Authentication Protocols for Ad Hoc Networks: Taxonomy and Research Issues

Nidal Aboudagga[1], Mohamed Tamer Refaei[2], Mohamed Eltoweissy[2], Luiz A. DaSilva[2], and Jean-Jacques Quisquater[1]

[1] UCL Crypto Group, Université Catholique de Louvain, Belgium

{aboudagg, quisquater}@dice.ucl.ac.be

[2] Bradley Department of Electrical and Computer Engineering, Virginia Tech, USA

{mtamer, toweissy, ldasilva}@vt.edu

ABSTACT

Ad hoc networks, such as sensor and mobile ad hoc networks, must overcome a myriad of security challenges to realize their potential in both civil and military applications. Typically, ad hoc networks are deployed in un-trusted environments. Consequently, authentication is a precursor to any secure interactions in these networks. Recently, numerous authentication protocols have been proposed for ad hoc networks. To date, there is no common framework to evaluate these protocols. Towards developing such a framework, this paper proposes a generic authentication process and a new taxonomy that clarifies similarities and differences among authentication protocols reported in the literature. The taxonomy is based upon the role of nodes in the authentication function, establishment of credentials, and type of credentials. We also motivate the need for an authentication management architecture and discuss some open research issues.

Categories and Subject Descriptors

A.1 [**General Literature**]: Introductory And Survey

General Terms

Security, Management, Performance

Keywords

Authentication, Network Security, Protocol Taxonomy, Ad Hoc Networks, Credentials, Identity Verification.

1. INTRODUCTION

Interest in ad hoc networks largely stems from the ability to rapidly deploy them under both normal and harsh conditions. These networks can be quickly deployed in situations where no infrastructure exists and it would be impractical or infeasible to deploy infrastructure. In such an infrastructure-less network, nodes are expected to cooperate to perform essential networking tasks such as routing. In order to provide network-wide connectivity, nodes in an ad hoc network are expected to route data packets on behalf of other nodes in the network that want to reach nodes out of their transmission range.

Ad hoc networks can be classified into static and mobile networks. Sensor networks (SensNets) typically are static ad hoc networks. On the other hand, mobile ad hoc networks (MANETs) are autonomous systems of mobile nodes that are free to move at will. A hybrid network may also exist. For example sensor nodes can form a tier in a network that is managed by a higher tier of mobile gateway nodes.

From a security standpoint, ad hoc networks face a number of challenges. The wireless medium has no observable boundaries and is significantly less reliable than wired media. Unlike wire-line networking, where an attacker must physically break into the network infrastructure, tap into network cables, or logically break through several lines of defenses (such as firewalls) before he can take control or tamper with any network component, wireless attacks may come from anywhere and from all directions [18]. Additionally, the lack of a clear line of defense and traffic concentration points poses a challenge to deploying security solutions in ad hoc networks. The broadcast nature of the transmission medium and the dynamically changing topology add even more complications. Furthermore, the reliance on node collaboration as a key factor of network connectivity presents another obstacle.

In order to provide network security, support for authentication, confidentiality, integrity, non-repudiation, and access control should be provided. We believe that authentication is the cornerstone service, since other services depend on the authentication of communication entities [19] [7]. Authentication supports privacy protection by ensuring that entities verify and validate one another before disclosing any secret information. In addition, it supports confidentiality and access control, by allowing access to services and infrastructure to authorized entities only, while denying unauthorized entities access to sensitive data.

A significant number of authentication protocols have recently been proposed for ad hoc networks; examples include [1] [2] [3] [4] [5] [6] [8] [9] [10] [11] [12] [13] [17] [18] [19] [24]. A classification is needed to interpret the similarities between sets of related protocols and to understand the motivation behind each. A classification also enables us to better analyze and compare protocols with respect to their encapsulating class rather than

Q2SWinet'05, October 13, 2005, Montreal, Quebec, Canada.

comparing individual protocols; to identify common vulnerabilities and attacks against each class of authentication protocols; and to identify common architectural elements in each class.

This paper presents a new taxonomy for the classification of authentication protocols in ad hoc networks. We identify three major criteria for classification, based on a node's role in the authentication process, the type of credentials used for authentication, and the phase during which the establishment of credentials takes place. The paper also motivates the need for an authentication management architecture and presents some open research issues.

The remainder of this paper is organized as follows. In section 2 we introduce different components of the authentication process in an ad hoc network and the authentication states of a supplicant (the entity requesting authentication). In section 3 we provide an overview of our taxonomy and present the three classification criteria proposed. In sections 4, 5 and 6 we discuss each of the three primary classes of the taxonomy. In section 7 we present an analysis motivating the need for authentication management architecture. Finally, section 8 concludes the paper and discusses directions for future work.

2. AUTHENTICATION IN AD HOC NETWORKS

Authentication is a process that involves an *authenticator* communicating with a *supplicant* using an *authentication protocol* to verify *credentials* presented by the supplicant in order to determine the supplicant's access privileges. A *Trusted Third Party* (TTP) may be involved as part of the authentication protocol.

The *supplicant* is an entity that is looking to gain access to some protected resources by being authenticated via an authenticator. An *authenticator* is an entity that protects and controls access to some resources. The authenticator facilitates the authentication process and makes authentication decisions. An *authentication protocol* is a sequence of message exchanges between entities (supplicant(s) and authenticator(s)) that either distributes secrets to some of those principals or allows the use of some secret to be recognized [20]. A *credential* is an identifier that can be used to authenticate a supplicant with high confidence. Finally, a *Trusted Third Party* is an entity that is mutually trusted by the supplicant and the authenticator and that can facilitate mutual authentication between the two parties.

