Postprint of Secure Terminal Access Scheme for Mobile Tactical Environments

Authors: Qin Qiuyang, Jiang Lingyun

Date: 2020-09-28T00:00:00+00:00

Abstract

To address the challenge that traditional security authentication mechanisms fail to achieve secure terminal access in search, rescue, and military combat environments due to harsh communication conditions and frequent terminal mobility, this paper proposes a terminal secure access scheme for mobile tactical environments. The scheme adopts a certificateless key management mechanism, and analyzes the security authentication process following terminal mobility, as well as the security handling methods when terminals and gateways are compromised or intruded. Simulation results indicate that the proposed scheme enhances the security of authorization and authentication between gateways and terminals, effectively resists known attacks, resolves the issues of lacking mutual authentication and key escrow in tactical environments, and the certificateless key algorithm employed in this scheme provides superior security and reduced computational overhead compared to alternative algorithms, thereby enabling a trade-off between security and energy consumption for gateway access during terminal mobility.

Full Text

Preamble

Vol. 38 No. 3
Application Research of Computers
ChinaXiv Partner Journal

Secure Access Scheme for Terminals in Mobile Tactical Environments

Qin Qiuyang, Jiang Lingyun†
(College of Telecommunications & Information Engineering, Nanjing University of Posts & Telecommunications, Nanjing 210003, China)

Abstract: In search, rescue, military operations, and similar environments, poor communication conditions and frequent terminal mobility make it difficult for traditional security authentication systems to achieve secure terminal access. To address this problem, this paper proposes a terminal secure access scheme for mobile tactical environments. The scheme employs a certificateless key management mechanism and analyzes the security authentication process after terminal movement, as well as secure handling methods after terminals and gateways are compromised or invaded. Simulation results demonstrate that the scheme enhances the security of authorization and authentication between gateways and terminals, effectively resists known attacks, solves the problems of lacking mutual authentication and key escrow in tactical environments, and the certificateless key algorithm used in this scheme offers better security with lower computational overhead compared to other algorithms, enabling a trade-off between security and energy consumption during terminal mobility.

Keywords: tactical environment; secure terminal access; certificateless key management; Internet of Things security

0 Introduction

Compared with traditional Internet, the Internet of Things (IoT) extends the cyber world into the physical world, introducing more complex security challenges. IoT faces numerous security threats including device resource constraints, high terminal mobility, diverse attack vectors, and lack of standardized authentication, making it more difficult to ensure message confidentiality, integrity, secure authentication processes, and user privacy. While some existing security issues in civilian IoT systems have been effectively addressed through solutions such as key authentication systems, data tagging, IP traceback, and Software-Defined Networking (SDN)-based security schemes, these systems typically assume stable network connectivity. Tactical environments present far more complex scenarios with unstable network connections.

In tactical environments such as search, rescue, and military operations, obstacles, enemy interference, and long distances to the cloud cause significant network signal fluctuations, requiring operation in Disconnected, Intermittent, and Limited (DIL) network environments. Additionally, tactical environments may feature dynamic contexts, more limited computational resources, and compromised or invaded terminals and gateways, necessitating constant monitoring of security status and timely effective responses. Terminals in tactical environments exhibit higher mobility, requiring efficient and secure migration of authorized terminal verification information to edge gateways that the terminal is about to access during movement.

Existing terminal secure access schemes for tactical environments are insufficiently comprehensive, primarily focusing on terminal-to-gateway authorization and re-authentication after gateway information migration, without detailed

consideration of solutions for compromised or invaded terminals and gateways. These schemes also retain security vulnerabilities, such as lacking mutual verification during terminal access and absence of trust escrow. Certificateless key algorithms effectively address these security issues. First proposed by Riyami et al. and subsequently optimized, these algorithms can now better resist various known attacks.

1 Related Work

Numerous scholars have proposed terminal secure access schemes and algorithms. In conventional environments, literature [11] presented a secure authentication scheme between terminals and servers based on Elliptic Curve Cryptography (ECC) [12], which literature [13] later identified as lacking mutual authentication and subsequently improved. Literature [14] further identified security issues in the improved scheme and continued to refine it. However, these algorithms are based on normal IoT environments and differ significantly from verification schemes in tactical environments.

