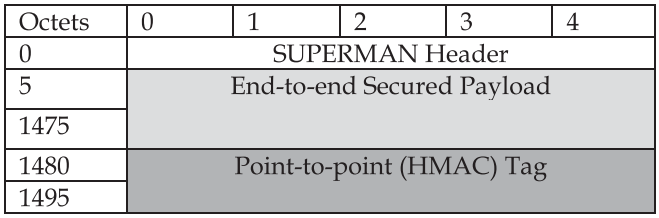

Fig. 6. Example of a SUPERMAN packet using AEAD and HMAC.

3.4.2 Point-to-point Communication

When protected, data is propagated over multiple hops, it is authenticated at each hop. This is achieved using a hashing algorithm, such as HMAC. This is applied to the entire packet to provide point-to-point integrity. A tag is generated using the shared *SKp* of the transmitting node and next hop, which is unique to the direct link in question. The tag is replaced at each intermediate hop, until the destination node is reached. Thus, the authenticity of a route is maintained, as each node on the route must prove their authenticity to the next hop. This tag can also be used for integrity checking.

Fig. 6. shows the structure of a SUPERMAN packet with end-to-end and point-to-point security services. The tag is assumed to be 20 bytes in length, but may be truncated depending on the scenario. A maximum size payload is used in this example.

3.4.3 Broadcast

When a node initiates a broadcast, it uses the broadcast address for the network. Instead of using a *SKe* or *SKp*, which would only function between two nodes, *SKbe* and *SKbp* are used. The packet is secured using the end-to-end and point-to-point methods previously described.

MANET routing protocols require broadcast capabilities. Both OLSR and AODV require broadcast communication for routes discovery. SUPERMAN provides broadcast communication security services to allow it to service the specific needs of MANET routing protocols.

3.5 Summary

SUPERMAN addresses the eight security dimensions detailed by X.805 by providing a closed-MANET, with end-to-end and point-to-point security features. The eight security dimensions are addressed as follows:

- *Access control* is provided by SUPERMAN's network joining method
- *Authentication* is provided by certificates, which allow the relationship between the node and TA to be confirmed
- *Non-repudiation* is provided by timestamps in each SUPERMAN packet header
- *Confidentiality* is provided end-to-end by payload encryption using AEAD
- *Communication* security is maintained by encrypting and performing source authentication end-to-end, and checking authenticity and integrity at each hop
- *Integrity* checking is provided by using a tag for packet integrity

TABLE 3
MATLAB Simulation Parameters

Number of Nodes:	10-100
Routing Algorithm:	Dijkstrka [30] (shortest path)
Number of Iterations:	100
Simulation Area:	100 m x 100 m
Communication Range:	100 m
Max Hop Count:	5
Random Seed:	11
Pseudo-random Number Generation Algorithm:	Mersenne Twister [31]
Key Share Size	128 and 256 bytes
Certificate Size	1,013 and 1,275 bytes

- *Availability* is maintained using each nodes security table, which stores valid authentication credentials. This is combined with the DSKpReq/DSKpRep referral mechanisms to increase availability.
- *Privacy* is provided by end-to-end encryption, with keys that are specific to the link between two nodes or a node and the network.

The next section will present and analyse the results of modelling performed to determine the characteristics of SUPERMAN and its cost in terms of bandwidth, service time and throughput.

4 Methodology and Results

To analyse SUPERMAN, the following key areas were investigated:

- Comparison of security dimension coverage
- Number of communication events required to secure communications between all nodes
- Number of bytes required to secure communications between all nodes
- Overhead of securing communication required for route generation
- Overhead of securing communication required by Consensus Based Bundle Algorithm (CBBA) and Cluster Form CBBA (CF-CBBA)

The eight key security dimensions, outlined in X.805 are evaluated by comparison between SUPERMAN, SAODV, SOLSR, and IPsec/MANIPsec. These are compared in terms of the services provided. This is important because it contextualizes the comparisons of the respective security and communication costs.

These costs represent the additional data or packets (based on the number of communication events) required to provide the security services, referred to from this point as the security overhead.

Overheads are calculated for the network layer of the OSI model. The Datalink and Physical layers of the network stack are not considered as this paper focuses on the network layer (OSI layer 3) specifically.

4.1 Simulation Parameters

All simulation is performed using MATLAB. Table 3 shows the parameters for the simulation environment.

It is assumed that all packets arrive intact without bit-error or loss, and that nodes are stationary during the initialisation and association phases.

