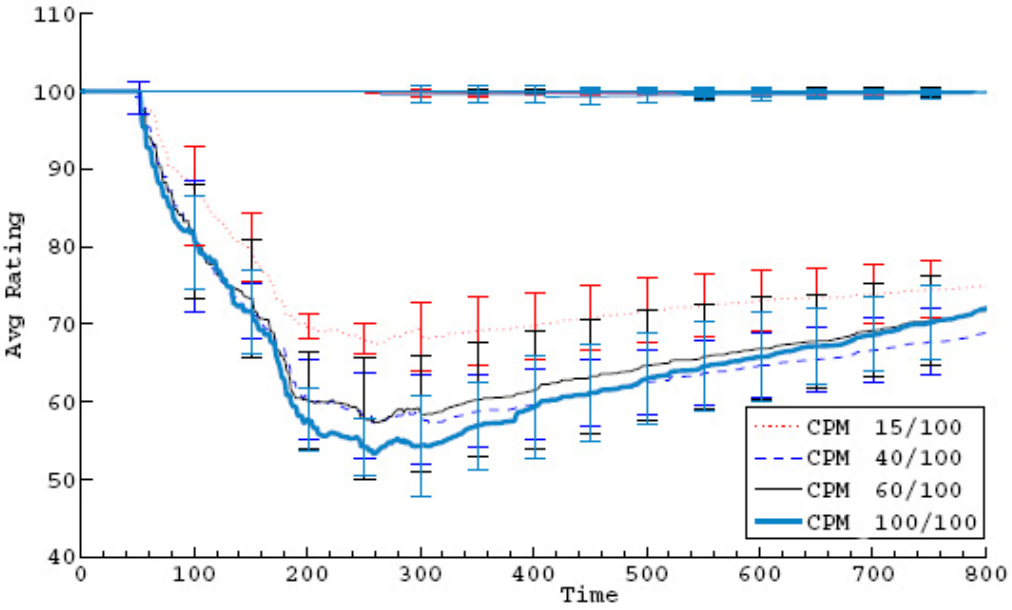
გრაფიკები კვანძთა საშუალო რეიტინგებით გვიჩვენებენ ქსელის ყველა კვანძის რეიტინგს. ზედა სხივები შეესაბამება ყველა სათანადო ქცევის კვანძის საშუალო რეიტინგს, შუა სხივები შეესაბამება არასათანადო ქცევის კვანძების საშუალო რეიტინგს CPM ყველა გათვალისწინებული მაჩვენებლისთვის. კონკრეტული A კვანძის საშუალო რეიტინგი R გვეუბნება, რომ თუ ტრაფიკი ქსელში თანაბრად არის განაწილებული, დასჯის მექანიზმი საშუალებას მოგვცემს, რომ A-ში შექმნილი ტრაფიკის საშუალოდ R% იქნას გადატანილი შემდეგ დანიშნულებამდე.

სიჭარბის გრაფიკები ძირითადად საშუალებას გვაძლევს შედარდეს ნამატი ხარჯები CPM მექანიზმისა, რომელიც შემოთავაზებულია უსაფრთხოების ჩვენს მიერ დამუშავებული სქემით და ნამატი ხარჯები ჩვეული OLSR ოპერაციისა.

მეოთხე ნახატის გრაფიკზე ვხედავთ, რომ ყალბი HELLO შეტყობინებების აღმოჩენის მექანიზმი მნიშვნელოვნად არ იცვლება CPM შეტყობინებათა მაჩვენებლის ცვლასთან ერთად. იცვლება აღდგენის მექანიზმი, რაც უფრო სწრაფია CPM შეტყობინების უფრო მაღალი სიდიდისას.



სურ. 4.6. კვანძთა საშუალო რეიტინგები (ყალბი TC, 1.4 მ/წ, 1 ყალბი ლინკი)



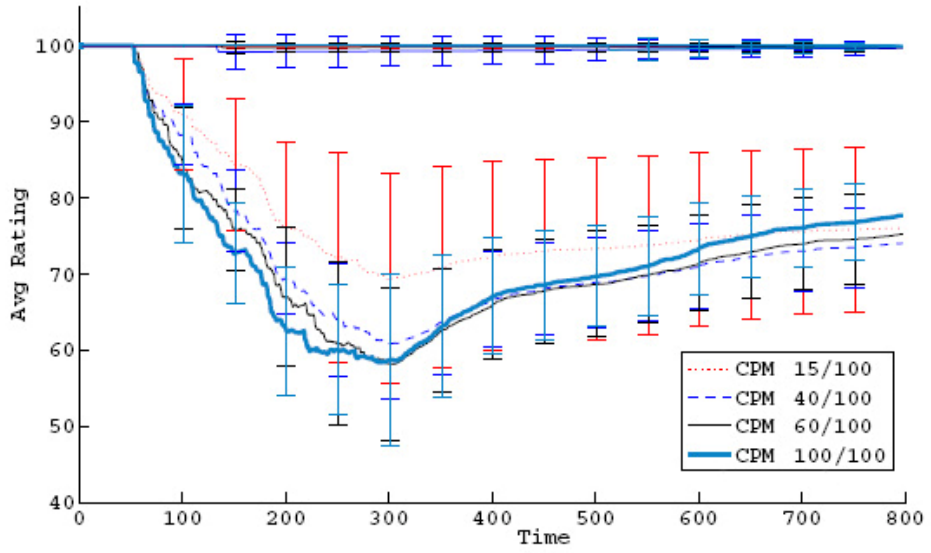
სურ. 4.7. კვანძთა საშუალო რეიტინგები (ყალბი TC, 1.4 მ/წ, 4 ყალბი ლინკი)

ყალბი TC შეტყობინების აღმოჩენასთან დაკავშირებით (ნახ. 5) ვხედავთ, რომ მოცემული მექანიზმი მეტად ექვემდებარება ცვლილებებს CPM შეტყობინების მაჩვენებელთან მიმართებით. კვანძის ორივე ტესტირებული სიჩქარისთვის არასათანადო ქცევის კვანძის საშუალო რეიტინგი უფრო სწრაფად ეცემა და უფრო დაბალ სიდიდემდე CPM შეტყობინების უფრო მაღალი მაჩვენებლის შემთხვევაში და კვლავ, აღდგენის მექანიზმი უფრო სწრაფია CPM შეტყობინებათა მაღალი მაჩვენებლისთვის. შედეგები 2.4 მ/წ სიჩქარისთვის გამოტოვებულია, რადგან განხილულის მსგავსია.

