ISSN: 2319-8753

International Journal of Innovative Research in

Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

PROTECTION OF WIRELESS SENSOR NETWORKS IN INDUSTRIAL AUTOMATION

S. Vivek Saravanan[1] and A. Solairaju[2]

[1] Final year B.E. Department of Electrical and Electronics Engineering, SRM Easwasri Engineering College, Ramapuram, Chennai, Tamilnadu, India

[2]Associate Professor of Mathematics, Jamal Mohamed College, Tiruchirappalli, Tamilnadu, India,

Abstract: We analyse a fundamental overview of information security and discusssome examples used throughout the rest of the paper. Our security engineering methodology is presented. This paper also presents the definition and evaluation of a new protocol for providing a wireless sensor network (WSN) with ahierarchical organization. Differently from previously proposed solutions, protocol, "clique clustering" includes in its operation a fail-safe mechanism for dealing with node failure or removal, which are typical of WSNs. The network is partitioned into clusters that are cliques (nodes in each clusters are directly connected to eachother's). An efficient mechanism for building a connected backbone among the clique clusters is provided. Clustering, backbone formation and backbone maintenance are completely localized, in the precise sense that only nodes physically close to a failing node are involved in the reconfiguration process.

I. INTRODUCTION

Using wireless communication has a similar effect as connecting the fabrication network to the Internet (the system becomes accessible from outside). The exchange of wired communication by wireless is much more severe than going for Internet access. This is due to the fact that wireless connections can be accessed from any position within the transmission range of the used wireless technology. Thus, potential attackers are no longer forced to enter the fabrication network at a well-defined entry point that is for wired Internet connections. Such entry points are normally powerful machines running strict firewalls. In contrast to those machines the new entry points might even be small sensor nodes, which have limited energy resources, limited processing power etc. Deploying standard protection means on sensor nodes might for example increase the processing time that much that dependability constraints will be violated. So, a straight forward re-use of those concepts on sensor nodes is infeasible. When designing new security solutions for automation networks it must be ensured that the core functionality (controlling a manufacturing site is not influenced by the security solution). This means constraints such as dependability issues and the current set-up of the system – consisting of software, protocols etc. need to be taken into account.

Two relatively new tendencies in the domain of automation systems are extremely relevant. The first is that more and more fabrication sites are connected to each other via public accessible networks such as the Internet. This approach is motivated by reducing cost for monitoring with a centralized control centre. The result is that the formerly isolated fabrication networks are now accessible from everywhere in the world, and by that all Internet-based attacks can be run against fabrication networks. The second tendency is to use wireless communication to a larger intent than up to now. The idea here is to allow more flexible set-ups of manufacturing sites and reduce cost for monitoring of difficult-to-reach devices.

We reflect this by introducing an additional engineering constraint into our semi-automatic approach, namely environment. The contribution of this paper is the introduction of a holistic but still easy to implement approach which allows engineering security solutions for automation networks. Our approach considers formerly not modeled constraints (environment) and dependability issues as well as the idea of economically secure systems. By this term we denote the fact that a security solution must ensure that thecost of an attacker to break the security solution is higher than his/her potential benefit.

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

II. PREVIOUS WORK

In this section, we outline results of related work that can provide ideas for further refinement of our approach. Though it is generally accepted that security issues can only be solved in holistic approaches including security protocols, physical environment and general policies, scientific solutions covering the full spectrum are rather rare. Even the question if a given system implementation satisfies the security goal in a well-defined environment is still a challenging question.

Ebyet. al. [2007] discussed the integration of security aspects into a formal method based development of networked embedded systems. The focus of the security analysis language (SAL) is merely on information flow between networked entities. By that it might be a way to model security requirements of applications and to verify whether or not the correct security modules were selected. VEST [Stankovic, 2003] (Virginia Embedded Systems Toolkit) focuses on the development of effective composition, configuration, and the associated dependency analysis. The tool helps the developer select and compose software components to a product. The analysis part even allows checking security properties, though it does not provide formal proof of correctness. Rather it applies key checks 7 and analysis to avoid many common problems. A tool that could be an example for a small solution library, entirely focuses on 'Security Through Usability' and has been published by the CRISIS project [Alcaraz et.al. 2008]. The authors categorized several key distribution schemes for sensor network applications. Based on the user inputs a suitable selection of protocols is presented. After entering main and secondary properties (examples: small memory, connectivity, scalability, resilience). The tool delivers a list of key distribution schemes that fulfill the requirements. Additionally the tool lists specific advantages and disadvantages of the algorithms, so that competent users have further information supporting the selection process. Security architectural patterns are discussed. Though the context of the studies is not as broad on system level, studying the proposed terminology can help improving the definition of our solution library. For example the notion of a security degree as part of a pattern descriptioncan be valuable for the objective security assessment process as it is required in our selection algorithm. Composition of security mechanisms is discussed in [Canetti, 2001]. They propose a framework that breaks down the security protocols in atomic cryptographic tasks that can be combined to composed protocols. An application of the idea inside the selection algorithm as well as an extension of the described cryptographic task towards combinable building blocks for safety and environment could be a promising approach.

III. GENERAL SECURITY TERMS

A general overview of information security is discussed. Then we show why standard solutions are not always suitable for WSANs and in particular in the context of industrial automation. Since even two instances of wireless implementations of automation systems can differ substantially, we finally conclude that first, there is no one-fits-all solution and second, that a clearly distinguished definition of the requirements is needed in order to find a suitable system architecture.

