

A Privacy Preserving Authentication Protocol for
Low Power Devices
ComS/CprE 554 Project Report
Supervised by Dr. Wensheng Zhang

Michael Fong
mcfong@iastate.edu

May 8, 2009

1 Introduction

A wireless sensor network (WSN) is an ad-hoc based network, composed of small sensor nodes deployed in large numbers to collect critical data in the physical world, such as surveillance or monitoring of natural and manmade environments. This view is achieved by deploying massive amount of small wireless sensors, (so called sensor nodes, motes, or dust). As sensor network applications expand to perform sensitive measurements of everyday life, such an extensive use of technology will expose to many security attacks. Several security concerns [1, 2] have already been identified, and out of which violation of the user privacy becomes an increasingly important topic.

In addition to the real-life need for WSN, the very same scenario could also happen to another variation of wireless communication - the Radio Frequency Identification (RFID), which is a system that enables wirelessly massive identification and tracking of items. Two components involved in the systems are RFID tags and readers. The tags contain a radio frequency transponder and a read-only (sometimes re-writable) memory chip that contains a unique identifier. Tags get queried by readers which are more complex and usually connected to a back-end system, i.e. a database. Upon being queried, the tags respond with their IDs. The readers are capable to query multiple tagged items at once and distinguish between each one of them. We distinguish between active and passive tags. Active tags carry a small battery while the cheaper and much more common passive tags receive their power from the reader. The reader emits an electric field while querying the tags, which also powers the tags. Nevertheless, in practice, both parties are constantly exposed in the untrusted environment, which might lack communication confidentiality, data integrity, or mutual authentication, and thereby damaging customer privacy[3].

In this project, we will introduce an alternative protocol of low-cost computation. This scheme is able to authenticate both parties, while user identity is not revealed to the reader side. In the other words, user privacy is ultimately preserved.

1.1 Threats to Privacy

Wireless Sensor Network: A general overview of security issues for general wireless networks can be found in related work done by Stajano and Anderson [4]. Ad-hoc sensor network (ASN) is a special form of wireless sensor networks (WSN), in which the nodes can change, new nodes can be added, old nodes retired, and sometimes nodes might be moved to a different network. Here are some examples for WSN in which security concerns they raise.

1. In a naive communication scenario between two sensor mote, there is a potential possibility for an unauthorized attacker to intercept the communication, steal private secrecy, or even forge user's identity. For instance, the unprotected sensor in vehicle might give out the vehicle and driver information to a rogue reading sensor on the roadside. Hence, without providing proper privacy and security protection, such applications of WSN is not practical.
2. WSN nodes can be used for military purposes. The sensors can either be dropped off an aircraft over enemy territory after which they lie stationary on the ground, or they are carried by each soldier and vehicle and therefore form a mobile ASN. Its goal is to detect and gain as much information as possible about enemy movements. This information is relayed to a mobile command posts that helps the commanders to make decisions about troop movements, calls for air support, etc. In this scenario it is of utmost importance to prevent the enemy from intercepting the transmitted data. In addition to this, it is very likely that some nodes fail and stop operating, or some nodes were compromised by the enemy.

Radio Frequency Identification Devices: Security aspects of RFIDs that work at different layer (Physical Layer[5], Communication Layer[6], or Application Layer[7, 8, 9]), have been proposed. We are presenting two popular applications for RFID tags and highlight their security concerns.

1. The wireless nature with no line-of-sight requirement makes RFID ideal for inventory control and fast check out. Thus, each tag embedded with a unique identifier follows a standardized electronic product code (EPC), which is going to replace the optical barcode in near future.

2. The US government finally issued its first passports containing RFID (Radio Frequency ID) chips in October 2006. The embedded chips in the new passports contains the same information in the old printed document, including including the name, nationality, sex, date of birth, place of birth, fingerprint, and a photo of the passport holder. According to government officials, the use of the RFID chip allows passports to be scanned and cross-referenced with security databases more easily. Due to the nature of communication in RFID, identity theft can be done wirelessly (within 10-300 feet range)[10], because your private information is just up there available in the air for hackers, who would hack into the device, snap personal information, and walk away. The threat of unauthorized duplication of your passport have affected millions of Americans.