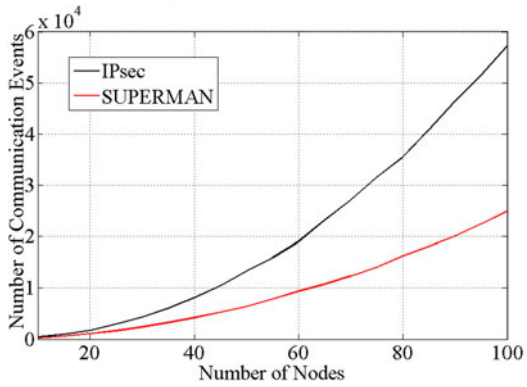

Fig. 7. Graph comparing the number of communication events to secure connections between all nodes under SUPERMAN and IPsec.

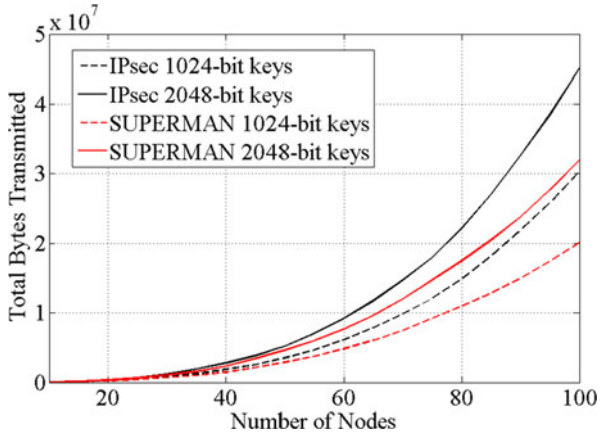

Fig. 8. Graph comparing the number of bytes required to secure connections between all nodes under SUPERMAN and IPsec.

4.2 Initialisation cost of SUPERMAN and IPsec

4.2.1 Method

Comparison of the control overhead required by SUPERMAN and IPsec to initialise a secure network environment allows for the identification of the initialisation costs associated with each approach. These costs may occur throughout the lifetime of the network, but are incurred only when nodes join the network. Two metrics are considered:

- The number of communication events
- The number of bytes transmitted

Both metrics are measured until all nodes in a static set have joined the network.

4.2.2 Results

Fig. 7 compares the number of communication events required to secure all end-to-end connections in a MANET, using SUPERMAN or IPsec. All SUPERMAN nodes have authenticated with the network at this stage, and all IPsec nodes have performed IKE.

The number of communication events represents the total number of messages sent, regardless of packet size. This metric allows one to compare the verbosity of protocols, and comparisons regarding scalability may be made. It also provides data regarding the length of routes, as each relay of a given message will increment the communication event count.

MANETs of 15 nodes require 1,407 events for SUPERMAN and 1,609 for IPsec to form security associations between all nodes. SUPERMAN requires 87 percent of the communication events needed by IPsec, showing immediate gains in security association overhead.

SUPERMAN quickly demonstrates the effectiveness of its referral mechanism, showing itself to be far more scalable than IPsec. A clear trend is shown, in which SUPERMAN more slowly increases in security overhead compared to IPsec. In 100 node simulations, SUPERMAN requires only 42.1 percent of the communication events needed compared with IPsec. This is the result of SUPERMAN node being able to authenticate each other, without reference to a central trusted authority in the field. Pre-initialisation of nodes by a TA implies trusted status when unable to contact the TA directly.

Fig. 8 compares the number of bytes required to secure connections between all nodes in a MANET, using SUPERMAN and IPsec.

SUPERMAN consistently outperforms IPsec in terms of the number of bytes required to secure all nodes in a MANET. For smaller networks, this difference is less pronounced, but for 100 node MANETs, SUPERMAN requires 20.3 megabytes compared to IPsec's requirement of 30.5 megabytes, when using 1,024-bit symmetric keys. SUPERMAN requires only 60 percent of the data required by IPsec to achieve the same outcome, secure communications between all nodes. This trend continues for 2,048-bit keys. SUPERMAN benefits from the cooperative nature of MANETs in both experiments, whereas IPsec requires each node to check in with a coordinator during the authentication process. By allowing nodes to vouch for other nodes that they have already formed secure links with, SUPERMAN reduces the length of routes by not requiring DKSpReq and DKSpRep packets to propagate the full length of the route between source and destination.

