

Perbezaan serahan KAZ-KA v1.0 kali pertama dan kedua

Bil	Serahan pertama	Serahan kedua	Catatan																																																												
1.	<p style="text-align: center;">Kriptografi Atasi Zarah Key Agreement (KAZ-KA)</p> <p style="text-align: center;">Algorithm Specifications and Supporting Documentation (KAZ-KA 1.0)</p> <p style="text-align: center;">Muhammad Reza Kamel Ariffin¹ Abderrahmane Nitaj² Nor Azman Abu³ Zahari Mahad¹</p> <p style="text-align: center;">¹Universiti Putra Malaysia ²Université de Caen Normandie, France ³Universiti Teknikal Malaysia Melaka</p>	<p style="text-align: center;">Kriptografi Atasi Zarah Key Agreement (KAZ-KA)</p> <p style="text-align: center;">Algorithm Specifications and Supporting Documentation (KAZ-KA 1.0)</p> <p style="text-align: center;">Muhammad Reza Kamel Ariffin¹ Abderrahmane Nitaj² Nor Azman Abu³ Zahari Mahad¹ Muhammad Asyraf Asbullah¹ Amir Hamzah Abd Ghafar¹</p> <p style="text-align: center;">¹Universiti Putra Malaysia ²Université de Caen Normandie, France ³Universiti Teknikal Malaysia Melaka</p>	Tambahan pengarang																																																												
2.	<table><tr><td>1 INTRODUCTION</td><td>1</td></tr><tr><td>2 THE DESIGN IDEALISME</td><td>1</td></tr><tr><td>3 THE HIDDEN NUMBER PROBLEM (HNP) (Boneh and Venkatesan, 2001)</td><td>2</td></tr><tr><td>4 COMPLEXITY OF SOLVING THE HNP</td><td>2</td></tr><tr><td>5 THE KAZ-KA KEY AGREEMENT ALGORITHM</td><td>2</td></tr><tr><td>5.1 Background</td><td>2</td></tr><tr><td>5.2 Utilized Functions</td><td>2</td></tr><tr><td>5.3 System Parameters</td><td>2</td></tr><tr><td>5.4 KAZ-KA Algorithms</td><td>3</td></tr><tr><td>6 PROOF OF CORRECTNESS</td><td>3</td></tr><tr><td>7 IMPLEMENTATION OF THE HIDDEN NUMBER PROBLEM (HNP)</td><td>4</td></tr><tr><td>8 DISCRETE LOGARITHM PROBLEM ANALYSIS</td><td>4</td></tr><tr><td>9 DERIVING THE SECURITY LEVEL OF KAZ-KA</td><td>5</td></tr></table>	1 INTRODUCTION	1	2 THE DESIGN IDEALISME	1	3 THE HIDDEN NUMBER PROBLEM (HNP) (Boneh and Venkatesan, 2001)	2	4 COMPLEXITY OF SOLVING THE HNP	2	5 THE KAZ-KA KEY AGREEMENT ALGORITHM	2	5.1 Background	2	5.2 Utilized Functions	2	5.3 System Parameters	2	5.4 KAZ-KA Algorithms	3	6 PROOF OF CORRECTNESS	3	7 IMPLEMENTATION OF THE HIDDEN NUMBER PROBLEM (HNP)	4	8 DISCRETE LOGARITHM PROBLEM ANALYSIS	4	9 DERIVING THE SECURITY LEVEL OF KAZ-KA	5	<table><tr><td>1 INTRODUCTION</td><td>1</td></tr><tr><td>2 THE DESIGN IDEALISME</td><td>1</td></tr><tr><td>3 THE DOUBLE DISCRETE LOGARITHM PROBLEM (DDLp)</td><td>2</td></tr><tr><td>4 UNIQUENESS</td><td>2</td></tr><tr><td>5 COMPLEXITY OF SOLVING THE DDLp</td><td>2</td></tr><tr><td>6 THE HIDDEN NUMBER PROBLEM (HNP) (Boneh and Venkatesan, 2001)</td><td>2</td></tr><tr><td>7 COMPLEXITY OF SOLVING THE HNP</td><td>3</td></tr><tr><td>8 THE KAZ-KA KEY AGREEMENT ALGORITHM</td><td>3</td></tr><tr><td>8.1 Background</td><td>3</td></tr><tr><td>8.2 Utilized Functions</td><td>3</td></tr><tr><td>8.3 System Parameters</td><td>3</td></tr><tr><td>8.4 KAZ-KA Algorithms</td><td>3</td></tr><tr><td>9 PROOF OF CORRECTNESS</td><td>4</td></tr><tr><td>10 IMPLEMENTATION OF THE DOUBLE DISCRETE LOGARITHM PROBLEM (DDLp)</td><td>4</td></tr><tr><td>11 IMPLEMENTATION OF THE HIDDEN NUMBER PROBLEM (HNP)</td><td>5</td></tr><tr><td>12 DISCRETE LOGARITHM PROBLEM ANALYSIS</td><td>5</td></tr><tr><td>13 DERIVING THE SECURITY LEVEL OF KAZ-KA</td><td>5</td></tr></table>	1 INTRODUCTION	1	2 THE DESIGN IDEALISME	1	3 THE DOUBLE DISCRETE LOGARITHM PROBLEM (DDLp)	2	4 UNIQUENESS	2	5 COMPLEXITY OF SOLVING THE DDLp	2	6 THE HIDDEN NUMBER PROBLEM (HNP) (Boneh and Venkatesan, 2001)	2	7 COMPLEXITY OF SOLVING THE HNP	3	8 THE KAZ-KA KEY AGREEMENT ALGORITHM	3	8.1 Background	3	8.2 Utilized Functions	3	8.3 System Parameters	3	8.4 KAZ-KA Algorithms	3	9 PROOF OF CORRECTNESS	4	10 IMPLEMENTATION OF THE DOUBLE DISCRETE LOGARITHM PROBLEM (DDLp)	4	11 IMPLEMENTATION OF THE HIDDEN NUMBER PROBLEM (HNP)	5	12 DISCRETE LOGARITHM PROBLEM ANALYSIS	5	13 DERIVING THE SECURITY LEVEL OF KAZ-KA	5	Tambahan isi kandungan