2 Security Concern

Security in WSN is quite different from traditional (wired) network security. Due to the hardware obstacle, the WSN is difficult to directly employ the existing security approaches to the area of wireless sensor networks. Most of the security concerns can be addressed by the services of confidentiality , availability and integrity. When we review a real system in practice, however, authenticity and privacy are also within our consideration.

2.1 Confidentiality

¹

A sensor network should not leak sensor readings to any unauthorized parties (this illegal act is sometimes called skimming), therefore it is extremely important to build a secure channel in a wireless sensor network. Especially in a military application, the data stored in the WSN node is highly sensitive. The standard solution for keeping sensitive data secret is to encrypt the data with a secret key, known only to the sender and receiver. The receiver would then decrypt the data, and thus achieve confidentiality.

2.2 Integrity

²

In the wireless world, the information exchanged between two parties needs to be confidential when sensitive data, such as secret keys, must not be collected by an eavesdropper ³. Fortunately, with the implementation

¹Confidentiality refers to the concealment of information.

²Data Integrity ensures that any received data has not been altered in transit.

³Eavesdropping is unauthorized listening/intercepting, through the use of radio receiving equipment, of an authorized transmission to monitor or record data between sender

of confidentiality, the attacker may be unable to steal information. However, the adversary may modify the message in transit without knowing the message content. For instance, the lack of authentication in the pure Diffie-Hellman key exchange protocol makes it vulnerable to man in the middle attack. Message authentication codes, hash functions and digital signatures can guarantee message integrity and as well as authenticity.

2.3 Availability

4

Availability is an important aspect of reliability, especially when a reader needs to be ready to authenticate every incoming user that may enters its communication range at certain time intervals. Even without the threat of a malicious node, a single point failure with no presence of centralized management would cause data loss or damage. For example, the functionality of the sensor network must be ensured to resist denial-of-service attacks (DoS). The typical countermeasures include Quality of Service (Qos), but is our focus in this project.

2.4 Authenticity

In any network communication, authentication proves the claimed identity of of the other parties, and it is an important security measure for preventing counterfeiting behaviors. Both the sender and receiver need to confirm the identity of other party involved in the communication The use of authentication may also be required in applications, this project focus on, such as security entry systems. In addition, a system equipped with strong Authentication indicates a system of proving knowledge of a secret of the other party without revealing it.

2.5 Privacy

Privacy, in general, refers to the ability of an entity to stop information about themselves from becoming known to people other than those whom they choose to give the information. In the world of wireless sensor network, while this technology promises to produce a massive amount of data collection, an adversaries can use seemly irrelevant fragment of data, assemble them, and derive much more sensitive information. Therefore, the data aggregation in sensor network enlarge the problem of privacy, because they make large amount of information easily available through remote access [11]. Hence, adversaries does not have not be physically present to main-

and receiver

⁴Availability refers to the ability to use the information desired

tain surveillance, but they can gather information in a low-risk, anonymous manner.

- **Eavesdropping:** By listening to the data, the adversary could easily discover the communication contents. When the traffic conveys the control information about the sensor network configuration, which contains potentially more detailed information about the communication protocol, the eavesdropping can act effectively against the privacy protection.
- **Traffic Analysis:** This is an advanced version of combination with monitoring and eavesdropping. An attacker could potentially monitor the increase in the number of transmitted packets between certain nodes, and could signal that a specific sensor has registered activity. Through the analysis on the traffic, some sensors with previous activities can be effectively identified. Thus, this act violates the identity privacy.
- **Camouflage:** After an adversary inserts a rogue node or compromises a legitimate node in the sensor network, the attackers make those nodes impersonate as a normal node to attract the packets, and misroute the packets. e.g. another variation of man-in-the-middle attack that forwards the packets to the nodes conducting the privacy analysis.

2.6 Physical Attack

Sensor nodes in the real world are typically exposed in outdoor environments, and their deployment makes them highly compromisable by attackers. Instead of finding the weakness in an algorithm via cryptanalysis approach or brute force hack, a well-trained adversary could launch an attack based on cryptographic information gained from physical implementation of the node. Some attacks include measurement on how much time that various cryptographic systems take to perform (Timing Attack), make use of varying power consumption by the hardware during cryptographic computation (Power Consumption Attack), or even attack on leaked electromagnetic radiation that could directly provide un-encrypted plaintext messages. Recent work has shown that a sensor node that lacks tamper-resistant hardware protection, such as the MICA2 nodes, can be compromised in less than one minute [12] by a well-trained attacker. If an adversary compromises a sensor node, then the code inside the physical node may be modified.