An entity, be it a supplicant or authenticator, may be any of the following:

- *Person*: A person is a human user who is seeking authorization to use some resource (for example to use the email service offered by the university).
- *Agent*: An agent is a program that performs some service on a regular schedule without the user's immediate participation.
- *Service*: To access a service, such as an online banking system, a supplicant must authenticate itself to the service first before being granted access.
- *Node*: A node usually refers to a computing device that is connected to the network. Networks can have tens, thousands, or even millions of nodes. Laptops, personal digital assistants (PDA), sensors, and personal computers (PC) are all examples of nodes.
- *Group*: A group is a set of nodes or persons with common access privileges. Groups are common under UNIX based systems, where *persons* are grouped into *groups* that have similar access rights to the system.
- *Network*: In some cases, entities authenticate directly to the network, such as when participating in a Virtual Private Network (VPN).

2.1 Components of the Authentication Process

A generic authentication process has six major phases as shown in figure 1. *Bootstrapping* is the first phase, where a supplicant is securely provided, either offline or online, with something that it should *have* (a key) or something that it should *know* (a password) that authenticators would trust as a proof of the supplicant's eligibility to access protected resources or offer service. In [5], for example, bootstrapping is done by assigning a global network key to each new node joining the network, while in [2], nodes are bootstrapped by assigning each a list of trusted nodes.

Once the bootstrapping phase is completed, the supplicant is ready to participate in the network. The *pre-authentication* process is where a supplicant presents its credentials to an authenticator in an attempt to prove its eligibility to access protected resources or offer services. In [5] new nodes must demonstrate knowledge of the global network key (using challenge response, for example).

Once the supplicant's credentials are verified, a *credential establishment* process is invoked to establish the supplicant's new credentials, which it will use as a proof of its identity and as a verification of its authorized state thereafter. A credential could be a symmetric key, a public/private key pair, a commitment of a hash key chain, or some contextual information. The established credentials might be tagged with an expiry date after which the supplicant has to re-negotiate a new "certificate" of credentials. In [5], a node is assigned a portion of the network's private key in a *(k, n)* threshold cryptography mechanism. In [2], the authenticating parties use a chain of trust established between nodes in their trusted list to generate and perform a key exchange between them. In [13], a commitment key to a TESLA [22] based one-way key-chain is generated and distributed as a node's credentials.

Upon success of all of the steps above, a supplicant is considered authenticated, which means that it is authorized to access resources protected by the authenticator. Within the *authentication* state, all communication between the supplicant and the authenticator is authenticated by the source and validated at the destination using the established credentials. While authenticated, a supplicant's behavior is monitored for fear of its being compromised or misbehaving. A compromised supplicant may get its credentials *revoked* (as in [6]) or its re-establishment of credentials request denied when its credentials expire. In both cases, the supplicant is isolated from the network.

In this paper, we will focus on node-to-node authentication. To better understand the authentication process and protocols, we

will describe the authentication state diagram for a supplicant in the next section. The authenticator's state diagram may be easily constructed following the supplicant's state diagram and therefore it is not described here.

2.2 Authentication States for a Supplicant

The state diagram in figure 2 represents possible states of a supplicant during the authentication process. The first state *initializes* the supplicant. In this state, the supplicant is usually supplied with necessary tools to carry on an authentication function. These tools could be supported authentication protocols (e.g., TESLA, 802.1x), authentication credentials (e.g., signed certificates), or identities of trusted entities. At the end of the initialization state, a supplicant has all necessary tools to authenticate to an authenticator.

Once a supplicant is initialized, it is ready to move on to the next state, which is *discovery*. During the discovery state, a supplicant scans for reachable services of interest. Each available service is expected to advertise its presence and list service-access requirements. A reachable service is one that is capable of directly making the supplicant aware of its presence (e.g., through periodic advertisements). At the end of the discovery state, a supplicant has a list of reachable services and the service-access requirements for each.

The following state is the *selection* state. Based on the list of reachable services and the service-access requirements of each, a supplicant filters accessible services of interest. The supplicant matches the tools it was supplied with during the initialization state to the service-access requirements advertised by each service. If none of the services match, the supplicant goes back to the discovery state. At the end of the selection state, a supplicant has a list of matching accessible services that are of interest to it.

The next state is the *authenticating* state. The supplicant uses the tools it was supplied with during the initialization state to attempt to authenticate to the authenticator. If the authentication process was successful, the supplicant moves to the authenticated state; if it fails the supplicant goes back to the discovery state. Within the authenticated state, the supplicant is considered trusted and is given appropriate access privileges to resources protected by the authenticator. The supplicant is bootstrapped with credentials that can be used to prove its access rights from there after.

Following the authenticated state, the supplicant frequently enters an *evaluation* state where its behavior is examined. Based on the outcome of the evaluation process the supplicant could either return back to the authenticated state (i.e. well behaving) or is put under probation (i.e. selfish or malicious).

The *probation* state comes next, in which the supplicant enters as a penalty if it was determined to have behaved inappropriately. Eventually, the supplicant would be re-evaluated and given a chance to recover.