In complex tactical environments, literature [15] combined Identity-Based Cryptography (IBC) [16,17] with Trusted Third Party (TTP)-free [18] secure key protocols to achieve secure terminal access to edge gateways when nodes lose connection with the cloud. However, this approach and subsequent literature [19] cannot guarantee trust escrow or mutual authentication between terminals and gateways. Moreover, these works employ identical authentication schemes that require gateways to fully trust other gateways, creating significant security vulnerabilities.

In dynamic networks, Seung et al. proposed certificateless key management schemes [20,21] that address the trust escrow problem mentioned above, enabling mutual verification between nodes during key formation and enhancing resistance to known attacks. However, these schemes inadequately address node mobility and lack detailed handling procedures for compromised or invaded nodes.

To address these limitations, this paper continues using the tactical environment from literature [15], introduces the concept of a master control node, and combines certificateless key management schemes to solve the problems of lacking trust escrow and mutual authentication. We provide a detailed analysis of the security authentication process after terminal movement in dynamic tactical networks and discuss handling methods for compromised or invaded terminals and gateways.

2 Tactical Environment Network Topology

The network topology for the tactical environment in this paper is shown in [Figure 1: see original paper]. Due to the complex conditions and poor communication quality in tactical environments, nodes may lose connection with the

cloud at any time. In our tactical model, when nodes disconnect from the cloud, three main types of nodes exist in the network: terminals, edge gateways, and master control nodes.

In specific tactical environments, soldiers or machines such as unmanned vehicles and drones carry multiple sensors, actuators, and an Android client. This Android client serves as the terminal, connecting to all devices for data collection and simple processing. Terminals continuously move and connect to edge gateways, obtaining additional services while receiving commands from gateways. Arrows in [Figure 1: see original paper] indicate terminal movement directions; they can move within the signal range of their connected gateway or transition to another gateway' s range.

Gateways are computing devices with strong data processing and signal reception capabilities, carried by fixed personnel in each squad or equipped on larger unmanned vehicles or drones. Each gateway in [Figure 1: see original paper] has its own signal coverage range and connects to the master control node via single-hop or multi-hop communication. Since gateways have smaller movement ranges and less frequent movement compared to terminals, and gateway movement can be equivalently treated as relative terminal movement, this paper only considers terminal movement.

Among all gateways, one with large processing capacity and low mobility is selected as the master control node, such as a temporary command post or large combat vehicle. The master control node generally remains stationary or has minimal mobility, serving three primary functions: (1) generating system parameters for all nodes before deployment for future mutual verification and pairing key formation; (2) utilizing robust intrusion detection mechanisms during operation to identify compromised or invaded nodes; and (3) aggregating all data and issuing commands to other nodes.

3 Certificateless Key Management Based Terminal Access Scheme

In DIL tactical environments, terminals and gateways must form pairing keys and verify each other' s trustworthiness. As terminals continuously move, they transition from one gateway' s signal coverage to another' s. After movement, terminals connect to new gateways, requiring secure migration of processed information from the original gateway. Furthermore, due to the complexity of DIL tactical environments, communication interruptions, terminal damage, and terminals moving out of gateway signal range without notification may occur. This section details all these scenarios and proposes handling methods.

3.1 Mutual Verification Between Nodes

Mutual verification occurs in three scenarios: terminal-to-terminal, terminal-to-gateway, and gateway-to-gateway. Since these verification methods are similar,

this section focuses on establishing mutual trust between terminals and gateways. Secure terminal access to gateways involves two steps: first, terminals and gateways form pairing keys while mutually verifying each other; second, gateways generate BLS certificates [22] using device IDs sent by terminals, which facilitate verification when terminals later access other gateways.

3.1.1 Generation of Public Key, Private Key, and Individual Key Based on certificateless key management, each node must receive system parameters before deployment in the tactical environment. The Key Generation Center (KGC) of the master control node generates system parameters $\Omega = \{$ $\cdots\}$ and a completely confidential master key x. (The specific process refers to literature [21].)