3.1 Information Security:Information security describes the properties of information systems, which ensure confidentiality, integrityand availability. In WSAN based automatic systems isthe information security particularly an economic factor. It defines the cost for an attacker to get important business information or to disturb the error-free operation of an industrial plant.

The three terms –known as CIA triad, Figure 1– confidentiality, integrity and availability are the core principles of information security. In this section we present a short description for these principles. Confidentiality is the ability to ensure that information is accessible only to authorized people or systems to have access. Integrity is the ability to ensure that data is an accurate and unchanged representation of the original secureinformation. Availability is the ability to ensure that data are readily accessible to the authorized all times. For an error-free operation of a facility it is very important that information are trustable and accessible all times, since measured data and controlling information regulate the workflow in sensor and actuator based industrial plants. If data regarding the facility's

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

workflow can be easily obtained by an attacker, it would pose the feasibility to gather important business information with a minimal investment.

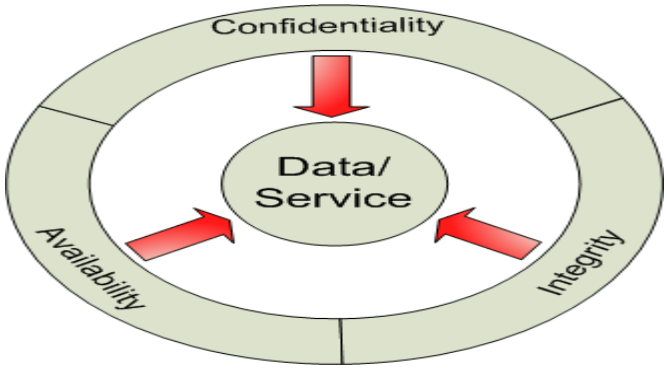


Figure 1. CIA triad: Confidentiality, Integrity and Availability are three key security principles in any kind of information system

3.2 Standard mechanisms: For sensor nodes in addition it can reduce data processing efforts because invalid packets do not need to processed in higher layers. Accountability and non-repudiation are very important for e-commerce and e-business systems. In WSANs these properties can be mostly covered with the mechanism used to ensure integrity.

Table Standard mechanisms for the seven

Protection	**Goals**
Protection	Goal Mechanism
Confidentiality	Encryption
availability	redundancy, filtering
authorization passwords	Filtering
Non-Repudiation signatures,	Logging
Authentication	Signatures
integrity	secure hashes
accountability audit	Logging

For all security goals mechanisms have been developedin the Table. Standard mechanisms for the seven protection goals

3.3 Security in WSANs:With the substitution of wired communication with wireless communication –not only in the context of automation– also a lot of new security threats emerge. First of all there is the insecure open channel. Everyone in radio transmission range can eavesdrop messages. Also everyone in range could interfere, i.e. send or modify messages. This means that protocols such as industrial Ethernet and field buses do no longer satisfy all requirements with respects to reliability and security under the new conditions. The access to critical systems cannot be controlled by administrative guide lines or physical barriers anymore. Consequently, the embedded security means do not hold in the new environments. The first idea would be to change the existing protocols so that they fit in the new environment.

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

But that is rarely possible for existing industry plants. Instead usually new components and sub-systems, like the wireless system, have to be adapted in order to be attached to the existing infrastructure. Therefore it is essential that the new wireless components ensure security parameters in such a way that the security and dependability requirements of the existing system will not be compromised.

In the following we introduce two examples of existing automation systems that should be extended with wireless networks. Regarding protection goal security, we will clarify that existing mechanisms of wired networks can not be transferred directly to WSAN systems. Instead it will be necessary to add additional mechanisms to reach the same level of security.

3.4 Example:Waterworks

The current architecture of the given waterworks facility consists of a wired network which connects a set of sensors on wells, filters and clear water pumps with decentral nodes, a programmable logic controller (PLC) and a central monitoring and control station. Industrial Ethernet connects PLC, the central station, and decentral peripheral nodes. The sensors are directly connected peripheral nodes. The basic idea is to replace the direct linkof the peripheral nodes by a wireless connection. The sensors will be placed in a wide area with a long distance to the peripheral nodes. The links are used for measurement data and controlling commands. The captured data will be the base for the controlling commands and a smoothlyoperation of the system. The system has to achieve the following basic security requirements:

_ Confidentiality of captured data

_ Authenticity of communication peer

_ Access control for open access points

For wired infrastructures only two security problems are imminent: first the internal unintended misuse, which can be solved by training the administrators and using an easy to use and clearly user interface, and secondly a potential connection between the productive and a public accessible network. The latter can be solved by using a very strict firewall or physical isolation of productive from public networks. In contrast for WSANs the connection to public accessible networks is ubiquitous. The isolation of the networks can not longer be achieved by fences orby the enclosing of the systems. The attacker –insider or foreigner– can use a mobile device to penetrate the network and has to be only in the near of a wireless node or access point. Furthermore the existing problems will be more severe. Every misuse can deactivate the necessary protection system and make the whole plant vulnerable. Mobile devices become a gateway from the Internet to the sensor nodes.

3.5 Example: Robot Cell

The robot cell is an example of the factory industry and consists of a robot arm with changeable tools and a tool depot. Every tool has a different set of sensors and actuators, which are controlled by a central unit. The central unit is usually a PLC as part of the robot cell or is located in the near of it.For the project demonstrator we use a wireless connection for the sensors and actuators of a tool. The motivation is that with wireless connections the replacement process can be faster and less error-prone. The system has to achieve the following basic requirements:Covering of control data; Authenticity of sensors and actuator

In a wired infrastructure the basic requirements can be covered without additional security means, since access to the communication network is strongly restricted by fences, production hall etc. For a WSAN this is not longer a valid assumption. The wireless components can also be controlled by an attacking unit outside the production hall.