3 Proposed Protocol

The goals of this project is to design a protocol that the reader authenticate the incoming user while preserving the privacy of the user. The use of this protocol may, for instance, be used in application, such as secure entry systems. We will not, however, focus on protection against availability or physical tampering attack. This simple protocol involves two flows with a challenge-response approach. In addition, it uses nonces (random numbers) to provide anonymity for each user response, so that reader would have no knowledge of user's identity during the process of authentication. Thus, the privacy is preserved.

3.1 Preliminary Assumption

The protocol uses a hash function, and a pseudo-random number generator.

3.1.1 Hash

A hash function h is an one-way function that maps an arbitrary length input to a k -bit output, i.e. $h : \{0, 1\}^* \rightarrow \{0, 1\}^k$. The typical requirement for this cryptographic checksum functions are

- Pre-image resistance: for any given input x , it is computationally efficient to compute $h(x)$. Nevertheless, given an arbitrary output y , it is computationally infeasible to find an input such that $h(x) = y$.
- 2nd pre-image resistance: given x , it is computationally infeasible to find $x' \neq x$, such that $h(x) = h(x')$
- Collision resistance: it is computationally infeasible to find any pair of distinct inputs x and x' , such that $h(x) = h(x')$

3.1.2 Pseudo-random Number Generator

A pseudo-random number generator (PRNG) is a deterministic algorithm, which given an input seed and outputs a binary sequence of length k , which appears to be random.

3.2 Assumption

Our protocol works under the following assumption

- Each user device has a Psuedo-random number generator, that performs a 16-bit number at (seemly) random.
- Each user device has a rewritable memory, which is used to preload two secret function, and its identity parameter.

- Each user device is implemented with a standardized cryptographic hash functions, such as Sha-1, which was pre-assumably implementable on RFID tags.
- The reader and the user device communicate over an insecure channel, so their communications are subject to eavesdropping.

3.3 Set Up Phase

Each user i is preloaded with two polynomial functions:

$$\begin{aligned} A_1(x, y) &= a_3xy + a_2x + a_1y + a_0 \\ A_2(x, y) &= \hat{a}_3xy + \hat{a}_2x + \hat{a}_1y + \hat{a}_0 \end{aligned}$$

, where different users have different coefficients $(\{a_1, a_2, a_3\}, \{\hat{a}_1, \hat{a}_2, \hat{a}_3\})$.

On the other hand, reader r is preloaded with three different polynomial functions

$$\begin{aligned} B_1(x) &= b_1x + b_0 \\ B_2(x) &= \hat{b}_1x + \hat{b}_0 \\ B_3(x) &= \sum_{t=0}^2 \bar{b}_t x^t \\ &= \bar{b}_2 x^2 + \bar{b}_1 x + \bar{b}_0 \end{aligned}$$

, where the coefficients $(\{b_1, b_2\}, \{\hat{b}_1, \hat{b}_2\}, \{\bar{b}_2, \bar{b}_1, \bar{b}_0\})$ varies among different readers.

Those coefficients needs to be carefully selected, so that $A_1(x, y) \cdot B_1(x) + A_2(x, y) \cdot B_2(x) + B_3(x) \equiv 0$ is guaranteed. The coefficients can be found via algorithms that solves linear equations.

3.4 Authentication Process

The protocol is summarized in Figure.1

1. Reader: A reader generates a random bit-string $r_0 \in \{0, 1\}^l$, and sends this nonce $M_0 = \langle r_0 \rangle$ to user i .
2. User: The user i generates a random bit-string $r_1 \in \{0, 1\}^l$ as temporary session secret, and computes $r_2 = h(r_0|r_1)$, where $h(x)$ is our hash function. Next, user computes its polynomial functions $A_1(r_2, i)$ and $A_2(r_2, i)$, and sends back to the reader with message $M_1 = \langle r_1, A_1(r_2, i), A_2(r_2, i) \rangle$.