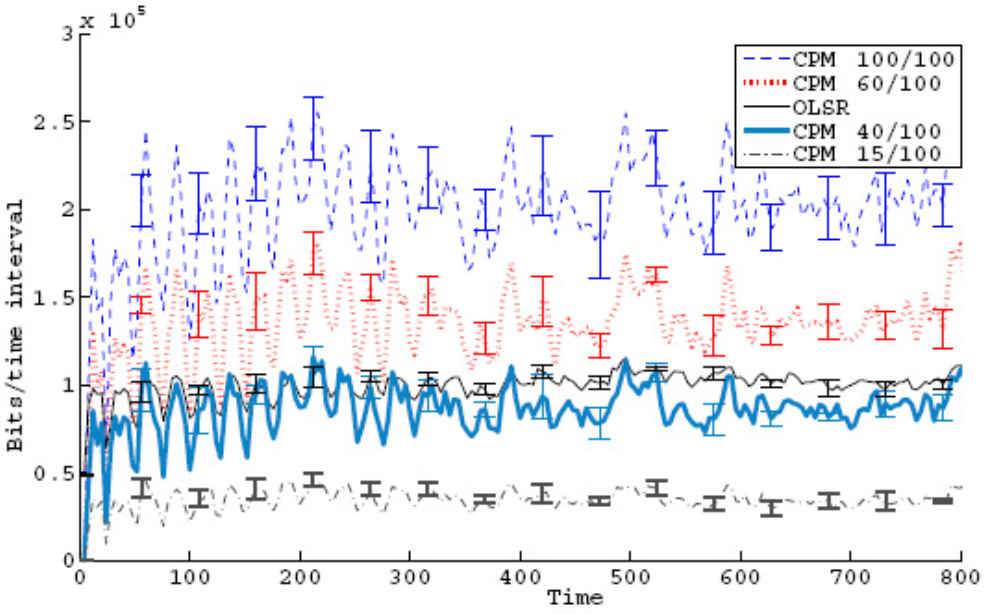
შესაძლოა მოგვეჩვენოს, რომ მოცემული რეიტინგები უფრო მკაცრი უნდა იყოს, თუმცა მნიშვნელოვანია შევნიშნოთ, რომ წარმოდგენილი საშუალო რეიტინგები ითვალისწინებს მთელ ქსელს. შესაბამისად, საბოლოოდ ჩართულია ის კვანძებიც, რომლებთანაც არასათანადო ქცევის კვანძი არ ურთიერთქმედებს (მაგალითად, რადგან ისინი არ იქცევიან MPR-დ და, შესაბამისად, არ გადასცემენ ტრეფიკს, რაც არასათანადო ქცევის კვანძებისთვის მაღალი რეიტინგის შენარჩუნებას ნიშნავს). გარდა ამისა, ყალბი TC შეტყობინებების დადგენისთვის ტესტები ჩატარდა თავდამსხმელის გათვალისწინებით, რომელიც ცალკეულ ყალბ ლინკს აცხადებს. ყალბი ლინკების რაოდენობის ზრდასთან ერთად საშუალო პირველადი რეიტინგები შემდგომშიც ეცემა. მაგალითისთვის იხილეთ ნახ. 6, სადაც 4 ყალბი ლინკით პირველადი რეიტინგი უფრო დაბალ სიდიდემდე ეცემა, ვიდრე

წინამორბედ გრაფიკებზე და აღწევს მინიმალურ 55 ქულას, როდესაც CPM შეტყობინების მაჩვენებელი 100%-ია.

აქამდე ნაჩვენებ ყველა გრაფიკზე საშუალო პაუზის სიდიდედ 1 წამი იყო აღებული. მეშვიდე ნახატის შედეგების მიხედვით, სადაც საშუალო პაუზად 5 წამია აღებული, შეგვიძლია დავინახოთ, რომ (1) არასათანადო ქცევის კვანძი უფრო მკაცრად ისჯება და (2) სათანადო ქცევის კვანძებს ოდნავ უარესი საშუალო რეიტინგი აქვთ. აღნიშნული შეესაბამება ფაქტს, რომ უფრო ხანგრძლივი პაუზებით კვანძები, ჩვეულებრივ, ნაკლებად ურთიერთქმედებენ ერთმანეთთან და, შესაბამისად, აღდგენის მექანიზმი (რომელიც პირდაპირ ურთიერთქმედებაზეა დაფუძნებული) ნაკლებად ეფექტური იქნება. აღნიშნული ფაქტი შედეგად გვაძლევს მკვეთრ ცვლილებას საშუალო პირველადი რეიტინგის გაზრდაში, როდესაც (300წ) თავდამსხმელი კვანძი წყვეტს არასათანადო ქცევას. აღნიშნული არასათანადო ქცევა ნაკლებად გასაგები იყო წინამორბედ გრაფიკებზე, სადაც 1-წამიანი საშუალო პაუზა იყო გამოყენებული, რადგან კვანძებს შორის ურთიერთქმედებათა დიდი რაოდენობა ზრდის რეიტინგებზე აღდგენის მექანიზმის გავლენას.