3.6 Environment-Driven Constraints:Using standard solutions for the new security threats ishardly suitable for embedded WSANs. Industrial plantsare using a lot of different communication protocols andhardware with an intense focus on dependability. Standardsecurity solutions are often very expensive in computationand generate a significant protocol overhead. Tofit a protection goal in embedded devices an adoption of3

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

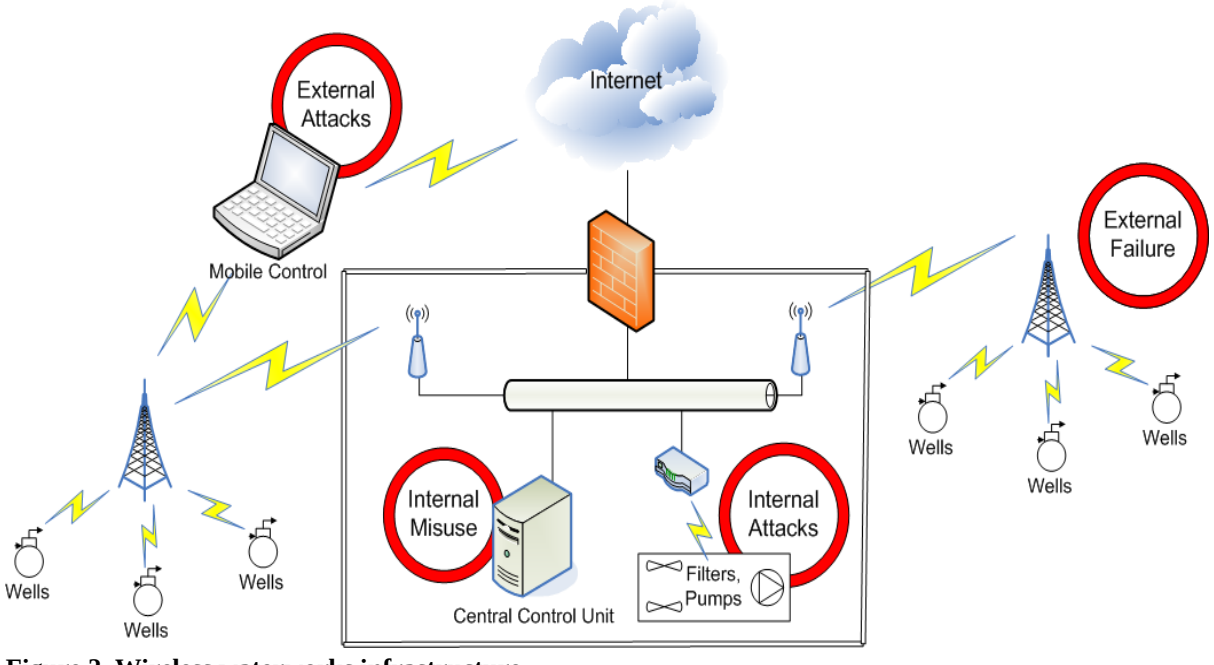


Figure 2. Wireless waterworks infrastructure

An existing solution or a distribution in hardware and softwarebecomes necessary. Identifying the ideal trade-off isa non-trivial issue.In both examples the new components will be connectedby wireless links and hence will be vulnerable andcan be miss-used as point of entrance. In this section wewill explain why a good embedded security solution forthe waterworks is not a good solution for the robot cell.In both examples the final link to the sensors should bereplaced by a wireless connection. The data transferredover these links are controlling and measure data.

Accessingor overtaking a sensor node by an attacker cancompromise the operation of both plants. Capturing datacan gain a benefit in an imaginable industrial or nationalcompetition. It would be necessary to cover the securitygoals confidentiality, authenticity and authorization inthose wireless architectures.A potential solution for waterworks is using standardencryption for the controlling and measured data. Thesensor nodes and the access points are powerful enoughand have no problems with power consumption. Becauseof the easy accessible location of the sensor nodes especiallyat the wells, we need also good authentication, integrityand authorization.

3.7 Development Flow

The fundamental idea of our approach is shown as Figure3. The result of the system analysis process is a list oftarget properties (Security goals, dependability requirementsand environmental constraints). Driven by the requirementswe start an iterative process that successively4