4.3 Data Communication Cost of SUPERMAN and IPsec

4.3.1 Method

The MATLAB simulation allows the size of the added communication overhead (number of additional bytes) to be determined. Two scenarios were simulated supported by the parameters outlined in previously in Table 3:

- CBBA task allocation involving 18 nodes
- CF-CBBA task allocation involving 6 clusters of 3 nodes (18 nodes in total)
- Both DTA processes have a task list of between 1 and 50 tasks all of which must be assigned
- In both scenarios is it assumed that all nodes may communicate with each other, over routes that are no longer than the maximum hop count defined for the simulation

4.3.2 Results

Fig. 9 compares the security overhead of SUPERMAN and IPSEC performing CBBA.

The lines in this graph incorporate the noise inherent in the CBBA algorithm. The combination of network size

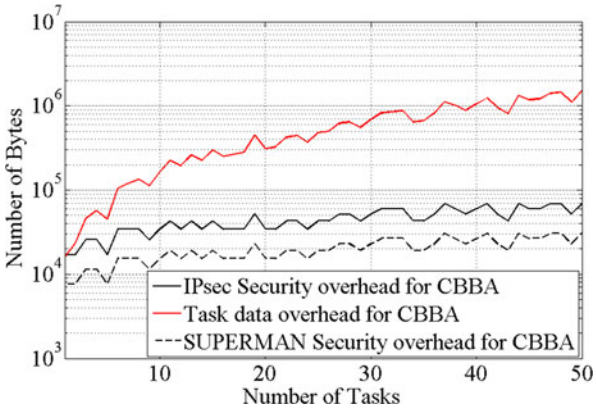

Fig. 9. Chart to compare the number of additional bytes required for the security overhead of IPsec and SUPERMAN when performing CBBA.

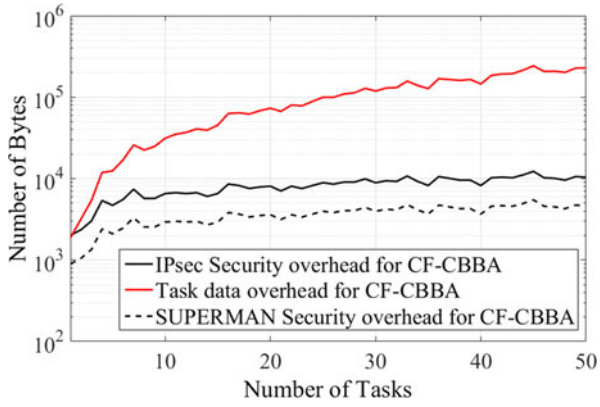

Fig. 10. Chart to compare the number of additional bytes required for the security overhead of IPsec and SUPERMAN when performing CF-CBBA.

and number of tasks can result in the number of CBBA runs required to achieve consensus varying greatly. The value is usually constrained to between 2 and 5 runs of CBBA to reach a solution, and the irregularities in Fig. 9 are a result of higher or lower numbers of runs being required by a given node/task combination. It is not trivial to calculate the number of CBBA rounds, as the number required depends on the size of the network, positions of individual nodes rleatives to tasks and the number of tasks.

The number of bytes required by CBBA is shown to grow rapidly with the size of the CBBA problem domain (the number of nodes and tasks involved). As more nodes are added to the network, the complexity of CBBA communication increases at a cubic rate. IPsec and SUPERMAN add additional security data, requiring that all outbound packets are encapsulated with appropriate headers and tags.

IPsec's overhead is larger than the size (17.1 KB compared with a payload of 15.9 KB). SUPERMAN requires only 7.6 KB of additional data, but this is still 47.7 percent of the size of the payload being protected. This is a result of having assumed the worst case for tag size (20 bytes). Both IPsec and SUPERMAN security overheads reduce in relative size for larger problem domains. For 50 task problems, SUPERMAN requires 30.6 KB and IPsec requires 68.5 KB, to protect a payload of 1.5 MB. SUPERMAN adds approximately 2 percent more data to provide security for this size of problem domain, with IPsec adding 4.5 percent. SUPERMAN requires half of the overhead generated by IPsec to provide the same level of protection to the task allocation process.

SUPERMAN does not require the two IP headers that IPsec needs. As SUPERMAN is integrated at the network layer, it does not re-encapsulate the packet. IPsec encapsulates a payload packet in an IPsec security layer, both of which must have IP headers. By avoiding this redundancy and stripping settings data from its header, SUPERMAN reduces its security overhead by a minimum of 32 bytes per packet.