სურ. 4.8. კვანძთა საშუალო რეიტინგები (ყალბი TC, 1.4 მ/წ, 1 ყალბი ლინკი, საშუალო პაუზა 5წ).



სურ. 4.9. CPM მექანიზმის ჭარბი ხარჯები OLSR-ს საპირისპიროდ (1.4 მ/წ).

მერვე ნახატზე წარმოდგენილი ჭარბი ხარჯების შედეგების კუთხით, როგორც ეს მოსალოდნელი იყო, უსაფრთხოების ჩვენს მიერ დამუშავებული სქემის მაღალი ნამატი დანახარჯებია მიღებული, როდესაც გამოიყენება CPM შეტყობინებათა მაჩვენებელი 100% და, ბუნებრივად, CPM შეტყობინებათა მაჩვენებლის შემცირებასთან ერთად ნამატი ხარჯებიც მცირდება და ძალიან დაბალ სიდიდეს აღწევს, როდესაც CPM შეტყობინების მაჩვენებელი 15%-ია. შედეგები კვანძის სიხშირისთვის 2.4 მ/წ გამოტოვებულია, რადგან მოცემულთან ძალიან ახლოს არის.

საბოლოოდ, რადგან ყალბი HELLO შეტყობინებების აღმოჩენის მექანიზმი, სავარაუდოდ, უკეთ მუშაობს გათვალისწინებული მაჩვენებლებისთვის, როგორც ექსპერიმენტებიდან ჩანს, CPM შეტყობინების მაჩვენებელი 15% ყველაზე სწორი გადაწყვეტილება იქნება. ყალბი TC შეტყობინების აღმოჩენის კუთხით, CPM შეტყობინებების მაჩვენებელი 15-დან 40%-მდე დასაბუთებულ სასჯელს უზრუნველყოფს არასათანადო ქცევის კვანძებისთვის, რომელთა რეიტინგი 75-80-ია ყველაზე სუსტი თავდამსხმელისთვის (ერთი ყალბი ლინკი), როდესაც საკმაოდ დაბალი იქნება ნამატი ხარჯები ქსელის ტრაფიკის თვალსაზრისით.

4.6.6. დასკვნები

ჩვენ წარმოვადგინეთ უსაფრთხოების მოდიფიცირებული ალგორითმი OLSR პროტოკოლის უსაფრთხოებისთვის. მექანიზმს საქმე აქვს ყალბი HELLO და ყალბი TC შეტყობინებების გენერირებასთან – ორ თავდასხმასთან, რომელთათვის დღემდე დამაკმაყოფილებელი გადაწყვეტილება არ არსებობს. გარდა უსაფრთხოების მოცემული პრობლემების ბუნებრივი გადაწყვეტილების უზრუნველყოფისა, ჩვენი პრაქტიკული სქემა მედეგია რეიტინგების სისტემის ზოგადი პრობლემებისადმი. კონკრეტულად, ჩვენი ალგორითმი არ იძლევა საშუალებას ქსელში მოხდეს რეიტინგების ინფორმაციის განფანტვა რაც შედეგად იძლევა კვანძების მცდარად დადანაშაულებას ან შექებას – შემდეგ ისინი იწყებენ CPM ყალბი შეტყობინებების გენერირებას (რისგან დაცვა კრიპტოგრაფიული მექანიზმებით შეიძლება) ან CPM ძველი შეტყობინებების განმეორებას (რისგან დაცვას დროითი ნიშნულის მექანიზმი ახორციელებს).

ჩვენს მიერ დამუშავებულ უსაფრთხოების მოდიფიცირებულ ალგირითმს შემდეგი მახასიათებლები გააჩნია:

- ის ეყრდნობა რეპუტაციის კონცეფციას, იყენებს რეიტინგების ცნებას და უშუალო დაკვირვებას;
- იგი უზრუნველყოფს ახალი და სანდო მონიტორინგის მექანიზმს, დაფუძნებულს CPM შეტყობინებებზე, რომელიც, ოდნავ გაზრდილი სიხშირის ხარჯზე, აღმოფხვრის "მეთვალყურის" კონცეფციის ნაკლოვანებებს;
- მას უნარი აქვს აღმოაჩინოს და დასაჯოს მარშრუტიზაციის კონტროლის ყალბი ტრაფიკის გენერირება (ლინკის იმიტირების თავდასხმა), თუმცა მას ახასიათებს მდგრადობა დანარჩენი თავდასხმებისაგანაც;
- იგი შეიცავს არასათანადო ქცევის კვანძების საყოველთაო აღმოჩენის მექანიზმს საგანგაშო სიგნალების გავრცელების გარეშე, რომლებიც შესაძლოა გამოყენებულ იქნან შავი სიის თავდასხმებისთვის, სადაც უფლებამოსილ კვანძებს ბრალს სდებენ არასათანადო ქცევაში;

- იგი იცავს შავი სიის თავდასხმებისგან, რაც წარმოადგენს ყალბი CPM შეტყობინებების გენერირების შედეგს იმავე სახის მექანიზმების გამოყენებით;
- მოდიფიცირებულ OLSR პროტოკოლს ემატება მხოლოდ ორი ელემენტი: CPM შეტყობინება და რეიტინგების ცხრილი;
- ამოწმებს HELLO და TC ყალბ შეტყობინებებს;
- აღნიშნული HELLO და TC შეტყობინებების შემოწმება წარმოებს CPM-ში ორი, სამი და მეტი ბიჯით დაცილებული კვანძების შემოწმებით, რაც უფრო სანდოს ხდის თავად შემოწმებას და თავიდან გვაცილებს გადაადგილების შეცდომების გენერირებას;
- გააჩნია კვანძების რეიტინგების საწყისი მნიშვნელობების მინიჭების მოქნილი მექანიზმი, რომლის თანახმად შესაძლებელია კვანძები ამუშავდეს განსხვავებული აქტივობით;
- შესაძლებელია იმის კონტროლი, თუ როგორ ხდება ყალბი მდგომარეობიდან კვანძის აღდგენა;
- გააჩნია მოქნილი დასჯისა და დავიწყების მექანიზმები, რომლებიც უზრუნველყოფენ კვანძების უფრო ეფექტურ მუშაობას;
- შესაძლებელია CPM შეტობინებების სიხშირის მართვა.