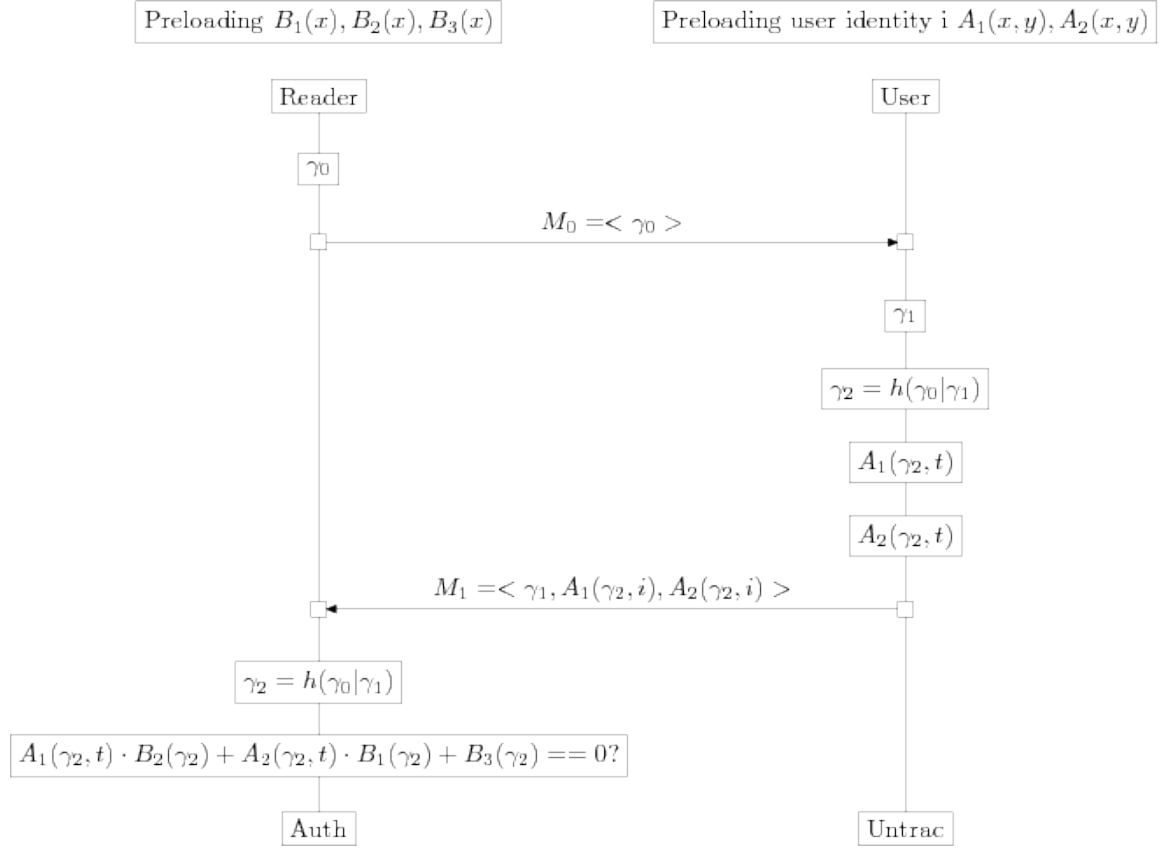


Figure 1: Protocol

3. Reader: The reader is not able to compute $r_2 = h(r_0|r_1)$, and computes its secret functions with r_2 . The reader will authenticate user i , if $A_1(r_2, i) \cdot B_1(r_2) + A_2(r_2, i) \cdot B_2(r_2) + B_3(r_2) \equiv 0$; reject this user, otherwise.

3.5 Security Analysis

The protocol has satisfied following privacy and security properties

- **Strong Authentication:** The identity of user i is never exposed in any phase of authentication, even to a valid reader. Therefore, the user identity stays anonymous.
- **User Impersonation Attack:** For a attacker to clone a user, it needs to compute valid coefficients $(\{a_1, a_2, a_3\}, \{\hat{a}_1, \hat{a}_2, \hat{a}_3\})$, which were pre-computed onto user and stayed secret in transmission. Thus, it is hard to compute such valid pairs of coefficients without any prior acknowledgement of the user. However, an attack under the scenario of

compromise of a legitimate reader enables the adversary to produce a valid user, and we will address this issue and solution below.

- **Reply Attack:** The scheme resists reply attack, since it is a challenge-response authentication protocol using several random numbers generated as temporary secret for each session. The message M_0 and M_1 are composed with those nonces, and thus those message cannot be reused in future sessions.