(1) provides a mathematical expression for the security overhead of CBBA, under a given security framework.

$$x = \frac{(f(c) * (n(n-1))) * (h+t)}{p} \tag{1}$$

The function of c represents the number of rounds required by a given consensus based distributed task allocation algorithm. The number of nodes is represented by n. The header and tag size (are represented by h and t respectively. It is assumed that the payload of a packet will not exceed the Maximum Transmission Unit (MTU) of the network interface. Therefore, header and tag size is only counted once per bundle transmission. Header size includes the IP header when considering protocols that are not integrated into the network stack (e.g., IPsec).

The probability of a packet being delivered is represented by the variable p, which is set to the value of 1 for this investigation, assuming no packet loss in all experiments reported on in this paper. This equation holds true for any non-clustered method of distributing tasks throughout a MANET.

Fig. 10 shows the comparison of SUPERMAN and IPSEC performing CF-CBBA in terms of the number of additional bytes needed to secure data transfer during the DTA process.

IPsec requires 1.8 KB for CF-CBBA communicating a one task problem, compared with 2 KB of data for CBBA. SUPERMAN generates an overhead of 900 bytes for one task CF-CBBA problems. For 50 task problems, SUPERMAN generates security overheads 45 percent the size of IPsec's, while adding only 1.9 percent more data to the bundle exchange process for 50 task CF-CBBA problems. This is driven by the smaller packet size of SUPERMAN.

Equation (2) expands on the previously shown (1), to describe how the security overhead of a given protocol can be derived for CF-CBBA task allocation.

$$y = \left(\sum\nolimits_{1 \le i \le L} x(i)\right) + x(p) \tag{2}$$

The total number of bytes, y, is the product of the sum of all cluster allocation (represented as instances of x). The variable p of x represents the cluster head allocation of CF-CBBA, which is performed prior to pushing the resulting task lists to the cluster level for final allocation among cluster members.

For both CBBA and CF-CBBA, SUPERMAN's smaller packet size reduces the security overhead required. It is notable that security overheads are relatively large for smaller task allocation problems, with larger problems becoming more efficient in terms of the data being protected

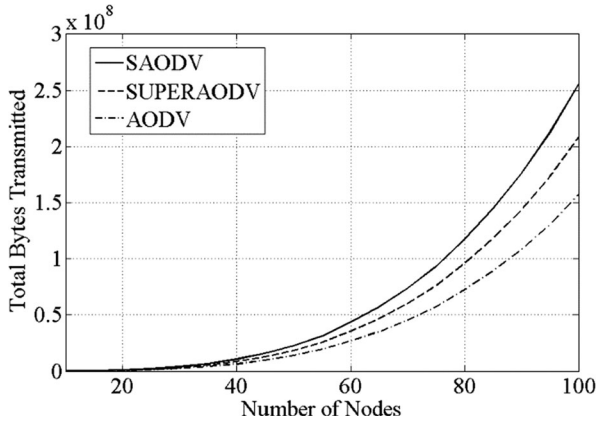

Fig. 11. Chart comparing the number of additional bytes required to secure routing packets using SUPERAODV, SAODV, and AODV.

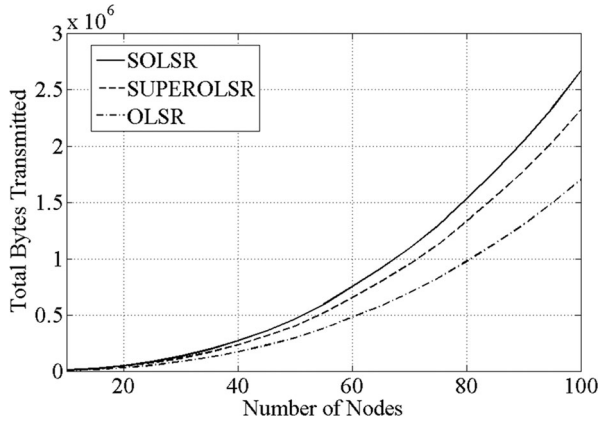

Fig. 12. Chart comparing the number of additional bytes required to secure routing packets using SUPEROLSR, SOLSR, and OLSR.

relative to the data required to provide that protection. This may be mitigated by reducing the size of the tag appended to each packet from 20 bytes to a more manageable size, such as 4 bytes. For this research, the maximum tag size has been chosen to reflect a worst-case scenario and maintain parity with the tag sizes observed for SAODV and SOLSR.