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

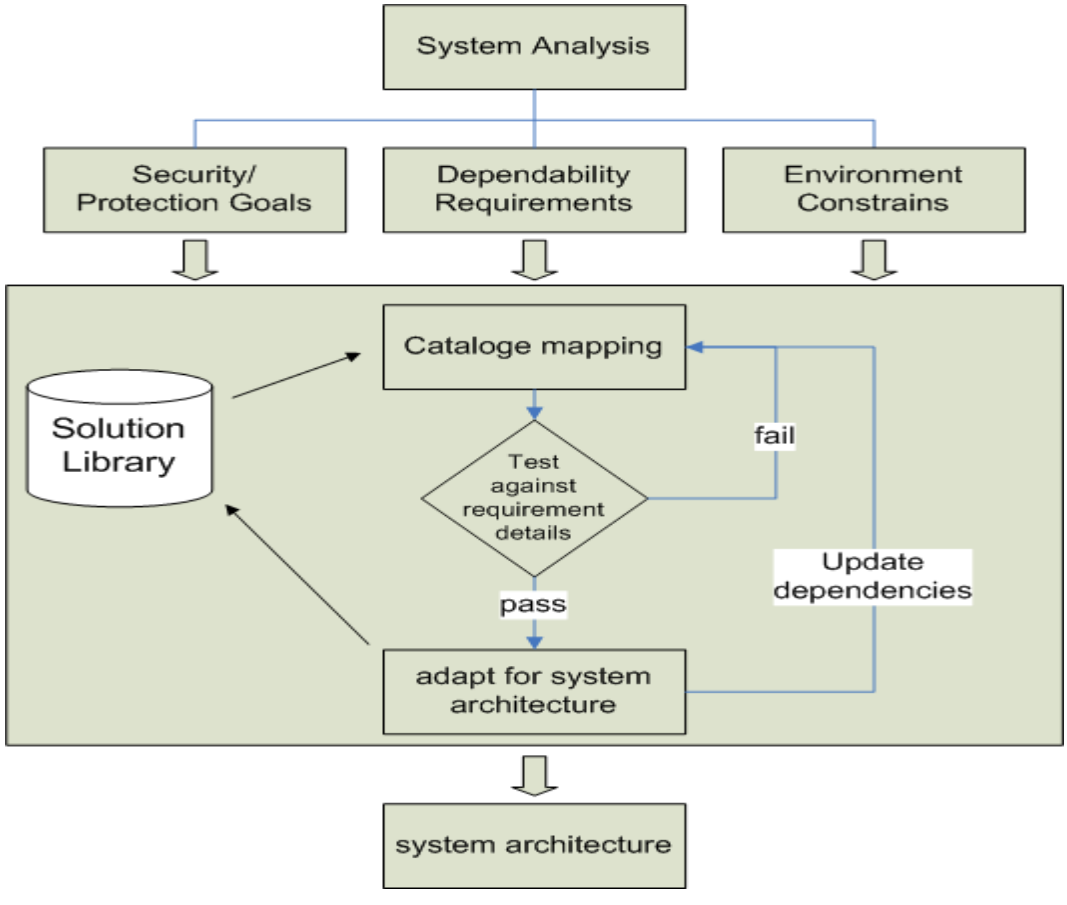


Figure 3. Flow of the selection process: Definitionof Security, Dependability and Environmentare input for iterative selection process.

The solution library allows reuse; takes promising solutions out of a solution library andtries to attach them to the system under development. Anevaluation after each step tests the outcome of the incrementallyextended system. If the extension is beneficial,i.e. the test is passed, the system architecture will beadapted. The new system –even if it does not satisfy allgiven requirements– will be added to the solution library,so that the knowledge base is extended for the future. Afterupdating the dependencies and solved requirements ofthe new system, a new iteration of the development willstart. That process will be repeated until a system architectureis found that satisfies all given requirements. Thisarchitecture will be the blueprint for the actual system integration.

3.8 Confidentiality as security goal:

Its need does notdepend on the environment. It depends on data and asort of degree characterizing the security strength. Consequentlythe requirement definition of concealment –justlike for the other requirements– must be defined isolatedfrom environmental aspects. For example the requirementsregarding integrity of data in a facility are unaffectedby the used network. If the facility switchesfrom wired to wireless, the security requirements will notchange, but just the environment. Indeed the eventual solutionwill change significantly but the inputs to our processwill change just slightly.Due to the strict separation of security, dependabilityand environment in our definition process we are able topose questions that aim toward a precise and objectiveproblem definition. At this point the questions mostlyconcern whether specific properties (e.g. concealment, integrity)are needed. For a precise definition process it isalso imperative to define the degree of each feature. Potentialmetrics are the assumed cost or

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

duration that areneeded to break the mechanism. However, in order to illustratethe general idea, in this paper we stress the pureexistence of specific requirement.

For automated integration arather formal description of dependencies, constraint andproperties would be required.Each sub-square contains a set of solutions for the specified requirement combination.

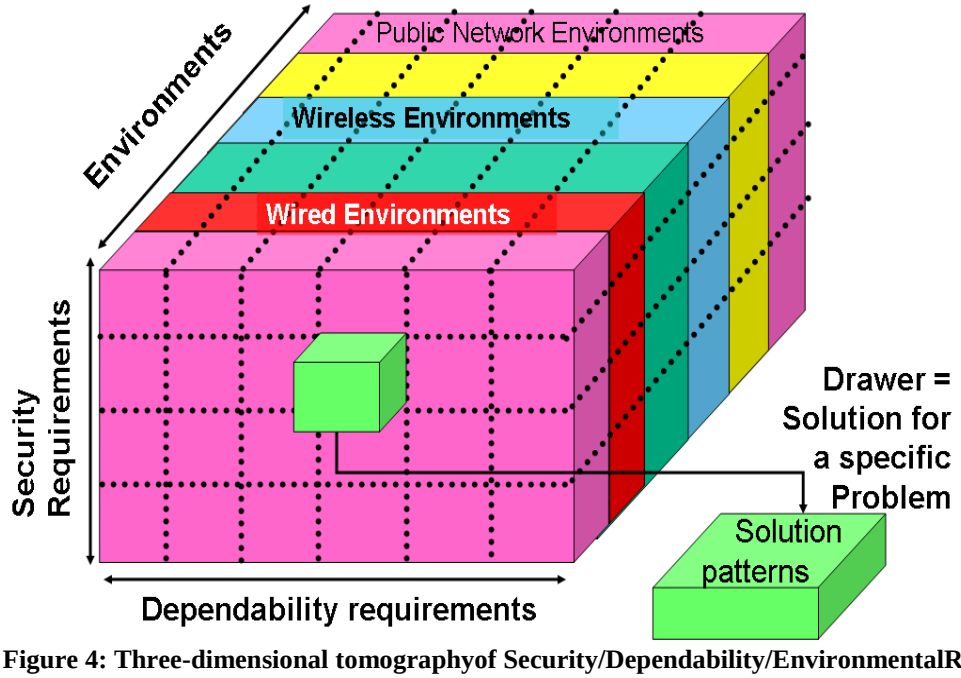


Figure 4: Three-dimensional tomographyof Security/Dependability/EnvironmentalRequirements.