3. TAXONOMY OF AUTHENTICATION PROTOCOLS

We present a taxonomy based on the role played by nodes in the authentication, the type of credentials and when credentials are established.

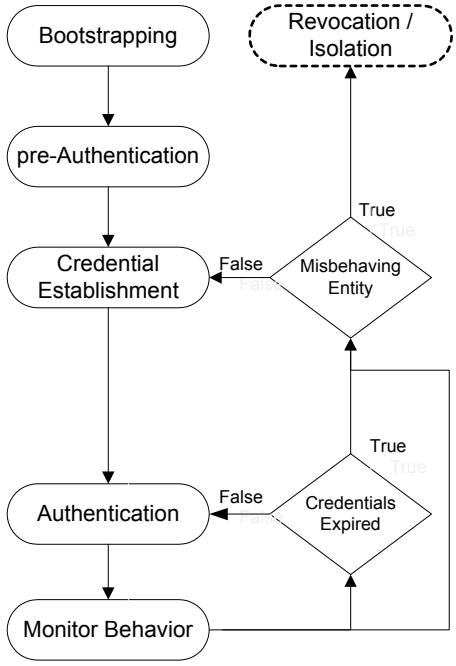


Figure 1 Functions in a Generic Authentication Process in Ad hoc Networks

Authentication protocols described in the literature have introduced a variety of ways in which the authentication function may be carried out. Some protocols assume reliance on a third party that is trusted by all nodes. The trusted third party represents a service whose signature on a supplicant's credentials is considered a proof of its identity and is relied on to make authentication decisions. On the other hand, other protocols assume no such service in the network. The first classification of our taxonomy recognizes such differences by categorizing authentication protocols based on the roles assigned to nodes in the network with respect to the authentication operation. Based on that, authentication protocols can be classified into two classes: homogeneous and heterogeneous.

The second classification recognizes different types of credentials used for authentication and categorizes authentication protocols based on that. As stated earlier, a credential is a unique identifier that can be used to authenticate a node with high confidence. Credentials may be classified into two classes. The first class identifies the supplicant based on a unique possession, while the second class identifies the supplicant based on context.

The third classification recognizes the phase when credentials are established. Some protocols establish credentials prior to node deployment, while other protocols assume credentials are established post node deployment. A third possibility exists, when some credentials are pre-distributed offline, but the actual credentials used for authentication are derived from the pre-distributed credentials.

While other bases for classification are possible, we believe that this classification is important since it captures variations shown in the literature in two important components that affect the operation of

authentication, the deployment scenario of an authentication infrastructure and means for authentication.

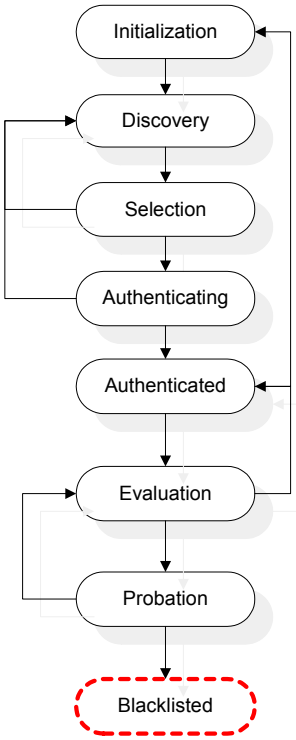


Figure 2 Node Authentication States

4. CLASSIFICATION BASED ON AUTHENTICATION FUNCTION

4.1 Homogeneous

Homogeneity indicates that all nodes in the network have the same role with respect to the authentication operation. This class of authentication protocols assumes that nodes in the network either make authentication decisions autonomously or they depend on information contributed by other nodes in the network to make such decisions.

Under the dependent homogeneous class of authentication protocols, authenticators rely on information from their trusted peers to make authentication decisions. Trust based mechanisms that use trust chains (i.e. recommendations from trusted nodes) fall under this class. On the other hand, in the autonomous homogeneous class, authenticators make authentication decisions autonomously without relying on their peers or any overlaying infrastructure. The use of demonstrative identification, identity based cryptography, and reputation based mechanisms such as [27] is common among protocols in this class.

In general, trust based mechanisms fall under the homogeneous class of authentication protocols ([15] provides seven different classes of trust that might be required in the interaction between entities wishing to communicate securely).

Examples of schemes that fall under the homogeneous autonomous subclass are [1] [3] [25] [6] [8] [11] [13], while [2] [5] [32] [23] [9] [10] [18] [26] are schemes that fall under the homogeneous dependant subclass.

4.2 Heterogeneous

The heterogeneous class of protocols indicates that nodes in the network have different roles with respect to the authentication operation. This suggests that there is an underlying service in the network that is meant to aid other nodes in making authentication decisions (e.g., a trusted third party). The underlying service could be centralized, where one specialized node is responsible for providing that service, distributed, where service nodes are deployed anywhere in the network responding to service requests from any node, or clustered, where nodes are clustered and each cluster has a unique provider of the authentication service.

Authentication protocols that are based on PKI or symmetric key fall under the heterogeneous authentication class.

Examples of schemes that fall under the heterogeneous centralized subclass is [14], while [16] & [17] are schemes that fall under the heterogeneous distributed subclass, and [4] & [24] are schemes that follow the heterogeneous clustered subclass.