გამოყენებული ლიტერატურა

[1] A. Mishra, *Security and Quality of Service in Add Hoc Wireless Networks*, Cambridge University Press, 2008
[2] The Official Wi-Fi Technology Info site: www.wi-fi.org
[3] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic (Editors), *Mobile Ad Hoc Networking*, John Wiley and Sons, 2004.
[4] The Official Bluetooth Technology Info site: www.bluetooth.com
[5] The Official WiMAX Technology Info site: http://www.wimaxforum.org/home/
[6] C. Eklund, R. B. Marks, L. Kenneth, "IEEE Standard 802.16: A Technical Overview of the WirelessMAN™ Air Interface for Broadband Wireless Access", *IEEE Communications Magazine*, pp. 98-107, June 2002.
[7] F. Anjum and P. Mouchtaris, *Security for wireless add-hoc networks*, John Wiley & Sons, Inc, 2007.
[8] S. Axelsson, Intrusion Detection Systems: "A Taxomomy and Survey, Technical report" no. 99–15, Dept. Computer Engineering, Chalmers University of Technology, Sweden, Mar. 2000.
[9] Y. Zhang and W. Lee, ''Intrusion detection in wireless ad-hoc networks,'' *6th International Conference on Mobile Computing and Networking (MOBICOM'00)*, Aug. pp. 275–283, 2000.
[10] M. Bishop, *Computer Security: Art and Science*, Addison Wesley, Nov 2002.
[11] W. Stallings, *Cryptography and Network Security Principles and Practices*, Fourth Edition, Prentice Hall, Nov 2005
[12] A. Sabir, S. Murphy, and Y. Yang, "Generic Threats to Routing Protocols," *draft-ietf-rpsec-routing-threats-07*, October 2004.
[13] ITU-T Recommendation X.800, Security Architecture for OSI for CCITT applications, 1991.[14] M. Bishop, *Computer Security: Art and Science*. Boston: Addison-Wesley, 2003.[15] M. Bishop, *Introduction to Computer Security*. Boston: Addison-Wesley, 2005.[16] J.Pieprzyk, T. Hardjono, and J.Seberry, *Fundamentals of Computer Security*. New York: Springer-Verlag, 2003.
[17] K. Wu and J. Harms, ''QoS support in mobile ad hoc networks,'' *Crossing Boundaries*, vol. 1, no. 1, pp. 92–106., Fall 2001.
[18] J. Newsome, E. Shi, D. Song, and A. Perrig, ''The Sybil attack in sensor networks: analysis and defenses,'' *3rd Int. Symposium on Information Processing in Sensor Networks*, pp. 171–179, 2003.
[19] G. Simmons, *Contemporary Cryptology: The Science of Information Integrity. Piscataway*, NJ: IEEE Press, 1992.
[20] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, October 1996.
[21] H. Feistel, "Cryptography and Computer Privacy." *Scientific American*, Vol. 228, No 5, pp. 15-23, May 1973.
[22] X. Wang, Y. L. Yin, and H. Yu, "Finding Collisions in the Full SHA-1", CRYPTO, 2005.
[23] X. Wang, Y. Yin, and H. Yu, "Finding Collisions in the Full SHA-1", *Proceedings, Crypto '05*, published by Springer-Verlag, 2005.
[24] M. Bellare; R. Canetti, and H. Krawczyk, "Keying Hash Functions for Message Authentication." *Proceedings, CRYPTO '96*, published by Springer-Verlag, August 1996

[25] W. Du, J. Deng, Y. S. Han, S. Chen, and P. K. Varshney, “A Key Management Scheme for Wireless Sensor Networks using Deployment Knowledge,” in *INFOCOM*, 2004.
[26] W. Du, J. Deng, Y. Han, and P. Varshney, “A Pairwise Key Predistribution Scheme for Wireless Sensor Networks,” in *Proceedings of the Tenth ACM Conference on Computer and Communications Security (CCS 2003),* pp. 42–51, October 2003.
[27] W. Du, J. Deng, Y. S. Han, P. Varshney, J. Katz, and A. Khalili, “A Pairwise Key Predistribution Scheme for Wireless Sensor Networks,” *ACM Transactions on Information and System Security (TISSEC)*, pp. 228–258, 2005.
[28] N. Asokan and P. Ginzboorg, ‘‘Key agreement in ad hoc networks,’’ *Computer Communications*, vol. 23, 2000, pp. 1627–1637.
[29] J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, ‘‘Providing robust and ubiquitous security support for mobile ad-hoc networks,’’ *9th IEEE International Conference on Network Protocols*, Riverside, CA, Nov. 2001, pp. 251–260.
[30] L. Zhou and Z. Haas, “Securing Ad Hoc Networks,” IEEE Network, 13(6), 24–30 (1999).
[31] W. Diffie, and M. Hellman, "Multiuser Cryptographic Techniques." IEEE Transactions on Information Theory, November 1976.
[32] M. Cagalj, S. Capkun, and J. P. Hubaux, “Key Agreement in Peer-to-Peer WirelessNetworks,” in Proceedings of the IEEE (Special Issue on Cryptography and Security), pp. 467–478, Feb. 2006.
[33] C. Ellison and S. Dohrmann, “Public-key Support for Group Collaboration,” ACM Transactions on Information Systems Security, 6(4), pp.547–565, 2003.
[34] C. Gehrmann, C. Mitchell, and K. Nyberg, “Manual Authentication for Wireless Devices,” *RSA Cryptobytes*, 7(1), pp.29–37, 2004.
[35] P. Papadimitratos and Z. J. Haas, “Secure Link State Routing for Mobile Ad Hoc Networks,” in Proceedings of the 2003 Symposium on Applications and the Internet Workshops (SAINT’03 Workshops), 2003, p. 379.
[36] R. Perlman, *Interconnections: Bridges and Routers*. Addison-Wesley, Reading, MA, 1993.
[37] T. Clausen and P. Jacquet, eds, “Optimized Link State Routing Protocol (OLSR),” IETF RFC 3626, October 2003.
[38] P. Jacquet, P. M¨uhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. “Optimized Link State Routing protocol for ad hoc networks”. I*n Proceedings of the IEEE International Multitopic Conference (INMIC 2001)*, Pakistan, 2001.
[39] Y. Hu, A. Perrig, and D. Johnson, “Packet Leashes: a Defense Against Wormhole Attacks in Wireless Ad Hoc Networks,” *in Proceedings of INFOCOM, San Francisco*, CA, pp. 1976–1986, April 2003.
[40] L. Hu and D. Evans, “Using Directional Antennas to Prevent Wormhole Attacks,” *Network and Distributed System Security Symposium*, San Diego, CA, 5–6 Feb. 2004.
[41] R. Poovendran and L. Lazos, “A Graph Theoretic Framework for Preventing the Wormhole Attack in Wireless Ad Hoc Networks,” *ACM Journal on Wireless Networks (WINET)*, 2006.
[42] J. Douceur, “The Sybil Attack,” *in Proceedings of IPTPS 2002*, Cambridge, MA, pp. 251–260, March 2002.
[43] C. Karlof and D. Wagner, “Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures,” in *Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications*, May 2002.