The master control node publishes Ω and keeps x confidential. It assigns a unique identifier A to each gateway and terminal. Terminals connect to gateways via wireless connections such as Bluetooth or Wi-Fi. Using the system parameters and identifiers provided by the master control node, terminals and gateways generate their respective public and private keys.

Since the key generation steps are identical for terminals and gateways, we only describe the terminal' s key generation process. [Figure 2: see original paper] illustrates the complete generation process of a terminal node A' s private key, public key, and individual key.

3.1.2 Generation of Pairing Keys After all terminals and gateways generate their private keys, public keys, and individual keys, they are deployed to the tactical environment. Upon deployment, all communicable nodes must generate pairing keys. However, terminals can typically only connect to one gateway simultaneously, thus forming pairing keys with only one gateway at a time.

Any terminal or gateway broadcasts its identifier and public key. When node A receives a broadcast from node B, they establish a long-term master pairing key K_{AB} and an encryption pairing key k_{AB} through the process shown in [Figure 3: see original paper].

Since only nodes possessing system parameters can complete these steps, this scheme both forms pairing keys for encrypted communication between any two nodes and verifies the trustworthiness of nodes establishing keys. Additionally, because master pairing key generation requires ECC algorithms that constitute the main energy consumption in certificateless key schemes, but maintaining constant pairing keys may impact security, encryption keys can be periodically updated while keeping master pairing keys unchanged.

3.1.3 Generation of BLS Certificates After initial verification, terminals provide their device ID as a public key (literature [6,15] uses obtainable Android device IDs randomly generated before node deployment). Gateways generate BLS certificates [22] for terminals using the terminal' s device ID, gateway' s

device ID, and private key. These certificates are stored in the gateway, which then sends the BLS certificate and gateway device ID to the terminal.

During terminal-gateway communication, both parties encrypt and transmit messages using encryption keys. To enhance security, terminals must send their device ID and BLS certificate before requesting services. After verifying the BLS certificate, the gateway sends its device ID for terminal verification.

Through these two steps, complete trust is established between terminals and gateways. Gateways use individual keys to send messages to the master control node, reporting terminal information. Gateway-to-gateway trust establishment follows the same verification process. For terminal-to-terminal trust, since terminals do not need to provide services to each other, only pairing keys are required without BLS certificate generation.

3.2 Security Authentication After Node Movement

In tactical environments, terminals constantly change positions while gateway signal coverage is limited. When terminals move out of their current gateway' s signal range, they cannot continue communication and must connect to other gateways for services. However, establishing complete trust relationships is complex and energy-consuming. Therefore, after movement, the new gateway can communicate with the original gateway to request terminal information, reducing energy consumption for re-authentication. For convenience, we denote the terminal' s current gateway as CA and the post-movement gateway as CB.

3.2.1 Terminal and Gateway Disconnection Terminal movement involves two scenarios: actively submitting a leave request to CA, or unexpectedly disconnecting without warning. The handling procedures are shown in [Figure 4: see original paper].

Unexpected Disconnection: Due to complex tactical environments with poor signals or compromised/damaged/lost terminals, disconnection may occur without gateway notification. A large time threshold Tm is preset. If a terminal fails to connect to a new gateway within Tm (i.e., CA receives no request from other gateways for the terminal' s information, as described in Section 3.2.2, or the terminal does not reconnect to CA), the terminal is deemed completely compromised or invaded, and CA reports this to the master control node. If the terminal reconnects to CA within Tm (possibly due to intermittent connectivity from poor communication), a shorter threshold t is preset. If disconnection duration $\leq$ t, CA only requests the terminal' s BLS certificate and regenerates the encryption key. If disconnection $>$ t, the terminal must re-establish complete pairing keys with CA. After reconnection, CA submits event information to the master control node. If the terminal connects to another gateway within Tm, refer to Section 3.2.2.

Active Leave Request: If a terminal actively submits a leave request to CA but fails to connect to a new gateway within Tm, it is deemed compromised or

invaded, and CA reports this. If the terminal reconnects to CA within Tm, the disconnection duration is checked against preset threshold Tn (Tm > Tn). If exceeded, complete pairing keys must be regenerated; otherwise, only encryption keys are generated and BLS certificates are verified. If the terminal needs to connect to another gateway within Tm, refer to Section 3.2.2.