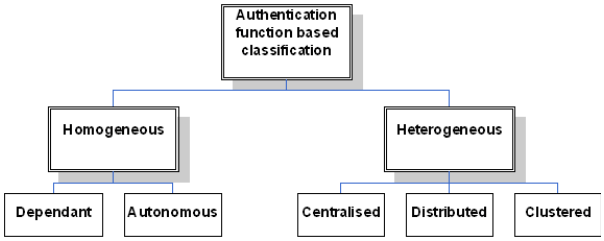


Figure 3 Classification based on node role

5. CLASSIFICATION BASED ON TYPE OF CREDENTIALS

This classification categorizes node authentication protocols based on the type of credentials used for authentication. Credentials can be classified into two classes: identity-based and context-based.

5.1 Identity-based credentials

This category recognizes a unique possession owned by the supplicant that could be used to identify it with high confidence. Usually, this is in the form of a key that is known to be unique to the supplicant. The authenticator could be assured of the supplicant's identity if it is certain that the supplicant possesses that key.

Identity based credentials can be further classified into encryption based and non-encryption based. An encryption based identity credential is a piece of information produced and cryptographically signed using the key possessed by the supplicant in order to verify its possession of the key, and hence prove its identity. In order to verify the supplicant's identity, the authenticator must either possess the same key (symmetric key cryptography), or the public-key component of the private-key owned by the supplicant (asymmetric key cryptography).

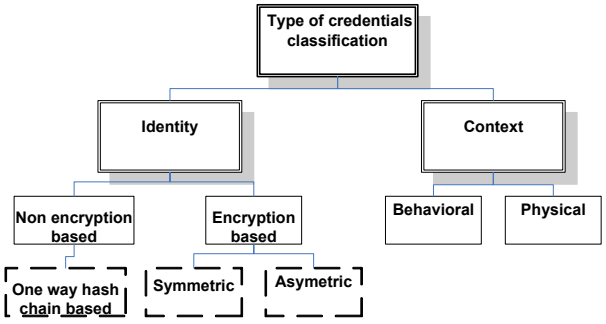


Figure 4 Classification based on type of credentials

Symmetric key based authentication is more common in sensor networks since it is less resource dependent compared to asymmetric key. On the other hand, asymmetric key based authentication, or public key cryptography, requires deployment of a public key infrastructure (PKI). In other words, it requires the presence of a trusted authority whose function is to bind entities' identities to their public keys and issue a signed certificate proving their authenticity. The service of such an authority must be available anytime anywhere.

One form of non-encryption based identity credential is information that is hashed using a one-way key-based hash function and the key possessed by the supplicant. In order to verify the supplicant's identity, the authenticator must possess the same key (symmetric key) and the hashed information as the supplicant in order to re-generate the hash value and verify the claimed identity of the supplicant. Another form of hash based non-encryption identity credential uses delayed key disclosure as in TESLA.

Another form of identity-based credential is a shared secret. A shared secret is not necessarily a key. Hence, it will not be used as the basis for any cryptographic operation. One example is root administrators of highly secure machines, who can prove their identity to the authenticator by creating a file in the root directory, which is an operation allowed only to the administrator. Thus, root proves its identity without revealing the password. The secret can be a bit position or any other secret. The authenticator has to challenge the supplicant until the supplicant convinces the authenticator that it knows that secret. This authentication mechanism is called zero knowledge proofs and it can be used in ad hoc networks.

5.2 Context Based Credentials

This category recognizes a unique contextual attribute of the supplicant that can be used to identify it with high confidence. Contextual based credentials can be behavioral or physical. Behavioral-based contextual credentials attempt to identify and authenticate a supplicant based on its pattern of behavior. In this scheme an authenticator would monitor the behavioral pattern of the supplicant with respect to certain functionality and classify it based on its performance. On the other hand, physical-characteristics based contextual credentials attempt to identify and authenticate a supplicant based on a physical characteristic that uniquely identifies it, such as its GPS location, RSSI (Received Signal Strength Indication), or SNR (Signal to Noise Ratio).

The context related credentials depend on the context where the authentication process is performed. We divide this kind of credentials in two subclasses: behavior related and physical data related credentials.

6. CLASSIFICATION BASED ON ESTABLISHMENT OF CREDENTIALS

The first category of authentication protocols under this classification assumes a pre-distribution offline phase (before deployment) where credentials are established. An example of that are pair-wise keys that are pre-distributed to all nodes to be used post deployment for node-to-node authentication. Pre-deployment of credentials is usually employed in symmetric-key-based protocols in SensNets. The second category of authentication protocols assumes that credentials are established post-deployment, such as protocols that rely on contextual information. The third category, like the first one, assumes pre-distribution of initial credentials. However, the actual credentials used for authentication are derived from the initial credentials post deployment.

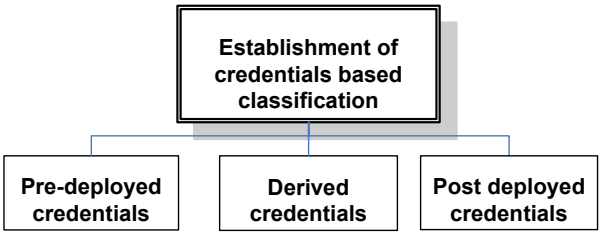


Figure 5 Classification based on establishment of credentials

7. AUTHENTICATION MANAGEMENT ARCHITECTURE

The introduction of wireless-based applications combined with their need for mobility and ubiquity introduced new challenges to conventional authentication approaches. Consequently, new and adapted authentication protocols were developed and customized to best suit the nature of these applications and their underlying networks, including ad hoc networks. As seen from our taxonomy, authentication protocols for ad-hoc networks introduced to date vary significantly with respect to their operating environment, node capability, and network configuration and functionality. An authentication protocol typically describes how the authentication operation is performed in terms of the functions of authentication as described in section 2.1. However, none of the proposed protocols address how the authentication architecture is deployed or managed.