3.6 Potential Attack

If a reader is compromised, then $B_1(x), B_2(x), B_3(x)$ are compromised. If the attack can somehow figure out \hat{A}_1, \hat{A}_2 , such that $\hat{A}_1(x, y) \cdot B_1(x) + \hat{A}_2(x, y) \cdot B_2(x) + B_3(x) = 0$. That means some other user device \hat{i} with secret functions \hat{A}_1, \hat{A}_2 is compromised.

To prevent this attack, for any reader that could be potentially compromised. We preloaded the reader functions with elliptic curve point α to convert the numeric output into another elliptic curve point, $\alpha \cdot B_1(x), \alpha \cdot B_2(x), \alpha \cdot B_3(x)$. Thus, now the original output is obfuscated in the form of elliptic curve point. If the adversary needs to reversely compute the integer $B(x)$ by brute force, this is equivalent of computing the discrete log problem of modular base 2^{160} , and thus this is a hopeless attack. Nevertheless, the tradeoff is that the reader will suffer from more expensive EC point computation, and thus increase the authentication time.

3.7 Misuse of Anonymous Authentication

While the anonymous authentication provides user privacy, but this property may be misused to crash the system. The most obvious approach is that when an attack compromise one or several valid user devices, and launch multiple DoS attack simultaneously. The entry system would follow with a shut-down of authenticating service. Since this simple model does not support any mechanism to further identify who the user was, hence, the attacker could go away without leaving any trace after his successful attack.

4 Implementation

We conduct the implementation on two TelosB Mote, one as reader and the other as user,⁵ supported by UC Berkeley's 2.x TinyOS[13] operating

⁵TelosB mote is integrating an IEEE 802.15.4/ZigBee compliant RF transceiver (ranging from 2.4 to 2.4835 GHz) at 250 kbps data rate, an integrated onboard antenna, and a 16 bit, 8MHz TI MSP430 microcontroller with 10kB RAM plus 1MB external flash for data logging, programming and data collection via USB. The mote is battery-chargeable of two AA batteries

Scenario	Authentication Time
Basic	103 ms
EC	3600 ms

Table 1: Execution Time

Scenario	Entry Code Size	User Code Size
Basic	21kB (ROM) + 1.0kB (RAM)	21kB (ROM) + 1.3kB (RAM)
EC	31kB (ROM) + 2.7kB (RAM)	21kB (ROM) + 1.3kB (RAM)

Table 2: Code Size

system⁶ and the NesC[14] programming language. For ECC implementation, we adapted a WM-ECC⁷ package for TinyOS, developed by a security team led by Dr. Qun Li at College of William and Mary. In experiments, we make appropriate modifications on their program to make them executable on TinyOS 2.x platform.

4.1 Performance Analysis

In experiments, we first measure the execution time of between the basic implementation and the scenario of adapting EC computation, which involves with several point multiplication and point addition. Here is the result:

The following table compares the code size of two implementation on both entry and user sides. The major difference is EC scenario occupy approximately 30kB ROM and 2.7kB RAM, whereas the basic model uses 21kB ROM and 1.0kB RAM. Both model implements Sha-1 hash function.

5 Conclusion

In this project, we show our implementation of a privacy-preserving authentication protocol on TelosB platform, and compare the performance with different enhanced features on TelosB. Meanwhile, our experiment shows that it takes a reasonable 3.6 seconds to conduct an authentication process.

⁶TinyOS is an open-source operating system designed for wireless embedded sensor networks. It features a component-based architecture with minimized code size as required by the severe memory constraints inherent in sensor networks.

⁷WM-ECC is an Elliptic Curve Cryptography Suite on sensor motes. Official website: <http://www.cs.wm.edu/wanghd/>

This result shows us that EC computation is feasible for sensor network security applications. Moreover, this scheme should be applicable on RFID device, since all of the expensive EC computation is conducted on reader, which is much more computationally powerful than regular sensor mote.

Acknowledgements

We would like to thank Chaun Wang for many useful discussions on EC computations, and uses of operations in WM-ECC library.

APPENDIX

A Elliptic Curve in a Nutshell

In this section, we briefly give a background introduction about elliptic curve cryptography. Elliptic curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields.