IV. FAILURE AND SAFE USING CLIQUE COVERING

4.1 Introduction: The extensive research and experimentation performed in thepast couple of decades witness the remarkable interest of theacademic and industrial community in wireless ad hoc and sensornetworks (WSNs), and in protocols that make possible to deploythese networks effectively and at reasonable cost. Among thetopics that have received particular attention, *clustering and backbone formation* have triggered quite a community-wide discussion.

The reasons are to be found in the traditional use ofa superimposed hierarchical structure over the "flat" network topology for favoring routing scalability in terms of routing table size, reduced routing overhead, etc. There are also reasons that pertain specifically to wireless multi-hop networks such as adhoc and WSNs. In the latter case, especially, clustering provides a natural choice for selecting aggregation points where to merge redundant sensed data. It has been demonstrated that the beneficialeffects of data fusion largely counteracts the sub-optimality of routes over the backbone [Marcucciet.al., 2003]. Backbone formation also provides a straightforward way for topology control. Once clusters and a backbone have been formed, one node per each cluster (the clusterhead) remains awake to perform the network operations, while the radio interface of all the other nodes is turned off (sleep mode) for energy saving and prolonged network lifetime [Basagni et.al., 2004].The majority of clustering and backbone formation protocols proposed for WSNs have been designed for static, or quasi static networks. This implies that when nodes move away, or fail, the clustering process must be re-executed. Recently someprotocols have been proposed for ad hoc networks [Basagni, 1999] and forWSNs [Basagniet.al., 2004] that explicitly cope with these different types of network dynamics without repeating the whole clustering protocol. Clustering maintenance, as well as backbone maintenance, however impose non-negligible overhead, and decrease the overall network performance. (For quite an extensive list of works onclustering, [see Basagni et.al., 2006]).

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

The topic we deal with in this paper concerns showing that a clique-based clustering can provide an effective solution for dealing with nodal failures. Furthermore, we demonstrate that this approach outperforms other well-known and established connected dominating sets-based approaches to clustering, which also deals with clustering and backbone reorganization after nodal removal/failure, such as DMAC [Basagni, 1999]. Here we review those solutions that have been presented so far for clique clustering.One of the first works on the subject is [Krishna et.al., 1997], where clustering is used for making routing in dynamic (ad hoc) networks more scalable. More precisely, the authors investigate the partitioning of the network into clusters of diameter k, with the particular case of a cluster being a clique when $k = 1$.

4.2 Cliques covering:We discuss and evaluate a new protocol where coping with nodal failures, and especially with the removal of a node from a WSN, is embedded in the design of the clustering and backbone formation protocols. The idea is that of building, and connecting, clusters which are resilient to the failures of one or more nodes. For this purpose, we introduce a new protocol where the clusters are *cliques*, i.e., subsets of the network nodes that are all connected to each other. The protocol, termed *clique covering*, or CC (*double c*) for short, is executedat each node (it is distributed) based on simple assumptions and information about the node's immediate neighbors (it is localized). Once the CC clusterheads have been selected a backbone is efficiently formed that is connected. We demonstrate CC through simulations. In particular, we show its effectiveness in quickly and efficiently responding to nodal failure/removal by comparing it with DMAC, a clustering protocol for which backbone construction and reorganization have been defined andtested. The Distributed Mobility-Adaptive Clustering (DMAC) was originally proposed to extend clustering algorithms for static multi-hop scenarios with mechanisms for coping with the mobility of nodes. (For details the reader is referred to [Basagggni, 1999].) DMAC gives an example of how the removal or failure of a node cantrigger a chain reaction involving nodes in the network that can be quite far from the one removed/failed. CC and DMAC arecompared with respect to metrics that show how building clusters that are cliques is effective in reducing time, overhead and the number of nodes involved in a reorganization while producing reasonably small backbones. Metrics of interest to our comparison include the following (all taken after nodal removal or failure): Time it takes to re-build a connected backbone, overhead for cluster construction and maintenance, number of nodes involvedin the backbone reorganization, backbone size and route length. We observed that while producing backbones bigger than DMAC (up to 39%), CC reacts to network changes efficiently and quickly.