A potential limitation of the lightweight SUPERMAN header is the lack of configuration data. SUPERMAN is not multi-mode, supporting only one mode of security. It is intended as a MANET only security protocol. This means that it lacks the flexibility of VPN protocols, such as IPsec, but provides more efficient, targeted security to MANETs.

4.4 Comparison of Security Overhead in Routing

4.4.1 Method

The additional cost of secure routing is analysed to determine the impact of SUPERMAN on a proactive and reactive MANET protocol. AODV and OLSR, along with their secure implementations, are compared against SUPERMAN secured routing using each protocol. Results have been obtained using a series of MATLAB simulations under the following conditions:

- Simulation parameters outlined in Table 3
- SUPERMAN is applied to OLSR and AODV routing packets
- SOLSR and SAODV are used for comparative analysis
- It is assumed that any pre-routing authentication or first contact handshakes have been performed prior to sending routing packets

The results of these simulations show the number of bytes transmitted during the routing process. Unsecured routing protocols have no security overhead, providing a baseline cost for the routing process. SUPERMAN and secured routing protocols incur this baseline cost, plsu security overhead. The outcome of these simulations will focus on the cost of additional security. Cost of security, in this context, is measured by subtracting the bytes transferred by the secure protocol(s) from the baseline values shown.

4.4.2 Results

Due to the nature of the experiments undertaken in this subsection, a large difference may be perceived between OLSR and AODV. This is due to the experiments focusing on a single instance of routing, in which all nodes in the network form routes with each other.

A single instance of network-wide routing is more demanding for AODV than OLSR (in terms of bytes required to complete the routing operation), but it must be noted that routes will be maintained under AODV until they time out. OLSR, however, will regenerate routes periodically. These results are therefore representative of the total cost for a network wide instance of routing, not the ongoing costs associated with routing on-demand or periodically.

Fig. 11 shows the number of bytes required by AODV, SAODV and SUPERAODV to generate a fully connected set of routes for a network comprised of between 10 and 100 nodes.

AODV provides the cheapest routing with no additional security data or behavioural requirements. In networks of 100 nodes, it requires an average of 73 percent of the bytes required by SUPERMAN protected AODV (SUPERAODV). AODV requires 60.2 percent of the communication required by SAODV.

SUPERAODV does not change the behaviour of AODV, but encapsulates all packets in a SUPERMAN header and tag to provide authentication, confidentiality and integrity to the routing process. SUPERAODV adds a security overhead of 36.9 percent more bytes to AODV, in networks of 100 nodes.

SOADV requires more complex routing behaviour than AODV and SUPERAODV, as well as the addition of header data and a tag to provide security services to the routing process. SAODV generates a security overhead of 66.6 percent more bytes, when compared to AODV in networks of 100 nodes.

Fig. 12 shows the number of bytes required by OLSR, SOLSR and SUPEROLSR to generate a fully connected set of routes for a network comprised of between 10 and 100 nodes.

SUPEROLSR, does not change the behaviour of OLSR but, like SUPEROLSR, it encapsulates routing packets in a secure header and footer (tag). SUPEROLSR requires an additional 40.8 percent of OLSR's byte requirement to provide security to an instance of routing operation performed between 100 nodes.

SOLSR requires 62.3 percent more bytes than OLSR to securely route between 100 nodes. SOLSR requires the

TABLE 4
Security Feature Comparison

Dimensions	Security Protocol			
	SUPERMAN	SOLSR	SAODV	IPsec/MANIPsec
Access Control	X			X
Authentication	X			X
Non-repudiation	X			X
Confidentiality	X			X
Communication Security	X	X	X	X
Data Integrity	X	X	X	X
Availability	X	X	X	
Privacy	X			X

addition of a tag and timestamp to each routing packet, incurring a significant overhead. This does, however, provide critical security services not offered by OLSR, as shown in Table 4 (OLSR provides none of the listed services).

For both AODV and OLSR, SUPERMAN is shown to generate lower overheads by preserving the behaviour of the routing algorithms and providing only the required security features needed to provide authentication, confidentiality and integrity services to the routing process. Mode selection variables and multiple digital signatures are avoided. To provide integrity and authentication services, SUPERMAN only requires a HMAC tag and SUPERMAN header.