An elliptic curve, \mathcal{C} , of the form $y^2 = x^3 + ax + b$, where $4a^3 + 27b^2 \neq 0$, is a cubic curve⁸. This curve is irreducible and has no singular points. Each value of the coefficients a and b give a different elliptic curve. All points (x,y) which satisfies the above equation plus a point at infinity lies on the elliptic curve.

The security of ECC depends on the difficulty of Elliptic Curve Discrete Logarithm Problem. Let P and Q be two points on an elliptic curve such that $k \cdot P = Q$, where k is a numeric scalar. Given P and Q , it is computationally infeasible to obtain k , if k is sufficiently large. In terms of discrete log problem, this is equivalent of saying k is the discrete logarithm of Q to the base P . That is, given P and Q , it is computationally infeasible to find an integer x , such that $Q \equiv P^k \pmod{p}$, in other words, solving $k \equiv \log_P(Q) \pmod{p}$, where p is finite field.

A.1 Point Addition

Point addition is the addition of two points P_1 and P_2 on an elliptic curve \mathcal{C} to obtain another point P_3 on the same elliptic curve.

If $P_1 \neq P_2$, then a line drawn through two points will intersect the curve \mathcal{C} at exactly one more point, $-P_3$. The reflection of the point P_3 with respect to x-axis gives the point P_3 , which is the result of $P_1 + P_2 = P_3$, as shown in figure (a).

Now consider, the case that $P_1 = P_2$, as shown figure (b), by definition, we include one more point, $E_f = \mathcal{C} \cup \{\mathcal{O}\}$, point at infinity, denoted as \mathcal{O} . \mathcal{O} is sitting at the top of the y-axis. Hence, $P_1 + (-P_1) = P_2 + (-P_2) = \mathcal{O}$

To calculate $P_3 = (x_3, y_3)$, first assume $P_1 = (x_1, y_1), P_2 = (x_2, y_2)$. Then

$$\begin{cases} x_3 = & m^2 - x_1 - x_2 \\ y_3 = & m \cdot (x_1 - x_3) - y_1 \end{cases}$$

⁸If $f(x, y)$ has degree 3, then the set $\mathcal{C} = (x, y) \in R^2 : f(x, y) = 0$ is called a cubic curve.

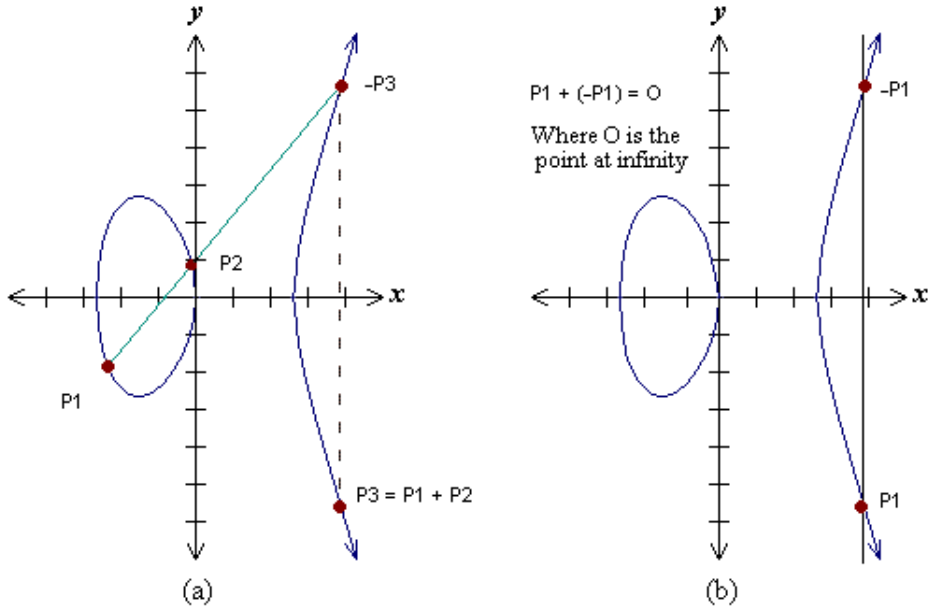


Figure 2: Point Addition

, and

$$m = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1} & \text{if } P_1 \neq P_2 \\ \frac{3x_1^2 + a}{2y_1} & \text{if } P_1 = P_2 \end{cases}$$

Last, if the slope m is infinite, then $P_1 + P_2 = \mathcal{O}$

A.2 Point Doubling

Point doubling is the addition of a point P_1 on the elliptic curve to itself to obtain another point P_2 on the same elliptic curve.