4.3 Clique Covering (CC)

The CC protocol produces a clustering that satisfies the followingproperties: 1) Every non-clusterhead node has at leasta clusterhead as neighbor (i.e., the set of clusterheads is adominating set); 2) Every node in a cluster can communicatedirectly with every other node in the cluster (clique property),and 3) every non-clusterhead node affiliates to the cluster of thefirst clusterhead inviting it.In describing CC we assume that every node knows its ownunique identifier (ID), its own *weight* and the ID and weight ofeach of its neighbors. The weight of a node is a real number ≥ 0which depends on the node current status and application requirements,and that indicates the suitability of the node for beingselected as a clusterhead. The higher the weight the better is anode for assuming that role. The protocol is executed at each node$v$ in such a way that v decides its role (clusterhead or ordinarynode) as soon as its "heavier" neighbors (neighbors with biggerweight) have decided their own role. The protocol is started bythe *init nodes*, i.e., by those nodes that have the bigger weightamong all their neighbors (ties are broken via the node uniqueID). An init node sends a (broadcast) message telling it will bea clusterhead. Upon receiving this message, a node exchangeswith the sender information about its own neighbors. Based onthe received information, a clusterhead selects all those neighborsthat can be associated to its own cluster while maintaining theclique property, and invites them to join it. A node whose heavierneighbors have joined other clusters or have finished invitingnodes, and that has not been invited to be part of any clusterdecides to be a clusterhead itself. The protocol terminates whenevery node belongs to a cluster being either a clusterhead oran ordinary node and knows the role and clusterhead of all itsneighbors. Except for the initial procedure, which is executedby each node when it starts the protocol operations, CC ismessage-triggered.

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

4.4 Six messages:We have six types of messages as follows.

1) CH(*v*): Sent by a node *v* to declare that it will be a clusterhead. Thisis a broadcast message, i.e., a message that has to reach all*v*'s neighbors.

2) NEIGHBORS(*v*,*u*): Sent by a node *v* to itsneighbor*u* to communicate *v*'s neighbor list. This is a unicastmessage from node *v* to node *u*.

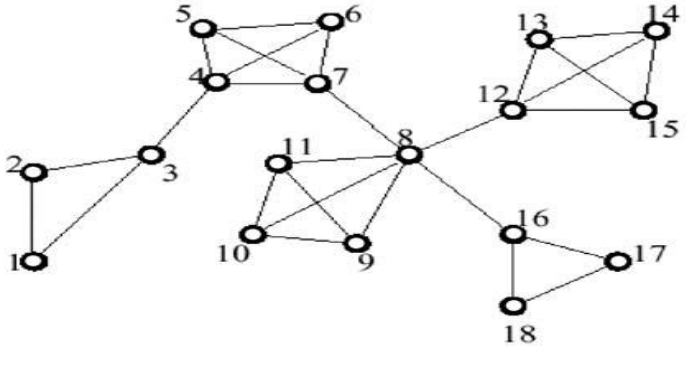
3) ASSOC REQUEST(*v*,*u*): Node *v* invites node *u* to be part of its cluster by sending tonode *u* this unicast message. Nodes are invited according to theirdegree, i.e., to the number of their neighbors: Nodes with higherdegree are invited first.

4) ASSOC RESPONSE(*u*,*z*): Sent by anode *v* to its neighbor*u* to communicate *v*'s response to *u*'sassociation request. If the response is **yes** node *v* communicatesto its clusterhead*u* to be part of *u*'s cluster, *z* = *u*, otherwise node*v* communicates to a clusterhead*u* to be part of *z*'s cluster, *z* _= *u*.

5) ASSOC FINISH(*v*): Sent by a node *v* to declare that it hasconcluded its cluster formation. This is a broadcast message. Thismessage has the meaning of communicating that a clusterhead hascompleted its cluster. It will not try to enroll any other nodes.

6)JOIN(*v*,*u*): Sent by a node *v* to declare that now it is associatedto clusterhead*u*. This is a broadcast message.Lack of space prevents us from giving the actual pseudo code ofthe CC's procedure.

4.5Example:Therefore we just describe the functioning ofthe algorithm through an example. Consider the Figure 5.consisting of a WSN, the CC-induced clustering and a backbone connecting the clusters.



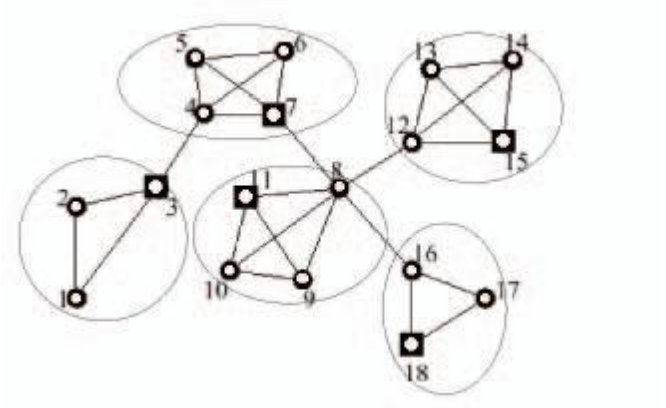
5(a) A sensor network

ISSN: 2319-8753

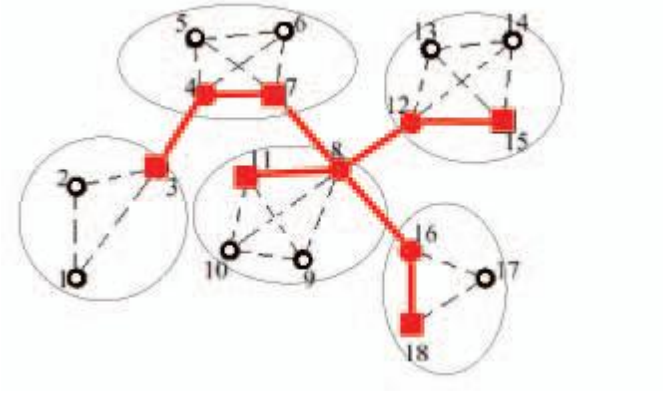
International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013



5(b) Clique covering



5(c) A connected backbone

4.6Backbone formation. The formation of a connected backbonestarts as soon as the cliques are formed. At this time, every nodeknows the ID and weight of each neighbor as well as the ID andthe weight of the clusterhead to which each neighbor is affiliated.This information is enough for each node to compute the pathsto clusterheads different from its own that are one and two hopsaway. For each clusterhead one path is selected. In particular, if anode is one hop away from a clusterhead different from its own,this path is chosen. Otherwise, a path with an intermediate nodeto another clusterhead is chosen, preferring an intermediate nodewith greater weight (ties are broken based on the ID).