The relatively low-cost of SUPERMAN can be ascribed to its use of a closed-network philosophy. By harnessing the control that the owner of a MANET has over the nodes, and the dual end-point/router nature of each node, it is possible to protect routing and application data using a network-stack integrated solution.

SAODV and SOLSR assume a potentially hostile network environment, due to the persistent open-medium problem they are assumed to have to deal with. By closing the network, SUPERMAN can reduce the cost of security by enforcing trustworthiness within the network.

4.5 Security Feature Comparison

SUPERMAN offers a full suite of security services, addressing all eight of the security dimensions outlined in the ITU Rec X.805 document. Table 4 compares the security services of SUPERMAN with SAODV, SOLSR and IPsec. This comparison provides context for the costs seen in the previous results, showing the services provided in return for the additional communication overheads incurred when using SUPERMAN, IPsec or secure routing protocols in a MANET.

IPsec extends seven of eight security services. It does not provide node checking availability services to determine the status of routes and current online members of a network. IPsec does not generally provide route monitoring or point-to-point security service, instead being primarily focused on end-to-end security.

Virtual private Network (VPN) protocols such as IPsec are designed to be adaptable to a variety of networks. They consider the medium itself to be unreliable, and thus focus on the protection of data transmitted over the network, rather than the protection of topology generation and maintenance traffic. This internet-centric design becomes apparent when applied to MANETs, where the vulnerability of the routing protocol can remain a significant threat even when communication security asppplied to application data is being provided by a VPN protocol.

SAODV and SOLSR are designed to secure the routes between nodes, providing protection for end-to-end and point-to-point communication for topology regeneration and route finding only. Data sent along such routes is not secured. The integrity of the route can be enforced, but confidentiality of data packets sent along the route is not.

SUPERMAN provides all eight security services. It is integrated at the network layer, providing lightweight security by avoiding the re-encapsulation process required by IPsec. It protects routing packets, as all packets passing through layer 3 of the network stack are protected. In this way, SUPERMAN provides protection for all data, safeguarding the network and data communicated over it.

In addition to protecting data end-to-end (like IPsec), protection is extended point-to-point, to ensure that the route between source and destination can be trusted. This is achievable due to the small size and direct ownership of MANETs compared to the scale of the Internet, which IPsec is designed to operate on.

MANETs could have thousands of nodes, but they will likely be owned by a single authority. Internet-like networks lack this concept of sole-ownership, making it difficult to implement integrated security solutions. This difficulty when attempting to implement an all-encompassing security solution encourages the use of IPsec and other network-agnostic VPN protocols).

By focusing specifically on securing communication in the context of MANETs, SUPERMAN avoids some of the higher costs associated with VPN approaches which target Internet-like networks. It protects all communication in the network, including routing traffic, protecting against man-in-the-middle attacks. It compares favourably with IPsec and secure routing protocols in terms of security overheads, due to its integration into layer 3 of the network stack.

5 Conclusion

SUPERMAN is a novel security framework that protects the network and communication in MANETs. The primary focus is to secure access to a virtual closed network (VCN) that allows expedient, reliable communication with confidentiality, integrity and authenticity services.

SUPERMAN addresses all eight security dimensions outlined in X.805. Thus, SUPERMAN can be said to implement a full suite of security services for autonomous MANETs. It fulfils more of the core services outlined in X.805 than IPsec, due to being network focused instead of end-to-end oriented.

IPsec is intended to provide a secure environment between two end-points regardless of route, and has been suggested by some researchers to be a viable candidate for MANET security. However, it does not extend protection to routing services. Nor does it provide low-cost security, requiring a lengthy set-up and teardown process, usually on a session basis.

Simulation has been undertaken and the results are reported and analysed to determine the relative cost of

security for SUPERMAN, compared against IPsec, SAODV and SOLSR where relevant.

SUPERMAN provides a VCN, in which the foundation-block of security is provided by authenticating nodes with the network. This enables further benefits, such as the security association referral and network merging. It also provides a relatively light-weight encapsulation packet and variable length tag.

Under both CBBA and CF-CBBA, the security overheads of SUPERMAN have been demonstrated to be lower than those of IPsec. Both DTA algorithms represent how a MANET can be made autonomous, by allowing problem solving without human intervention to occur on the network. Securing the communication required to facilitate this functionality is a critical consideration when providing a fully secured network. By providing lower cost security than existing alternatives, while providing security across all eight security dimensions, SUPERMAN proves it is a viable and competitive approach to securing the communication required by autonomous MANETs.