After each successful verification, gateways must submit information about newly connected terminals to the master control node using individual keys. If verification fails, the master control node must be informed.

3.2.2 Terminal Connecting to Gateway After Movement When a terminal actively submits a leave request to CA via encryption key, CA records the terminal' s status, certificates, keys, and data in a database and reports the departure to the master control node. Upon entering CB' s signal coverage, the terminal applies for connection. To save time and reduce energy consumption, verification can be completed through the process shown in [Figure 5: see original paper].

The specific steps are: The terminal wirelessly connects to CB and sends CA' s identifier. CB communicates with CA via encryption key (prerequisite: CB and CA have established complete trust and can communicate), requesting all relevant terminal information. If the terminal has never connected to CB before (no prior pairing keys), complete pairing keys must be formed, and CB generates a new BLS certificate, sends it with gateway device ID to the terminal, and reports terminal access to the master control node. If the terminal previously communicated with CB and actively disconnected, the time since disconnection is checked against system threshold T. If exceeded, pairing keys are regenerated; otherwise, encryption keys are generated from the existing master pairing key, the terminal provides its BLS certificate for CB verification (CB already obtained the certificate from CA), and CB sends its device ID to the terminal, enabling service requests. Upon successful verification, CB reports terminal information to the master control node. If verification fails, the master control node must be notified.

3.2.3 Edge Gateway Compromise If an edge gateway CA is compromised or invaded, all connected terminals must connect to other gateways. With strong computational capabilities and intrusion detection mechanisms, the master control node detects CA' s compromise and instructs all connected terminals to disconnect immediately and connect to alternative gateways.

Due to tactical environment complexity, the master control node may be unable to deliver CA compromise messages via individual keys to all connected terminals. The message can be propagated through the process shown in [Figure 6: see original paper].

The master control node sends CA compromise messages via individual keys to all communicable gateways and terminals. Terminals connected to CA immedi-

ately disconnect upon receiving the message. Gateways receiving the message forward it via encryption keys to all connected terminals, which further propagate it to other connected terminals. Terminals receiving two or more CA compromise messages immediately disconnect. After disconnecting, terminals needing new services follow the verification process described in Section 3.1.

4 Security Analysis and Performance Analysis

This section analyzes the security and performance of our scheme. Security analysis demonstrates resistance to known attacks, solves previous tactical environment terminal access problems, and compares our algorithm' s security requirements with other certificateless key algorithms [22,23]. Performance analysis compares our algorithm's time and computational advantages, and discusses the trade-off between security and energy consumption.

4.1 Security Analysis

a) Compromise-Resilience: If an attacker captures a node and obtains all its keys and certificates, they cannot acquire other nodes' keys because pairing keys between any two nodes are independent. Due to ECC complexity, attackers cannot determine the master control node' s master private key x, preventing impact on other nodes.

b) Resistance to Clone and Impersonation Attacks: If an attacker captures a node for cloning attacks, extracting its keys and replicating it in another domain, the master control node' s intrusion detection identifies the compromised node and notifies all nodes. For compromised gateways, refer to Section 3.2.3; for compromised terminals, the master control node informs the connected gateway to disconnect and reports back.

For impersonation attacks where attackers insert nodes to obtain BLS certificates, pairing algorithm advantages prevent key establishment. An impersonating node X attempting to connect with legitimate node A will fail because A verifies legitimacy through equations that require system parameters Ω from the master control node. Only nodes receiving Ω can compute valid parameters and decrypt messages. Furthermore, X cannot generate correct HMAC values without knowing the secret keys, preventing successful impersonation.

c) Resistance to Known-Key Attacks: Known-key attacks where attackers obtain encryption keys to extract master pairing keys are prevented by HMAC' s one-way property. Without master keys, attackers cannot derive new encryption keys when nodes periodically update them.

d) Mutual Verification: Both verification steps involve mutual authentication. The pairing key establishment process verifies both nodes, as detailed in the clone/impersonation resistance analysis. Only nodes with system parameters Ω from the master control node can pass authentication and form keys.