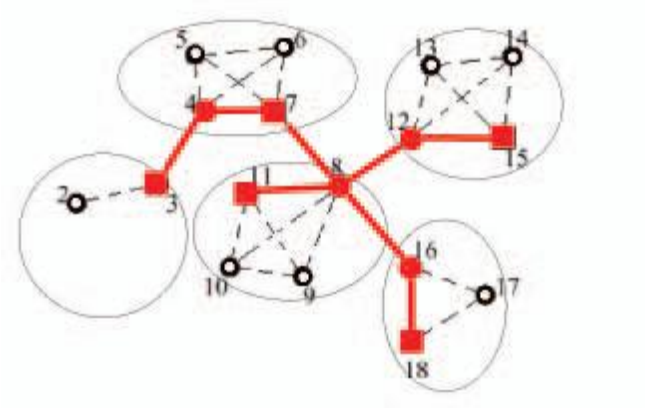
Finally, ifthe node is not a clusterhead, it informs its own clusterhead ofthe information just gathered.Once a clusterhead has determined its own connection informationand has received this information from its own nodes,it can determine the best paths to all the clusterheads that areat most three hops away. This is the *necessary* and *sufficient*condition to form a connected backbone (the proof closely followsthe lines of that in [Stankovic, 2003]). More specifically, clusterheads that areone hop away are joined by their direct link. Clusterheads thatare two hops away are connected via the common neighbor withthe greatest weight (ties are broken based on the unique nodalID). Finally, clusterheads that are three hops away are joined byselecting two *intermediate* nodes between them. When a clusterhead*v* has calculated the routes to all the otherclusterheads up to three hops away, it sends this information tothe nodes in its cluster. Therefore, each node knows if it is usedas gateway to interconnect to adjacent clusters, and how to reachneighboringclusterheads.

ISSN: 2319-8753

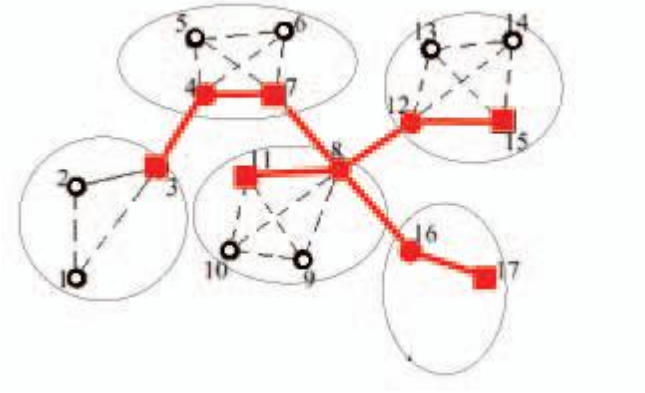
International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013



(a) Node 1 fails: Nothing changes



(b) Clusterhead18 fails: 17 is the new clusterhead

Figure 6. Backbone reorganization

***4.7 Backbone reorganization**:* Perhaps, what makes the CC solution most interesting is the ability of the nodes in the backbone to reorganize themselves quickly and efficiently when one of the backbone nodes fails. Here we give a brief overview of the actions undertaken by a node after the failure of a neighboring backbone node through the following example. (The actual "clique advantage" will be then made clear in the next section, where we quantify in terms or time and overhead the difference between CC and previous solutions). Let us consider the cluster partition depicted in Fig. 5(b).

Assume now that node 1 fails. Node 2 and clusterhead3 are made aware of the failure, and remove 1 from the list of nodes in their cluster. Node 1 is an ordinary node so no other reorganization or information exchange is needed. The result of the reorganization procedure is depicted in Fig. 6(a). Node 3's clusterhead is still a clique and the backbone is still connected. Assume now that clusterhead18 fails. In this case nodes 17 and 16 are made aware of the failure, and remove node 18 from the list of nodes in their cluster. Nodes 17 and 16 then need to check if they have to forward the failure information.

Node 17, not being a backbone node does not have to forward any information. However, node 16 needs to send a DEAD message to node 8 because it was used with node 8 to connect clusterheads11 and 18. Then each node checks if it is the better node in the cluster to act as clusterhead. Node 16 knows that node 17 is better and therefore does not promote itself to clusterhead. Node 17 instead knows that it will become the new clusterhead and

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

broadcasts this information via a CH message asking also for connection informations with a INFO REQUEST message.

When node 8 receives the DEAD message from node 16 it updates its information about node 18, forwards this information to its clusterhead11 and locally records that node 16 must select a new clusterhead. When node 16 receives the CH message sent by 17 it replies broadcasting a JOIN message to make aware all neighbors about its new clusterhead. When it receives the INFO REQUEST message it replies with an INFO message saying that it can reach clusterhead11 through node 8. When node 11 receives the DEAD message sent by 8 it updates its information about node 18 and sends an INFO REQUEST to nodes in its cluster to try to find a route to the new clusterhead. Node 8, after the reception of the JOIN message sent by node 16, knows the new clusterhead of node 16 and can reply to its clusterhead saying that it can reach the new clusterhead17 through node 16.