Management of authentication is motivated primarily by the need for enhanced performance and interoperability in today's networks. Given the dynamism of such networks, there are continual changes in the network environment in terms of time, space, and context that affect the authentication operation. Moreover, users' mobility combined with QoS and security

requirements dictate the need for interaction between the different types of autonomous networks that may be used by mobile applications. If not properly managed, the authentication operation might be rendered useless and hence might negatively impact the overall network performance and security.

To further justify the need for authentication management we use a demonstrative simulation study for a flat authentication server deployment model, which assumes that all authentication servers have knowledge of the authentication status of all nodes in the network. Among the factors that affect the performance of the authentication operation are the network traffic load, the number of authentication servers, and their placement. In figure 6, we show a topology that we use to study the effect of these factors on the performance of the authentication operation. The network is a 10X10 grid of nodes in a 500X500 topography. The communication range is set such that each node has 4 neighbors, with the exception of edge nodes that have 3 neighbors, and corner nodes that have 2 neighbors. To study the effect of load over the network, we randomly generate sets of 20, 40, 60, 80, 100, 150, and 200 UDP flows. Before a flow starts, the source and destination nodes should authenticate one another through an authentication server as shown in figure 7. Moreover, to study the effect of increasing the number of deployed servers, we deploy 1, 2, 3, and 4 authentication servers. Furthermore, to study the effect of placement of authentication servers, we experimented with two placement models. The first model places authentication servers in the middle of quadrants as shown in figure 6. The second model places servers at the edges of the network as seen in figure 6. Finally, we compare the flat deployment model used in the above simulations to a hierarchical deployment model, where the authentication status of each node is known to single authentication server.

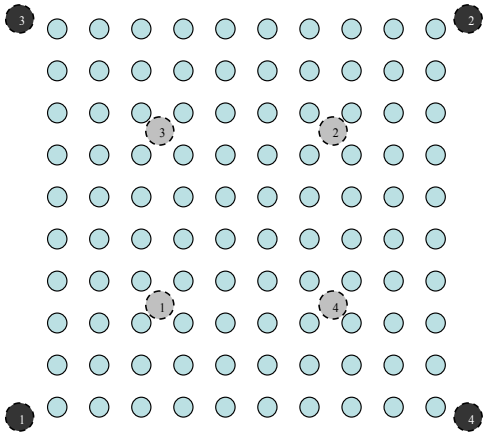


Figure 6 10X10 Grid Topology. First authentication server placement model is shown in gray. Second AS placement model is shown in black.

The performance of the authentication operation is measured in terms of the delay caused by node authentication, while that of the network is measured in terms of packet loss.

7.1 Effect of load

Our simulation results (shown in figures 9 & 10) indicate that the authentication delay increases as the load over the network increases. The results are consistent for both placement models and regardless of the number of authentication servers deployed.

7.2 Authentication of flows

While it is expected that the network performance decreases as we introduce the authentication operation into the network, our simulation results show that the packet loss decreases when authentication of nodes is mandated before a flow starts. This is due to the "backoff" effect of authentication (source and destination of flows are authenticated before flows are allowed in the network). Therefore, the overhead added by authentication may be offset by the benefit of backoff. Figure 13 compares packet loss when authentication is mandated before a flow starts versus when no authentication is required.

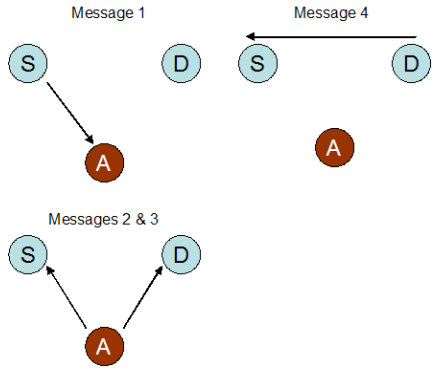


Figure 7 Flat Authentication Model. "S" denotes a source node, "D" denotes a destination node, and "A" denotes an authentication server.

7.3 Number of servers

Intuitively, the objective of increasing the number of authentication servers in the network is to distribute the load over the servers, hence, to enhance the performance of the authentication operation and the performance of the network accordingly. Our simulations show that as we increase the number of authentication servers, the authentication delay is decreased for 20, 40, 60, and 80 flows. This is expected since the replication of authentication servers should distribute the authentication overhead over the servers, which is expected to positively effect the performance of the authentication operation and hence the network performance as a whole. Interestingly, at higher number of flows, these results are reversed showing an increase in delay as the number of authentication severs increases as shown in figure 12. This can be explained as follows. The backoff effect of authentication decreases by increasing the number of servers. Therefore, while the increase in the number of authentication servers tend to decrease the authentication delay due to load distribution, on the other hand, the load on the network increases as a result of having flows start faster. Consequently, this leads to more packets in the network, which may lead to increasing the authentication delay. This is an important result indicating that the increase in the number of servers may not necessarily decrease authentication delay. System administrators need to be mindful of the different

factors involved. Authentication management is therefore needed to optimize the number of active servers under different network conditions.