[44] A. Halfslund, A. Tonnesen, R. B. Rotvik, J. Andersson, and O. Kure, “Secure Extension to the OLSR Protocol,” *OLSR Interop and Workshop*, 2004.
[45] D. Raffo, T. Clausen, C. Adjih, and P. Muhlethaler, “An Advanced Signature System for OLSR,” *SASN’04*, October 2004.
[46] C. Adjih, T. Clausen, P. Jacquet, A. Laouiti, P. Muhlethaler, and D. Raffo, “Securing the OLSR Protocol,” *Proceedings of Med-Hoc-Net*, June 2003.
[47] C. Adjih, T. Clausen, P. Jacquet, A. Laouiti, P. M¨uhlethaler, and D. Raffo, “Securing the OLSR protocol,” in *Proceedings of Med-Hoc-Net*, Mahdia, Tunisia, June 2003.
[48] C. Adjih, D. Raffo, and P. M¨ulethaler, “Attacks against OLSR: Distributed key management for security,” in *2005 OLSR Interop and Workshop*, Ecole Polytechnique, Palaiseau, France, July 28–29 2005.
[49] C. Adjih, T. Clausen, A. Laouiti, P. M¨uhlethaler, and D. Raffo, “Securing the OLSR routing protocol with or without compromised nodes in the network,” HIPERCOM Project, INRIA Rocquencourt, Tech. Rep. INRIA RR-5494, February 2005.
[50] D. Raffo, C. Adjih, T. Clausen, and P. M¨uhlethaler, “An advanced signature system for OLSR,” in *SASN ’04: Proceedings of the 2nd ACM Workshop on security of ad hoc and sensor networks*. New York, NY, USA: ACM Press, 2004, pp. 10–16.
[51] D. Dhillon, T. S. Randhawa, M. Wang, and L. Lamont, “Implementing a fully distributed Certificate Autorithy in an OLSR MANET,” in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2004)*, Atlanta, Georgia, USA, March 21–25 2004.
[52] L. Butty´an and J.-P. Hubaux, “Enforcing service availability in mobile ad-hoc wans,” in *MobiHoc ’00: Proceedings of the 1st ACM international symposium on mobile ad hoc networking & computing*. Piscataway, NJ, USA: IEEE Press, 2000.
[53] S. Zhong, J. Chen, and Y. R. Yang, “Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks,” in *INFOCOM*, 2003.
[54] L. Butty´an and J.-P. Hubaux, “Stimulating cooperation in self-organizing mobile ad hoc networks,” *Mobile Networks and Applications*.
[55] S. Marti, T. J. Giuli, K. Lai, and M. Baker, “Mitigating routing misbehavior in mobile ad hoc networks,” in *MobiCom ’00: Proceedings of the 6th annual international conference on mobile computing and networking*. New York, NY, USA: ACM Press, 2000.
[56] S. Buchegger and J.-Y. L. Boudec, “Performance analysis of the confidant protocol,” in *MobiHoc ’02: Proceedings of the 3rd ACM international symposium on mobile ad hoc networking & computing*. New York, NY, USA: ACM Press, 2002.
[57] P. Michiardi and R. Molva, “Core: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks,” in *Proc. Of the IFIP-Communication and Multimedia Security Conference*, Copenhagen, June 2002.
[58] Y. Hu, A. Perrig, and D. Johnson, “Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks,” *Wireless Networks*, 2005.
[59] F. J. Ros, “UM-OLSR,” obtain via: http://masimum.dif.um.es/.
[60] S. PalChaudhuri, J.-Y. L. Boudec, and M. Vojnovic, “Perfect simulations for random trip mobility models,” in *Annual Simulation Symposium*. IEEE Computer Society, 2005, [Online]. Available: http://dx.doi.org/10.1109/ANSS.2005.33
[61] Л. Мгебришвили, М. Тевдорадзе. «ЗАЩИТА ИНФОРМАЦИИ В БАНКОВСКИХ СИСТЕМАХ». Энергетика, Телекоммуникации и высшее образование в современных условиях. 3-я международная научно-техническая конференция. Алматы 2002.