During wireless connection, terminals receive BLS certificates and gateway device IDs, which are verified during each service request.

e) Key Escrow Solution: System parameters sent by the master control node are not complete certificates. Terminals and gateways generate their own master private keys and compute complete public/private key pairs using Ω. ECC complexity prevents any node from obtaining other nodes' master private keys or complete private keys.

compares the security of our algorithm with two other certificateless key algorithms [22,23], where "√" indicates the algorithm meets the security requirement and "×" indicates it does not.

4.2 Performance Analysis

We compare our algorithm's performance with two other certificateless key management algorithms. Computational overhead is compared at 160-bit security level (where Tp represents ECC scalar multiplication time), with results shown in .

Implementation uses the MIRACL big number library in C language, running on a 64-bit Windows OS. After 50 operations, the average Tp is 1.03ms. [Figure 7: see original paper] shows the average time consumption of different algorithms.

Communication costs are also compared, primarily including elliptic curve size (e bits) and message size (m bits). Since all three algorithms are ECC-based certificateless key algorithms, communication cost is 3e+m for each. Our algorithm' s communication cost is unchanged, but its computational overhead is 50% less than literature [22] and comparable to literature [23], while providing better security than literature [23].

4.3 Security and Energy Consumption Trade-off

By setting different values for time threshold T and considering various average speeds, [Figure 9: see original paper] shows the number of master pairing key formations required for 100 terminals within one day.

For different T values with constant terminal speed, daily encryption key generation frequency approaches a constant value consistent with master pairing key generation when T=0. shows encryption key generation frequencies for four different speeds.

During movement, terminals enter new gateway signal ranges and request connection. If a terminal previously established pairing keys with a gateway and the time between active disconnection and reconnection $\leq$ T, only encryption keys need regeneration from the existing master pairing key. If reconnection time $>$ T, complete pairing keys must be regenerated.

Larger T values reduce master pairing key generation frequency and energy consumption but decrease security. Smaller T values increase generation frequency,

energy consumption, and security. [Figure 8: see original paper] shows master key update frequency decreases rapidly within 0-500s and stabilizes after 500s. To ensure security while minimizing energy consumption and time waste, T is typically set to 500s.

5 Conclusion

Existing tactical environment security authentication schemes operate under the premise of disconnected core networks. Without a cloud-based, computationally powerful, fully trusted third party, conventional terminal trusted access algorithms are difficult to implement. Our certificateless key management-based secure access scheme effectively solves previous problems of lacking mutual authentication and trust escrow. We detailed various post-movement scenarios and handling methods, enabling trade-offs between energy consumption and security based on actual conditions. However, limitations remain: the random walk mobility model does not fully reflect terminal movement in tactical environments. Future work should consider mobility models for different scenarios combined with appropriate algorithms to precisely calculate T values for various environments. Additionally, our compromised node detection relies primarily on the master control node' s intrusion detection mechanism; future work could incorporate mutual monitoring between nodes to promptly identify compromised or invaded nodes.

References

[1] Sha Kewei, Wei Wei, T. Yang A, et al. On security challenges and open issues in Internet of Things [J]. Future Generation Computer Systems, 2018, 83: 326-337.

[2] Kouicem D E, Bouabdallaha A, Lakhlef H. Internet of things security: A top-down survey. [J]. Computer Networks, 2018, 141: 199–221.

[3] Marianne R, Brannsten. Federated Single Sign On in Disconnected, Intermittent and Limited (DIL) Networks [C]// 81st IEEE Vehicular Technology Conference, Piscataway, NJ: IEEE Press, 2015: 1-5.

[4] Echeverr S, Lewis G A, Root J. Cyber-Foraging for Improving Survivability of Mobile Systems [C]// 34st Anuual IEEE Military Communications Conference, Piscataway, NJ: IEEE Press, 2015: 1421-1426.

[5] Echeverr, Grace A. Lewis, James Root. Secure VM Migration in Tactical Cloutlets [C]// 2017 IEEE Military Communications Conference, Piscataway, NJ: IEEE Press, 2017: 388-393.