7.4 Placement of servers

Our results show that the first model of placement of authentication servers reduces the authentication delay compared to the second model as shown in figures 9 & 10. However, the placement of authentication servers within the network mixes authentication traffic with regular traffic within the core of the network. This increases the contention and results in an increased packet loss as compared to the second model, which attempts to place authentication servers outside of the network to push authentication traffic outwards. These results are shown in fig 10. This shows that there is a trade off between authentication delay and packet loss which needs to be considered when authentication servers are placed in the network.

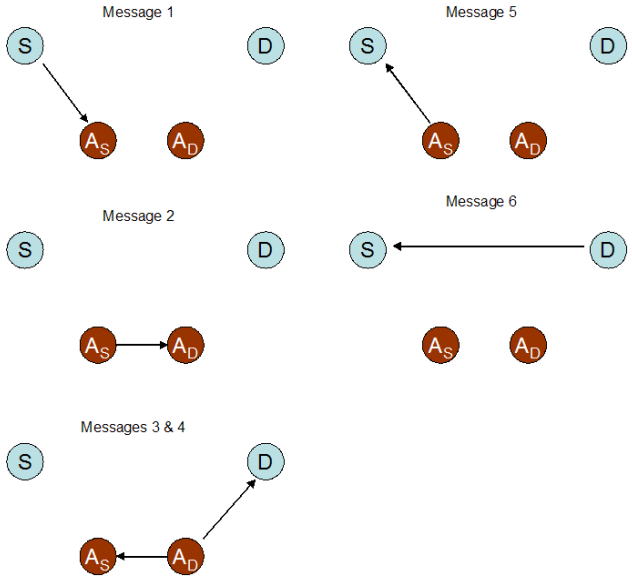


Figure 8 Hierarchical Authentication Model. "S" denotes a source node, "D" denotes a destination node, "AS" denotes an authentication server with whom node "S" is associated, "AD" denotes an authentication server with whom node "D" is associated.

7.5 Hierarchical deployment model

A hierarchical deployment model is a clustering mechanism which associates knowledge of the authentication information of a node to a single authentication server rather than all authentication servers. The goal behind such deployment model is to improve the security of the network by minimizing the impact of a compromised server. A compromised authentication server in a flat deployment model exposes authentication information about all nodes in the network, while a compromised server in a hierarchical deployment model exposes only the nodes associated with that server.

On the other hand, the performance, measured in terms of authentication delay, is decreased in a hierarchical deployment model compared to a flat model. Since knowledge of the authentication status of a node in a hierarchical deployment model is associated with only one authentication server, the authentication model deviates from the one shown for flat deployment shown in figure 7. A node *S* that is trying to authenticate to a node *D* will do so by having the authentication server A_S whom it is associated with contact the authentication server A_D with whom node *D* is associated as shown in figure 8. This results in higher authentication delay. This indicates a tradeoff between security and performance when choosing the appropriate deployment model of an authentication infrastructure. Simulation results for hierarchical deployment model were omitted due to lack of space.

Such scenarios among others motivate the need for an authentication management architecture. An authentication management architecture would overlook the authentication operation and authentication infrastructure within and across domains. Accordingly, such service would pick the appropriate deployment model of authentication infrastructure based on the security and performance requirements. Authentication management architecture would also address the deterioration in performance shown in the above example by placing authentication servers only when the need is justified and when the performance enhancement is guaranteed. Furthermore, the authentication architecture would act on clients' behalf so that functions such as roaming and handoff of clients between authentication services and across authentication domains would be carried out seamlessly.

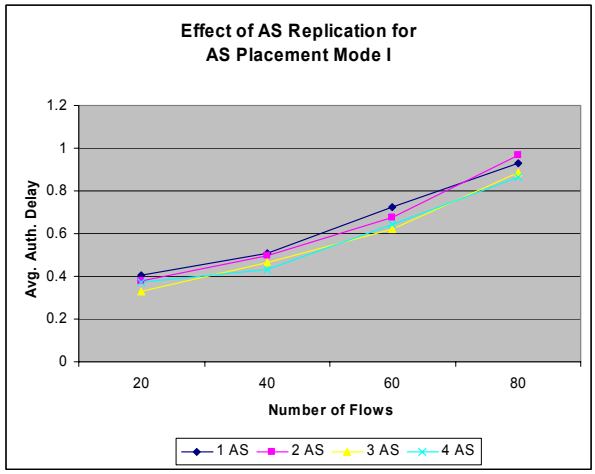


Figure 9 Simulation results showing authentication delay as the number of flows increases from 20-80 flows for 1-4 authentication servers placed using model I. Delay of each set of flows is averaged over 10 simulation runs.

8. CONCLUSIONS AND OPEN RESEARCH ISSUES

We have presented a generic authentication process and developed a taxonomy of authentication protocols. We have also shown through simulations, such as the counterintuitive increase in delay as the number of authentication severs increases for a high number of flows, indicate that an authentication model needs

to be carefully planned for the correct functioning of the authentication operation.