[62] Л. Мгебришвили, М. Тевдорадзе. «КЛАССИФИКАЦИЯ ВИРУСОВ». საერთაშორისო კონფერენცია - "ინფორმაციული ტექნოლოგიები - 2008"
[63] Л. Мгебришвили, М. Тевдорадзе. «УПРАВЛЕНИЕ РИСКАМИ В КОМПЬЮТЕРНЫХ СИСТЕМАХ И СЕТЯХ». პერიოდული სამეცნიერო ჟურნალი "ინტელექტი", აგვისტო 2008, №2(31).

დანართი 1.

მოდელირების დროს გამოყენებული კოდი

```
#
# ======================================================================
=
# Initialization
#
# ======================================================================
=
# Default node configuration
set nodeConfig "no-log 0; log-none ; log-route 1"

# Attacker types
set fake_link_hellos 1
set fake_link_tcs    2
set refuser          3

set cpmrate 15
set fake_links 1

if {$argc != 2} {
    puts "Usage: ns final.tcl \[default seed\] \[iteration\]"
    exit
}

set defSeed 1
if {$argc == 2} {
    set defSeed [lindex $argv 0]
    set iteration [lindex $argv 1]
}

if {$defSeed < 1} {
    set defSeed 1
}

puts "Def seed = $defSeed, Iterarion = $iteration"

# (possibly) Remove and create result directory
set dirName
"results_tc/30n_1.4ms_cpmrate_$cpmrate/fakelinks_$fake_links/seed_$de
fSeed/$iteration"
exec sh -c "rm -rf $dirName && mkdir -p $dirName"

#
# ======================================================================
=
# Define options
#
# ======================================================================
=
set opt(chan)           Channel/WirelessChannel     ;# channel type
set opt(prop)           Propagation/TwoRayGround    ;# radio-
propagation model
set opt(netif)          Phy/WirelessPhy             ;# network
interface type
set opt(mac)            Mac/802_11                  ;# MAC type
set opt(ifq)            Queue/DropTail/PriQueue     ;# interface
queue type
```

```
set opt(ll)             LL                          ;# link layer
type
set opt(ant)            Antenna/OmniAntenna         ;# antenna model
set opt(ifqlen)         200                         ;# max packet in
ifq
set opt(adhocRouting)   OLSR                        ;# routing
protocol
set opt(cp)             ""                          ;# connection
pattern file
set opt(sc)             "movement_1.4ms.tcl"        ;# node movement
file.
set opt(traffSessions)  25                          ;# number of
traffic pattern sessions
set opt(seed)           10.0                        ;# seed for
random number gen.
set opt(nn)             30                          ;# number of
mobilenodes

Mac set bandwidth_ 11Mb

set opt(x)              2000                        ;# x coordinate
of topology
set opt(y)              400                         ;# y coordinate
of topology
set opt(stop)           800.0                       ;# time to stop
simulation

# check for random seed
if {$opt(seed) > 0} {
    puts "Seeding Random Number Generator with $defSeed\n"
    global defaultRNG
    $defaultRNG seed $defSeed
}

#
# OLSR global agent configuration
# (commented lines have the default values)
#
Agent/OLSR set use_mac_    false
Agent/OLSR set debug_      false
# Agent/OLSR set debug_      false
# Agent/OLSR set willingness 3
# Agent/OLSR set hello_ival_ 2
# Agent/OLSR set tc_ival_    5

# JP_NEW
# A cpm rate of 50 means that, in average, CPMs are sent in response
to TC in 50% of the cases
Agent/OLSR set cpm_rate_    $cpmrate
Agent/OLSR set def_prating_ 100
Agent/OLSR set def_srating_ 100
Agent/OLSR set srating_dec_  -2
Agent/OLSR set srating_inc_  1
Agent/OLSR set detect_fake_hello_ 0
Agent/OLSR set detect_fake_tc_ 1

# Communication range = 250 meters
Phy/WirelessPhy set RXThresh_ 3.65262e-10

#
==========================================================================
=
```

```
# Create simulator instance
#
=========================================================================
=
set ns_ [new Simulator]

# trace settings
# $ns_ use-newtrace
set tracefd [open $dirName/trace_all.tr w]
set namtrace [open $dirName/animation.nam w]
$ns_ trace-all $tracefd

# initialize a namtrace file for logging node movements to
# be viewed in nam (must be called after mobility is defined)
$ns_ namtrace-all-wireless $namtrace $opt(x) $opt(y)

#
# create topography object
#
set topo [new Topography]

#
# define topology
#
$topo load_flatgrid $opt(x) $opt(y)