To double a point P_1 to get P_2 , i.e. $P_2 = 2 \cdot P_1$, consider a point P_1 on an elliptic curve as shown in figure (a). If y coordinate of the point J is not zero then the tangent line at J will intersect the elliptic curve at exactly one more point $-P_2$. The reflection of the point $-P_2$ with respect to x -axis gives the point P_2 , which is the result of doubling the point P_1 . Thus $P_2 = 2 \cdot P_1$.

A.3 Point Subtraction

Similar to point addition, except we take the reflection of second point to perform same addition operation. $P_1 - P_2 = P_1 + (-P_2) = P_3$.

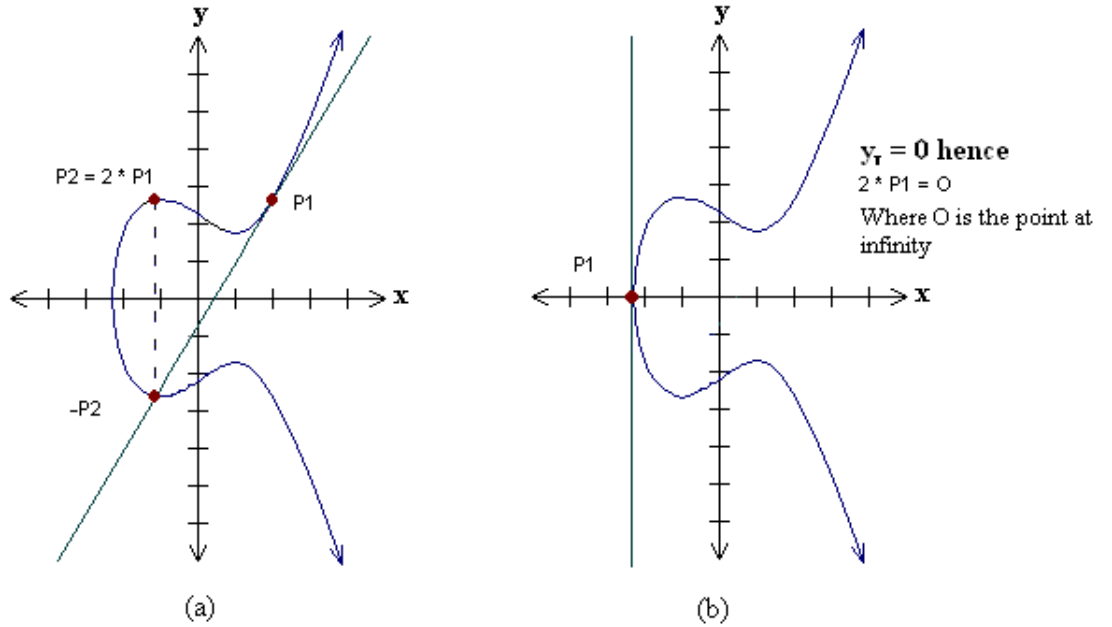


Figure 3: Point Doubling

A.4 Point Multiplication

In point multiplication, a point P_1 on the elliptic curve is multiplied with a scalar k using elliptic curve equation to obtain another point P_2 on the same elliptic curve. Point multiplication is achieved by two basic elliptic curve operations:

- Point addition, adding two points P_1 and P_2 to obtain another point P_3 i.e., $P_3 = P_1 + P_2$.
- Point doubling, adding a point P_1 to itself to obtain another point P_2 i.e. $P_2 = 2 \cdot P_1$.

, and follows the rule,

$$k \cdot P_1 = \begin{cases} P_1 & \text{if } k = 1 \\ \frac{k}{2} \cdot (P_1 + P_1) & \text{if } k \text{ is even} \\ P_1 + (k - 1) \cdot P_1 & \text{otherwise} \end{cases}$$

Therefore, to perform point multiplication over a point P with a scalar k is exactly completing point addition for k times.