Our current work focuses on developing a formal model for reasoning about the properties of authentication protocols, a unified framework for the quantitative analysis of authentication protocols, and a generic architecture for authentication management. Related open research issues include application-aware optimization of authentication protocols and protocol survivability in presence of different attacks.

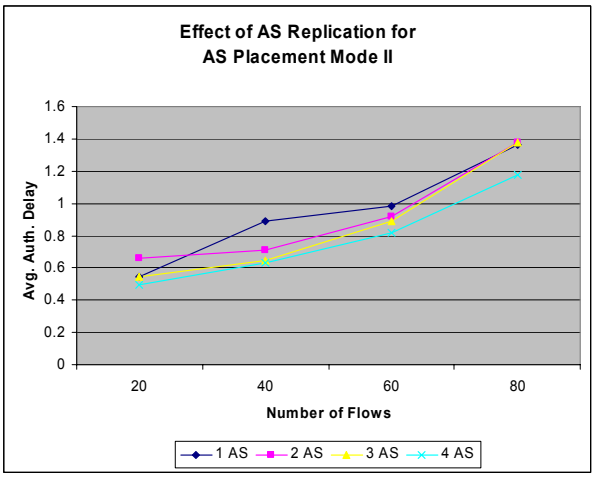


Figure 10 Simulation results showing authentication delay as the number of flows increases from 20-80 flows for 1-4 authentication servers placed using model II. Delay of each set of flows is averaged over 10 simulation runs.

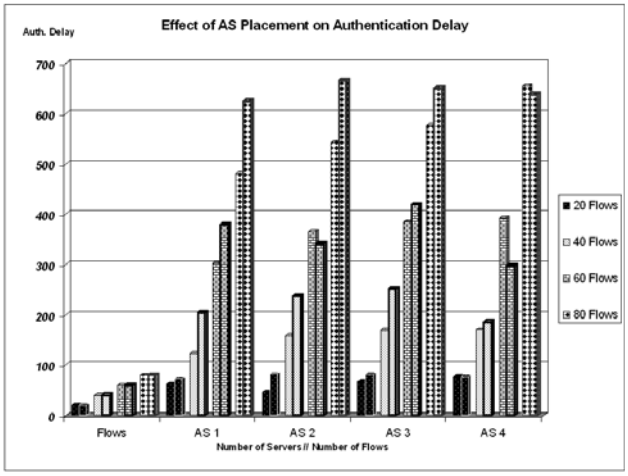


Figure 11 Simulation results comparing packet loss for placement model I & II. The number of flows increases from 20-80 flows for 1-4 authentication servers. Each pair of columns represents a comparison for a set of flows given a number of auth. servers. The left column represents model II and the right column represents model I.

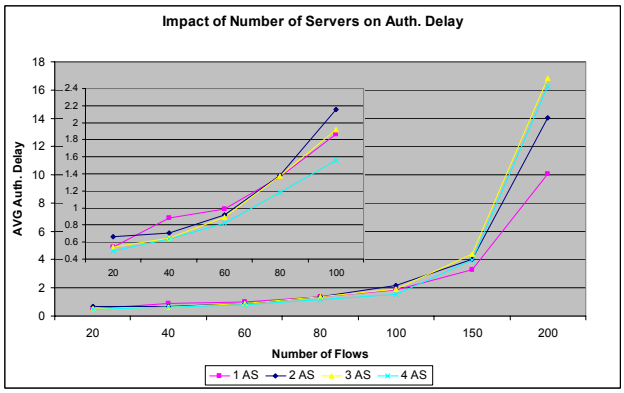
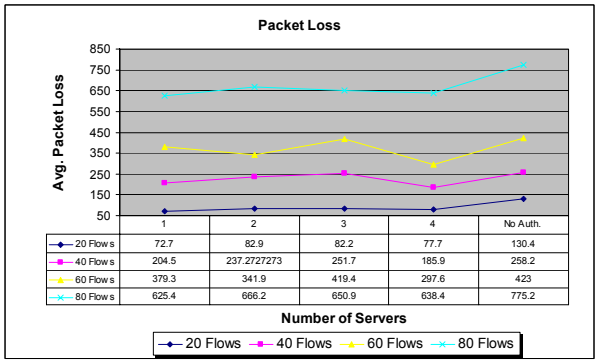


Figure 12 Simulation results showing authentication delay as the number of flows increases from 20-200 flows for 1-4 authentication servers placed using model I. Delay of each set of flows is averaged over 10 simulation runs. Delay for 20-100 flows is magnified in the embedded figure.



	1	2	3	4	No Auth.
20 Flows	72.7	82.9	82.2	77.7	130.4
40 Flows	204.5	237.2727273	251.7	185.9	258.2
60 Flows	379.3	341.9	419.4	297.6	423
80 Flows	625.4	666.2	650.9	638.4	775.2

Figure 13 Simulation results showing packet loss as the number of authentication servers increases 1-4 authentication servers placed using model I. Results also show packet loss when authentication is not required. Packet loss is averaged over 10 simulation runs.

9. REFERENCES

[1] S. Zhu, S. Xu, S. Setia and S. Jajodia, "LHAP: A lightweight hop-by-hop authentication protocol for ad-hoc networks." In Proc. of ICDCS 2003 International Workshop on Mobile and Wireless Network (MWN 2003), May 2003.