#
# create God
# god is used to store an array of the shortest number of
# hops required to reach from one node to another
# e.g. $ns_ at 899.00 "$god setdist 2 3 1"
#
set god [create-god $opt(nn)]

# JP_NEW
# create channel
#
set channel_ [new $opt(chan)]

#
# configure mobile nodes
#
$ns_ node-config -adhocRouting $opt(adhocRouting) \
                 -llType $opt(ll) \
                 -macType $opt(mac) \
                 -ifqType $opt(ifq) \
                 -ifqLen $opt(ifqlen) \
                 -antType $opt(ant) \
                 -propType $opt(prop) \
                 -phyType $opt(netif) \
                 -channel $channel_ \
                 -topoInstance $topo \
                 -agentTrace ON \
                 -wiredRouting OFF \
                 -routerTrace ON \
                 -macTrace OFF \
                 -movementTrace OFF    # mobilenode movement logging
turned ON or OFF
```

```
#
# ============================================================================
# Create & Place nodes
#
# ============================================================================
for {set i 0} {$i < $opt(nn)} {incr i} {
    puts $i
    set node_($i) [$ns_ node]
}

#
# define initial node position in nam
#
for {set i 0} {$i < $opt(nn)} {incr i} {
    $ns_ initial_node_pos $node_($i) 20
}

# -------- MALICIOUS NODES ------------

# Attack type 1 (=) Fake HELLO
# Attack type 2 (=) Fake TC

# $ns_ at 50 "[$node_(5) agent 255] attack-type 1"
# $ns_ at 300 "[$node_(5) agent 255] attack-type 0"
# $node_(5) set willingness 7; # will always (so that he is always
selected as mpr)

# node starts misbehaving at the 50 seconds
$ns_ at 50 "[$node_(5) agent 255] attack-type 2"
# number of links faked by the malicious node (only for attack-type
2)
$ns_ at 50 "[$node_(5) agent 255] max-fake-links $fake_links"
# node stops misbehaving at the 300 seconds
$ns_ at 300 "[$node_(5) agent 255] attack-type 0"

#
# source connection-pattern and node-movement scripts
#
if { $opt(cp) == "" } {    puts "*** NOTE: no connection pattern
specified."
    set opt(cp) "none"

} else {
    puts "Loading connection pattern..."
    source $opt(cp)
}
if { $opt(sc) == "" } {
    puts "*** NOTE: no scenario file specified."
    set opt(sc) "none"
} else {
    puts "Loading scenario file..."
    source $opt(sc)
    puts "Load complete..."
}

#
# Print (in the trace file) routing table and other
# internal data structures on a per-node basis
#
for {set i 0} {$i <= $opt(stop)} {incr i} {
```

```
    $ns_ at $i "[$node_(5) agent 255] print_mprselset"
    for {set j 0} {$j < $opt(nn)} {incr j} {
#     $ns_ at $i "[$node_($j) agent 255] print_nbset"
#     $ns_ at $i "[$node_($j) agent 255] print_nb2hopset"
      $ns_ at $i "[$node_($j) agent 255] print_rating_table"
#     $ns_ at $i "[$node_($j) agent 255] print_mprset"
#     $ns_ at $i "[$node_($j) agent 255] print_mprselset"
#     $ns_ at $i "[$node_($j) agent 255] print_nb2hopset"
#     $ns_ at $i "[$node_($j) agent 255] print_topologyset"
#     $ns_ at $i "[$node_($j) agent 255] print_rtable"
    }
#     $ns_ at $i "[$node_(0) agent 255] print_rtable"
#     # $ns_ at $i "[$node_(0) agent 255] print_linkset"
#     $ns_ at $i "[$node_(0) agent 255] print_nbset"
#     $ns_ at $i "[$node_(0) agent 255] print_nb2hopset"
#     $ns_ at $i "[$node_(1) agent 255] print_mprset"
#     # $ns_ at $i "[$node_(0) agent 255] print_topologyset"
}

#
# tell all nodes when the simulation ends
#
for {set i 0} {$i < $opt(nn) } {incr i} {
    $ns_ at $opt(stop).0 "$node_($i) reset";
}
#---------------------------------------------------------------------
-------
# Finishing procedure
#---------------------------------------------------------------------
-------

proc finishSimulation { } {
    global ns_ tracefd namtrace

    $ns_ flush-trace
    close $tracefd
    close $namtrace

    # Exit
    puts "Finished simulation."
    $ns_ halt
}
#---------------------------------------------------------------------
-------
# Run the simulation
#---------------------------------------------------------------------
-------
proc runSimulation { } {
    global ns_ finishSimulation opt
    for {set j 1.0} {$j < $opt(stop)} {set j [expr $j * 1.3 ]} {
      $ns_ at $j "puts t=$j"
    }
    $ns_ at $opt(stop) "finishSimulation"
    $ns_ run
}
puts "Starting Simulation..."
runSimulation
```