SUPERMAN has been shown to provide lower-cost security than SAODV and SOLSR for their respective routing protocols. By establishing a secure, closed network; one can assume a certain level of trust within that network. This reduces the need for costly secure routing behaviours designed to mitigate the effects of an untrusted environment (and untrusted nodes) on the routing process. By preventing the entry of potentially untrustworthy nodes to the network, and thus the routing process, a MANET may be protected from subversion of its routing services at a lower cost, as malicious nodes are barred from the process entirely.

SUPERMAN provides security to all data communicated over a MANET. It specifically targets the attributes of MANETs, it is not suitable for use in other types of network at this time. It sacrifices adaptability to a range of networks, to ensure that MANET communication is protected completely and efficiently. A single efficient method protects routing and application data, ensuring that the MANET provides reliable, confidential and trustworthy communication to all legitimate nodes.

Future work includes the implementation of SUPERMAN [32] on a simple mobile node platform to allow experimental observation and profiling of its performance, the proposal of network bridging solutions capable of providing SUPERMAN services between two closed networks over an insecure intermediate network, and investigating the effects of variable network topology on SUPERMAN to better understand the role of the credential referral mechanism on overhead mitigation in SUPERMAN networks.

REFERENCES

[1] P. S. Kiran, "Protocol architecture for mobile ad hoc networks," in *Proc. IEEE Int. Ad. Comput. Conf.*, 2009, pp. 2112–2117.

[2] A. Chandra, "Ontology for manet security threats," in *Proc. 2nd Nat. Conf. Netw. Eng.*, 2005, pp. 171–117.

[3] A. K. Rai, R. R. Tewari, and S. K. Upadhyay, "Different types of attacks on integrated manet-internet communication," *Int. J. Comput. Sci. Secur.*, vol. 4, no. 3, pp. 265–274, 2010.

[4] D. Smith, J. Wetherall, S. Woodhead, and A. Adekunle, "A cluster-based approach to consensus based distributed task allocation," in *Proc. 22nd Euromicro Int. Conf. Parallel, Distrib. Netw.-Based Process.*, 2014, pp. 428–431.

[5] I. D. Chakeres and E. M. Belding-Royer, "Aodv routing protocol implementation design," in *Proc. 24th Int. Conf. Distrib. Comput. Syst. Workshops*, 2004, pp. 698–703.

[6] T. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSR)," RFC 3626, Oct. 2003, Doi: 10.17487/RFC3626.

[7] M. Hyland, B. E. Mullins, R. O. Baldwin, and M. A. Temple, "Simulation-based performance evaluation of mobile ad hoc routing protocols in a swarm of unmanned aerial vehicles," in *Proc. 21st Int. Conf. Adv. Inf. Netw. Appl. Workshops*, 2007, vol. 2, pp. 249–256.

[8] J. Pojda, A. Wolff, M. Sbeiti, and C. Wietfeld, "Performance analysis of mesh routing protocols for uav swarming applications," in *Proc. 8th Int. Symp. Wireless Commun. Syst.*, 2011, pp. 317–321.

[9] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang, "Security in mobile ad hoc networks: Challenges and solutions," *IEEE Wireless Commun.*, vol. 11, no. 1, pp. 38–47, Feb. 2004.

[10] N. Garg and R. Mahapatra, "Manet security issues," *Int. J. Comput. Sci. Netw. Secur.*, vol. 9, no. 8, pp. 241–246, 2009.

[11] W. Ivancic, D. Stewart, D. Sullivan, and P. Finch, "An evaluation of protocols for uav science applications," NASA Technical Reports (NTRS), 2011 Earth Science Technology Forum (ESTF2011, Jun. 2011.

[12] A. R. McGee, U. Chandrashekhar, and S. H. Richman, "Using itu-t x. 805 for comprehensive network security assessment and planning," in *Proc. 11th Int. Telecommun. Netw. Strategy Planning Symp.*, 2004, pp. 273–278.

[13] M. G. Zapata, "Secure ad hoc on-demand distance vector routing," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 6, no. 3, pp. 106–107, 2002.

[14] F. Hong, L. Hong, and C. Fu, "Secure OLSR," in *Proc. 19th Int. Conf. Adv. Inf. Netw. Appl.*, 2005, vol. 1, pp. 713–718.