[2] A. Weimerskirch and G. Thonet, "A Distributed Light-Weight Authentication Model for Ad-hoc Networks." In Proc. of 4th International Conference on Information Security and Cryptology (ICISC 2001), 6-7 December 2001.

[3] D. Balfanz, D. K. Smetters, P. Stewart and H. Chi. Wong, "Talking to Strangers: Authentication in Ad-Hoc Wireless Networks." In Symposium on Network and Distributed Systems Security (NDSS '02).

[4] L. Venkatraman and D. Agrawal, "A Novel Authentication Scheme for Ad Hoc Networks." In IEEE Wireless Communications and Networking Conference (WCNC 2000), vol. 3, pp. 1268--1273, 2000.

[5] H. Deng, A. Mukherjee, D. P. Agrawal, “Threshold and Identity-Based Key Management and Authentication for Wireless Ad Hoc Networks.” International Conference on Information Technology: Coding and Computing (ITCC'04).

[6] S. Zhu, S. Setia and S. Jajodia, “LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks.” In 10th ACM Conference on Computer and Communications Security (CCS '03).

[7] D. Park, C. Boyd, E. Dawson. “Classification of Authentication Protocols: A Practical Approach.” Proceedings of the Third International Workshop on Information Security.

[8] A Perrig, R. Szewczyk, V. Wen, D. Culler, J.D. Tygar, “SPINS: Security Protocols for Sensor Networks.” In Proc. of ACM Mobicom'01.

[9] Edith C. H. Ngai and Michael R. Lyu, “Trust- and Clustering-based Authentication Services in Mobile Ad Hoc Networks.” In Proc. of 24th International Conference on Distributed Computing Systems Workshops - W4: MDC (ICDCSW'04).

[10] Edith C. H. Ngai and Michael R. Lyu and Roland T. Chin, “An Authentication Service Against Dishonest Users in Mobile Ad Hoc Networks.” In Proc. of 2004 IEEE Aerospace Conference, March 6-13 2004.

[11] F. Stajano and R. Anderson, “The Resurrecting Duckling: Security Issues in Ad-Hoc Wireless Networks.” In M. Roe B. Christianson, B. Crispo, editor, Security Protocols, 7th International Workshop Proceedings, LectureNotes in Computer Science. Springer Verlag, 1999.

[12] IEEE Std 802.11a-1999 Supplement to IEEE standard for information technology telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements. Part 11: wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: high-speed physical layer in the 5 GHz band

[13] A. Weirmerskirch and D. Westhoff, “Identity Certified Authentication for Ad-hoc Networks.” In Proc. of the 1st ACM workshop on Security of ad hoc and sensor networks, 2003, VA USA.

[14] S. Gokhale and P. Dasgupta, “Distributed Authentication for Peer-to-Peer Networks”, In Symposium on Applications and the Internet Workshops 2003 (SAINT'03 Workshops).

[15] R. Yahalom, B. Klein, Th. Beth, “Trust Relationships in Secure Systems- A Distributed Authentication Perspective.” In Proc. of the 1993 IEEE Symposium on Security and Privacy, CA USA.

[16] L. Zhou and Z.J. Haas, “Securing Ad Hoc Networks.” IEEE Network Journal, vol. 13, no. 6,1999, pp. 24-30.

[17] A. A. Pirzada, C. McDonald, “Kerberos Assisted Authentication in Mobile Ad hoc Networks.” Proceedings of the 27th conference on Australasian computer science.

[18] H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang, “Self-Securing Ad Hoc Wireless Networks.” In Seventh IEEE Symposium on Computers and Communications (ISCC '02), 2002.

[19] S. Hahm, Y. Jung, S. Yi, Y. Song, I. Chong, and K. Lim, “A Self-organized Authentication Architecture in Mobile Ad-hoc Networks.” International Conference on Information Networking (ICOIN) 2005.

[20] J. Clark and J. Jacob, “A Survey of Authentication Protocol Literature: Version 1.0”, 17 November 1997. Unpublished article available at http://www.cs.york.ac.uk/~jeremy

[21] S. Basagni and K. Herrin, “Secure pebblenets.” In Proc. of the 2nd ACM international symposium on Mobile ad hoc networking & computing, 2001.

[22] A. Perrig, R. Canetti, J. Tygar, D. Song, “Efficient authentication and signing of multicast streams over lossy channels.” In Proc. of IEEE Symposium on Security and Privacy, May 2000.

[23] Wenliang Du, RonghuaWang, and Peng Ning, “An Efficient Scheme for Authenticating Public Keys in Sensor Networks.” In Proc. of 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), 2005.

[24] M. Bechler, H.-J. Hof†, D. Kraft, F. Pählke, L. Wolf “A Cluster-Based Security Architecture for Ad Hoc Networks” INFOCOM 2004.

[25] J. Binder and H-P. Bischof, “Zero knowledge proofs of identity for ad hoc wireless networks.” 2003.

[26] A. Fritz and J.-F. Pâris, “Maille Authentication: A Novel Protocol for Distributed Authentication.” In Proc. of the 19th IFIP Information Security Conference (SEC 2004), Toulouse, France, Aug. 2004, pages 309–322.

[27] M. Tamer Refaei, V. Srivastava, L. DaSilva, and M. Eltoweissy “A Reputation-based mechanism for Isolating Selfish Nodes in Ad Hoc networks”. In Proc. of the IEEE Mobiquitous 2005, San Diego, CA.