[6] Al-Riyami S S, Paterson K G. Certificateless Public Key Cryptography [C]// 9th International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2003: 452-473.

[7] Lang Xiaoli, Cao Suzhen, Liu Xiangzhen, et al. A Certificateless Public Key Authenticated Searchable Encryption Scheme With Efficient Authorization [J]. Computer Engineering and Science, 2020, 42 (03): 418-426.

[8] Tedeschi P, Sciancalepore S, Eliyan A, et al. LiKe: Lightweight Certifificateless Key Agreement for Secure IoT Communications [J]. IEEE Internet of Things Journel, 2020, 7 (1): 621-638.

[9] Khan M A, Ullah I, Nisar S. An Efficient and Provably Secure Certificateless Key-Encapsulated Signcryption Scheme for Flying Ad-hoc Network [J]. IEEE Access, 2020, 8: 36807-36828.

[10] Zhao Nan, Zhang Anguo. Authenticated Privacy Protection Scheme Based On Certificateless Ring Signcryption In VANET [J]. Computer Science, 2020, 47 (03): 312-319.

[11] Kalra S, Sood S K. Secure authentication scheme for IoT and cloud servers [J]. Persasive and Mobile Computing, 2015, 24: 210-223.

[12] Konlitz N. Elliptic curve cryptosystems [J]. Mathematics of Computation, 1987, 48 (177): 203-209.

[13] Wang K H, Chen C M, Fang W, et al. A secure authentication scheme for IoT and cloud servers [J]. Pervasive and Mobile Computing, 2017, 42: 15-24.

[14] Chang C C, Wu H L, Sun C Y. Notes on "Secure authentication scheme for IoT and cloud servers" [J]. Pervasive and Mobile Computing, 2017, 38: 275-278.

[15] Echeverria S, Klinedinst D, Williams K. Establishing Trusted Identities in Disconnected Edge Environments [C]// 1st Symposium on Edge Computing, Piscataway, NJ: IEEE Press, 2016: 51-63.

[16] SHAMIRA. Identity-based cryptosystems and signature schemes [C]// Annual International Cryptology Conference, Berlin: Springer, 1984: 47-53.

[17] Boneh D, Franklin M. Identity-Based Encryption from the Weil Pairing [C]// 21st Anuual International Cryptology Conference, Berlin: Springer, 2001: 586-615.

[18] Pedersen T P. A Threshold Cryptosystem without a Trusted Party [C]// Proceedings of the 10th Annual International Conference, Berlin: Springer, 1991: 522-526.

[19] Echeverria S, Klinedinst D, Seitz L. Authentication and Authorization for IOT Devices in Disadvantage Environments [C]// World Forum on Internet of Things. Piscataway, NJ: IEEE Press, 2019: 368-372.

[20] Seo S H, Won J, Sultana S, et al. Effective Key Management in Dynamic Wireless Sensor Networks [J]. IEEE transactions on information forensics and security, 2015, 10 (2): 371-383.

[21] Seo S H, Won J, Bertino E. pCLSC-TKEM: a Pairing-free Certificateless Signcryption-tag Key Encapsulation Mechanism for a Privacy-Preserving IoT

[J]. Transactions on Data Privacy, 2016, 9 (2): 101-130.

[22] Boneh D, Lynn B, Shacham H, Short signatures from the Weil pairing [J]. Journal of Cryptology, 2004, 17 (4): 297–319.

[23] Zhou Caixue, Zhao Zhiqiang, Wan Zhou, et al. Certificateless Key-Insulated Generalized Signcryption Scheme without Bilinear Pairings [J]. Security & Communication Networks, 2017, 2017: 1-17.

[24] Xiong Hu, Mei Qian, Zhao Yanan. Efficient and Provably Secure Certificateless Parallel Key-Insulated Signature Without Pairing for IIoT Environments [J]. IEEE Systems Journal, 2020, 14 (1) 310-320.

[25] Bellare, Mihir, Canetti: Keying Hash Functions for Message Authentication [M]. Advances in Cryptology —CRYPTO' 96. Berlin: Springer, 1996.

Note: Figure translations are in progress. See original paper for figures.