Now, clusterheads11 and 17 know how to reach each other. At the end of the reorganization procedures the new clique is created and the backbone is still connected (Fig. 6(b)). Only local exchange of information are needed to react to a clusterhead failure. The case of gateway failure is treated similarly to the failure of a clusterhead. In this case, it is not needed to find a new clusterhead, and only the INFO REQUEST and INFO messages

are exchanged to find routes to connect the clusterheads.

V. CONCLUSION

A holistic has been presented approach for engineering security solutions for automation networks. One of the main innovations is the inclusion of non-security parameters such as dependability and engineering constraints resulting from existing systems, which we call environment. The latter is extremely important since it allows to explicitly model implicit assumptions (example: about confidentiality which is given if a system is physically shielded from its environment, which holds no longer true if wireless communication is used. The second innovative aspect is the semi-formal search for security solutions guided by the above mentioned constraints.

In addition, we have discussed a new distributed and localized protocol for clustering and backbone formation in wireless sensor networks. *Clique clustering* (CC) partitions the network into cliques, thus providing each cluster with a fail-safe mechanism for quickly reconfiguring itself when nodes fail or leave the network. Backbone formation and reorganization techniques render CC a solution for hierarchical organization of WSNs that efficiently deals with node removal/failure.

DMAC incurs almost seven times higher overhead than CC, and needs up to 63% longer time to reconfigure after nodal failure. Therefore, CC provides a suitable solution for organizing dynamic WSNs hierarchically in realistic scenarios, where overhead-imposed energy consumption and reorganization time must be minimized. Previous works on partitioning a multi-hop network into cliques are reviewed. It describes our CC protocol as well as backbone construction and reorganization after nodal failure/removal.

In addition, the focus is demonstrating the effectiveness of the presented ad hoc routing protocols in delivering packets among pairs of selected nodes, and the clustering is seen as a mean of favoring scalability. In other words, clustering is here only functional to routing and the description of the clustering organization, the analysis of its properties and its demonstration is not presented. Furthermore, in dense scenarios, this solutions is not practical, since it calculates all cliques a node can belong to and this computation generates quite a remarkable amount of overhead, with a detrimental effect on network performance.

REFERENCES

[1] C. Alcaraz, J. Lopez, and R. R. Castro. Choosing a key distribution protocol for your sensor network. http://www.lcc.uma.es/˜roman/KMSCRISIS/, Jan 2008.

[2] A. Marcucci, M. Nati, C. Petrioli, and A. Vitaletti. Hierarchical routing and data aggregation in wireless sensor networks. Technical report, Universit`a di Roma “La Sapienza”, Roma, Italy, October 2003.

ISSN: 2319-8753

International Journal of Innovative Research in Science,Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 12, December 2013

[3] S. Basagni, A. Carosi, and C. Petrioli. Sensor-DMAC: Dynamic topology control for wireless sensor network. In *Proceedings of the* 60*th IEEE Vehicular Technology Conference, VTC 2004 Fall*, volume 4, pages 2930–2935, Los Angles, CA, September 26–29 2004.

[4] S. Basagni. Distributed clustering for ad hoc networks. In A. Y. Zomaya, D. F. Hsu, O. Ibarra, S. Origuchi, D. Nassimi, and M. Palis, editors, *Proceedings of the 1999 International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN'99)*, pages 310–315, Perth/Fremantle, Australia, June 23–25 1999. IEEE Computer Society.

[5] S. Basagni, M. Mastrogiovanni, A. Panconesi, and C. Petrioli. Localized protocols for ad hoc clustering and backbone formation: A performance comparison. *IEEE Transactions on Parallel and Distributed Systems, Special Issue on Localized Communication and Topology Protocols for Ad HocNetworks*, 17(4):292–306, April 2006.

[6] R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In IEEE Symposium on Foundations of Computer Science, 2001.

[7] I. Chlamtac and A. Farag´o. A new approach to the design and analysis of peer-to-peer mobile networks. *Wireless Networks*, 5(3):149–156, May 1999.

[8] M. Eby, J. Werner, G. Karsai, and A. Ledeczi. Integrating security modeling into embedded system design. In International Conference and Workshop on the Engineering of Computer Based Systems. IEEE, 2007.

[9] Y. Fernandess and D Malkhi. *k*-clustering in wireless ad hoc networks. In *Proceedings of the Second ACM International Workshop on Principles of Mobile Computing, POMC 2002*, pages 31–37, Toulouse, France, October 30–31 2002.

[10] P. Krishna, N. H. Vaidya, M. Chatterjee, and D. K. Pradhan. A cluster-based approach for routing in dynamic networks. *ACM SIGCOMM Computer Communication Review*, 27(2):49–64, April 1997.

[11] J. Stankovic, R. Zhu, R. Poornalingam, C. Lu, Z. Yu, M. Humphrey, and B. Ellis. Vest: An aspect-based composition tool for real-time systems. In Proceedings of the IEEE Real-time Applications Symposium, 2003.

BIOGRAPHY

[1]S. VivekSaravanan is doing Final B.E. (Electrical and Electronics Emgineering), Eawasri Engineering College, Ramapuram, Chennai, Tamilnadu, India.

[2]Dr.A. Solairaju is working as an "Associate Professor of Mathematics", Jamal Mohamed College, Tiruchirappalli-620020, Tamilnadu, India. He wrote five books on Engineering Mathematics. He has published 150 research papers in National and International Journals. 10 students in Mathematics one student in Computer Science are awarded Ph.D. degree under this guidance. He is an editor in some National and International Journals.