[15] A. Hafslund, A. Tønnesen, R. B. Rotvik, J. Andersson, and Ø. Kure, "Secure extension to the OLSR protocol," presented at the OLSR Interop Workshop, San Diego, CA, USA, 2004.

[16] R. H. Jhaveri, S. J. Patel, and D. C. Jinwala, "Dos attacks in mobile ad hoc networks: A survey," in *Proc. 2nd Int. Conf. Adv. Comput. Commun. Technol.*, 2012, pp. 535–541.

[17] S. Maity and S. K. Ghosh, "Enforcement of access control policy for mobile ad hoc networks," in *Proc. 5th Int. Conf. Secur. Inf. Netw.*, 2012, pp. 47–52.

[18] D. Hurley-Smith, J. Wetherall, and A. Adekunle, "Virtual closed networks: A secure approach to autonomous mobile ad hoc networks," in *Proc. 10th Int. Conf. Internet Technol. Secured Trans.*, 2015, pp. 391–398.

[19] S. Bhattacharya and T. Basar, "Game-theoretic analysis of an aerial jamming attack on a uav communication network," in *Proc. Amer. Control Conf.*, 2010, pp. 818–823.

[20] S. Lu, L. Li, K.-Y. Lam, and L. Jia, "Saodv: A manet routing protocol that can withstand black hole attack," in *Proc. Int. Conf. Comput. Intell. Secur.*, 2009, vol. 2, pp. 421–425.

[21] S. Zhao, R. Kent, and A. Aggarwal, "A key management and secure routing integrated framework for mobile ad-hoc networks," *Ad Hoc Netw.*, vol. 11, no. 3, pp. 1046–1061, 2013.

[22] M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams, "X. 509 internet public key infrastructure online certificate status protocol-OCSP," *RFC 2560*, Jun. 1999, Doi: 10.17487/RFC2560.

[23] N. Doraswamy and D. Harkins, *IPSec: The New Security Standard for the Internet, Intranets, and Virtual Private Networks*. Upper Saddle River, NJ, USA: Prentice Hall Professional, 2003.

[24] A. Ghosh, R. Talpade, M. Elaoud, and M. Bereschinsky, "Securing ad-hoc networks using IPSEC," in *Proc. IEEE Mil. Commun. Conf.*, 2005, pp. 2948–2953.

[25] K. N. Ali, M. Basheeruddin, S. K. Moinuddin, and R. Lakkars, "Manipsec-ipsec in mobile ad-hoc networks," in *3rd IEEE Int. Conf. Comput. Sci. Inf. Technol.*, 2010, vol. 1, pp. 635–639.

[26] E. Rescorla, "Diffie-hellman key agreement method," *RFC 2631*, Jun. 1999, Doi: 10.17487/RFC2631.

[27] L. Harn, M. Mehta, and W.-J. Hsin, "Integrating diffie-hellman key exchange into the digital signature algorithm (DSA)," *IEEE Commun. Lett.*, vol. 8, no. 3, pp. 198–200, Mar. 2004.

[28] H. Krawczyk and P. Eronen, "Hmac-based extract-and-expand key derivation function (HKDF)," *RFC 5869*, May 2010, Doi: 10.17487/RFC5869.

[29] A. Adekunle and S. Woodhead, "An aead cryptographic framework and tinyaead construct for secure wsn communication," in *Proc. Wireless Adv.*, 2012, pp. 1–5.
[30] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
[31] M. Matsumoto and T. Nishimura, "Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Trans. Modeling Comput. Simul.*, vol. 8, no. 1, pp. 3–30, 1998.
[32] An open-source implementation of SUPERMAN is in development consisting of a Linux Kernel Module and Daemon. (2016). [Online]. Available: https://bitbucket.org/wj88/superman/

Darren Hurley-Smith received the BEng (Hons) degree in computer systems and software engineering from the University of Greenwich, in 2012, and the PhD degree from the University of Greenwich, in 2015. Currently, he is a postdoctoral research associate with the University of Kent's School of Computing.

Jodie Wetherall received the BEng (Hons) degree in 2001, and the PhD degree in 2010. He is currently a principal lecturer in the Faculty of Engineering and Science, University of Greenwich. His research interests include MANETs, security, scheduling, and automation.

Andrew Adekunle received the BENg (Hons.) and PhD degrees in network security. He is currently a lecturer in the Faculty of Engineering and Science, University of Greenwich. His research interests include network security and embedded networked systems.