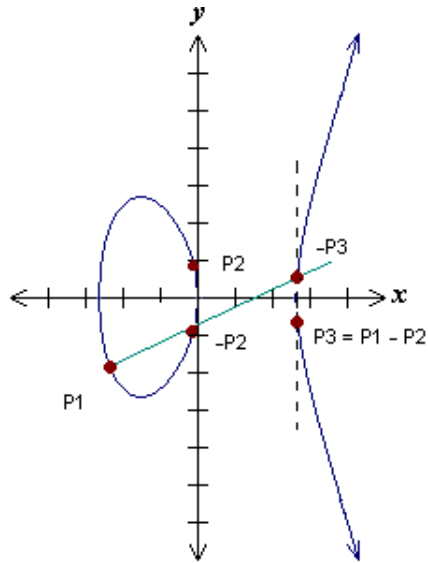


Figure 4: Point Subtraction

A.5 EC on Prime Field

The equation of the elliptic curve on a prime field F_p is $y^2 \equiv x^3 + ax + b \pmod{p}$, where $4a^3 + 27b^2 \pmod{p} \neq 0$. In addition, the elements of the finite field are integers between 0 and $p - 1$. All the operations such as addition, subtraction, division, multiplication involves integers between 0 and $p - 1$. The prime number p is chosen at large such that there is finitely large number of points on the elliptic curve to make the system secure. In WM-ECC, p is predefined as large as $p = 2^{160} - 2^{31} - 1$

References

- [1] F. Hu and N. K. Sharma., “Security considerations in ad hoc sensor networks,” *Ad Hoc Networks*, vol. 3, p. 6989, 2005.
- [2] A. Perrig, J. Stankovic, and D. Wagner, “Security in wireless sensor networks,” *Commun. ACM*, vol. 47(6), p. p5357, Jun 2004.
- [3] A. Juels, “Rfid security and privacy: a research survey,” *Selected Areas in Communications, IEEE Journal on*, vol. 24, no. 2, pp. 381–394, 2006.
- [4] F. Stajano and R. Anderson, “The resurrecting duckling: Security issues for ad-hoc wireless networks.,” *Security Protocols, 7th International Workshop, Berlin, Heidelberg, 1999. Springer Verlag*.
- [5] M. R. Rieback, B. Crispo, and A. S. Tanenbaum, “Keep on blockin’ in the free world: Personal access control for low-cost rfid tags,” in *Security Protocols Workshop*, pp. 51–59, 2005.
- [6] A. Juels, R. L. Rivest, and M. Szydlo, “The blocker tag: selective blocking of rfid tags for consumer privacy,” in *CCS ’03: Proceedings of the 10th ACM conference on Computer and communications security*, (New York, NY, USA), pp. 103–111, ACM Press, 2003.
- [7] Y. Nohara, S. Inoue, K. Baba, and H. Yasuura, “Quantitative evaluation of unlinkable id matching schemes,” in *WPES ’05: Proceedings of the 2005 ACM workshop on Privacy in the electronic society*, (New York, NY, USA), pp. 55–60, ACM, 2005.
- [8] B. Song and C. J. Mitchell, “Rfid authentication protocol for low-cost tags,” in *WiSec ’08: Proceedings of the first ACM conference on Wireless network security*, (New York, NY, USA), pp. 140–147, ACM, 2008.
- [9] S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels, “Security and privacy aspects of low-cost radio frequency identification systems,” in *Security in Pervasive Computing*, vol. 2802 of *Lecture Notes in Computer Science*, pp. 201–212, 2004.
- [10] K. Koscher, A. Juels, and T. Kohno, “Epc rfid tags in security applications: Passport cards, enhanced drivers licenses, and beyond,”
- [11] H. Chan, A. Perrig, B. Przydatek, and D. X. Song, “Sia: Secure information aggregation in sensor networks,” *Journal of Computer Security*, vol. 15, no. 1, pp. 69–102, 2007.
- [12] V. De and S. Borkar, “Technology and design challenges for low power and high performance [microprocessors],” *International Symposium on Low Power Electronics and Design (ISLPED) 1999*, p. pages 163168, 1999.

- [13] “Tinyos, <http://www.tinyos.net/>.”
- [14] R. v. B. M. W. E. B. David Gay, Phil Levis and D. Culler, “The nesc language: A holistic approach to networked embedded systems,” *In Programming Language Design and Implementation (PLDI)*, June 2003.