დანართი 2.

კვანძების გადაადგილების სცენარის ფრაგმენტი

```
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
#     Random Waypoint Model
#     numNodes   =     30
#     maxX       =   2000.00
#     maxY       =    400.00
#     endTime    =    800.00
#     speedMean  =    1.4000
#     speedDelta =    0.0000
#     pauseMean  =      1.00
#     pauseDelta =      0.00
#~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

# output format is NS2
#     Initial positions:
$node_(0) set X_ 655.610253277485
$node_(0) set Y_ 108.563122113177
$node_(0) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(0) setdest 685.222499615934
100.704663254809 1.400000000000"
$node_(1) set X_ 406.496790908904
$node_(1) set Y_ 272.771329093102
$node_(1) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(1) setdest 311.916352361372
289.279204668467 1.400000000000"
$node_(2) set X_ 1003.455950596534
$node_(2) set Y_ 84.314025550113
$node_(2) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(2) setdest 900.158169524625
219.392104183840 1.400000000000"
$node_(3) set X_ 614.279621461520
$node_(3) set Y_ 200.700196313696
$node_(3) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(3) setdest 1398.686677424510
220.166872064518 1.400000000000"
$node_(4) set X_ 677.215227275173
$node_(4) set Y_ 140.094137575755
$node_(4) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(4) setdest 1125.197152474063
12.793074886732 1.400000000000"
$node_(5) set X_ 438.165437892197
$node_(5) set Y_ 215.644344614315
$node_(5) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(5) setdest 571.360779955955
174.133008796793 1.400000000000"
$node_(6) set X_ 938.498451801635
$node_(6) set Y_ 212.825757166815
$node_(6) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(6) setdest 1448.089975055971
291.826898556836 1.400000000000"
$node_(7) set X_ 1037.200597695065
$node_(7) set Y_ 83.439080247497
$node_(7) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(7) setdest 1231.597459871274
79.227681520538 1.400000000000"
$node_(8) set X_ 1249.595648546719
$node_(8) set Y_ 29.859678224813
$node_(8) set Z_ 0.000000000000
```

```
$ns_ at 0.000000000000 "$node_(8) setdest 1432.244379741621
29.876949988506 1.400000000000"
$node_(9) set X_ 654.441215523928
$node_(9) set Y_ 45.174767878264
$node_(9) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(9) setdest 673.106870381260
39.282504904097 1.400000000000"
$node_(10) set X_ 312.373793207546
$node_(10) set Y_ 251.089051051471
$node_(10) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(10) setdest 359.341341951384
286.164006836894 1.400000000000"
$node_(11) set X_ 423.876766708258
$node_(11) set Y_ 164.593046871370
$node_(11) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(11) setdest 842.184315951740
173.586268434925 1.400000000000"
$node_(12) set X_ 1095.111374711666
$node_(12) set Y_ 172.789318671321
$node_(12) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(12) setdest 1269.182221135593
122.287184812319 1.400000000000"
$node_(13) set X_ 1383.790076775177
$node_(13) set Y_ 57.513108807300
$node_(13) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(13) setdest 271.096515301162
101.747775893275 1.400000000000"
$node_(14) set X_ 446.640587881878
$node_(14) set Y_ 186.153349632326
$node_(14) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(14) setdest 441.901365889199
187.229406104085 1.400000000000"
$node_(15) set X_ 547.616258667753
$node_(15) set Y_ 253.482991119668
$node_(15) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(15) setdest 1318.698752615530
192.643145185665 1.400000000000"
$node_(16) set X_ 400.462468447073
$node_(16) set Y_ 93.496084342724
$node_(16) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(16) setdest 283.107326811030
170.667651282049 1.400000000000"
$node_(17) set X_ 47.906815432499
$node_(17) set Y_ 210.453139549370
$node_(17) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(17) setdest 1313.250009983372
288.885285030761 1.400000000000"
$node_(18) set X_ 451.754773838931
$node_(18) set Y_ 164.674847209366
$node_(18) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(18) setdest 277.646503070483
229.436550077037 1.400000000000"
$node_(19) set X_ 949.435016046316
$node_(19) set Y_ 263.131698207685
$node_(19) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(19) setdest 205.888541382390
277.888340158315 1.400000000000"
$node_(20) set X_ 1304.973217374030
$node_(20) set Y_ 215.107101910347
$node_(20) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(20) setdest 504.851343909330
207.101038427086 1.400000000000"
```

```
$node_(21) set X_ 237.715371400180
$node_(21) set Y_ 160.558365159839
$node_(21) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(21) setdest 368.601983712136
58.804797850115 1.400000000000"
$node_(22) set X_ 891.356006549125
$node_(22) set Y_ 59.842299719869
$node_(22) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(22) setdest 738.016855966442
24.732606332775 1.400000000000"
$node_(23) set X_ 67.554204003726
$node_(23) set Y_ 172.335715757539
$node_(23) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(23) setdest 793.087868415094
17.766458881498 1.400000000000"
$node_(24) set X_ 238.519805801505
$node_(24) set Y_ 274.501688039486
$node_(24) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(24) setdest 1208.458913068853
229.699488385599 1.400000000000"
$node_(25) set X_ 1268.467194113869
$node_(25) set Y_ 192.141303979821
$node_(25) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(25) setdest 1175.108583089005
269.469577826314 1.400000000000"
$node_(26) set X_ 845.584750274672
$node_(26) set Y_ 54.900424659441
$node_(26) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(26) setdest 559.103911555983
25.686623549026 1.400000000000"
$node_(27) set X_ 1066.863745131681
$node_(27) set Y_ 54.336058548391
$node_(27) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(27) setdest 1189.503605809879
36.262953908121 1.400000000000"
$node_(28) set X_ 608.183973849892
$node_(28) set Y_ 40.449589866371
$node_(28) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(28) setdest 104.332560077776
37.808932122231 1.400000000000"
$node_(29) set X_ 1083.852989290221
$node_(29) set Y_ 109.538129481653
$node_(29) set Z_ 0.000000000000
$ns_ at 0.000000000000 "$node_(29) setdest 502.547955844304
146.760213227727 1.400000000000"


#     Movements:
$ns_ at 3.471320003174 "$node_(14) setdest 441.901365889199
187.229406104085 0.000000000000"
$ns_ at 4.471320003174 "$node_(14) setdest 155.522439862032
232.955413808856 1.400000000000"
$ns_ at 13.981135873423 "$node_(9) setdest 673.106870381260
39.282504904097 0.000000000000"
$ns_ at 14.981135873423 "$node_(9) setdest 921.911974000018
156.950832556647 1.400000000000"
$ns_ at 21.883743253720 "$node_(0) setdest 685.222499615934
100.704663254809 0.000000000000"
$ns_ at 22.883743253720 "$node_(0) setdest 1354.406086406592
75.087128569695 1.400000000000"
$ns_ at 41.870811751283 "$node_(10) setdest 359.341341951384
286.164006836894 0.000000000000"
```

```
$ns_ at 42.870811751283 "$node_(10) setdest 300.205351090616
150.597432315626 1.400000000000"
$ns_ at 68.578754489077 "$node_(1) setdest 311.916352361372
289.279204668467 0.000000000000"
$ns_ at 69.578754489077 "$node_(1) setdest 958.986142193914
251.626023154887 1.400000000000"
$ns_ at 86.589260859585 "$node_(25) setdest 1175.108583089005
269.469577826314 0.000000000000"
$ns_ at 87.589260859585 "$node_(25) setdest 1314.577177535430
67.837857019121 1.400000000000"
$ns_ at 88.546000204972 "$node_(27) setdest 1189.503605809879
36.262953908121 0.000000000000"
$ns_ at 89.546000204972 "$node_(27) setdest 767.424085913028
263.540937173061 1.400000000000"
```