1 INTRODUCTION	1																																																														
2 THE DESIGN IDEALISME	1																																																														
3 THE HIDDEN NUMBER PROBLEM (HNP) (Boneh and Venkatesan, 2001)	2																																																														
4 COMPLEXITY OF SOLVING THE HNP	2																																																														
5 THE KAZ-KA KEY AGREEMENT ALGORITHM	2																																																														
5.1 Background	2																																																														
5.2 Utilized Functions	2																																																														
5.3 System Parameters	2																																																														
5.4 KAZ-KA Algorithms	3																																																														
6 PROOF OF CORRECTNESS	3																																																														
7 IMPLEMENTATION OF THE HIDDEN NUMBER PROBLEM (HNP)	4																																																														
8 DISCRETE LOGARITHM PROBLEM ANALYSIS	4																																																														
9 DERIVING THE SECURITY LEVEL OF KAZ-KA	5																																																														
1 INTRODUCTION	1																																																														
2 THE DESIGN IDEALISME	1																																																														
3 THE DOUBLE DISCRETE LOGARITHM PROBLEM (DDLp)	2																																																														
4 UNIQUENESS	2																																																														
5 COMPLEXITY OF SOLVING THE DDLp	2																																																														
6 THE HIDDEN NUMBER PROBLEM (HNP) (Boneh and Venkatesan, 2001)	2																																																														
7 COMPLEXITY OF SOLVING THE HNP	3																																																														
8 THE KAZ-KA KEY AGREEMENT ALGORITHM	3																																																														
8.1 Background	3																																																														
8.2 Utilized Functions	3																																																														
8.3 System Parameters	3																																																														
8.4 KAZ-KA Algorithms	3																																																														
9 PROOF OF CORRECTNESS	4																																																														
10 IMPLEMENTATION OF THE DOUBLE DISCRETE LOGARITHM PROBLEM (DDLp)	4																																																														
11 IMPLEMENTATION OF THE HIDDEN NUMBER PROBLEM (HNP)	5																																																														
12 DISCRETE LOGARITHM PROBLEM ANALYSIS	5																																																														
13 DERIVING THE SECURITY LEVEL OF KAZ-KA	5																																																														
3.	<p>1. INTRODUCTION</p> <p>The proposed KAZ Key Agreement scheme, KAZ-KA (in Malay <i>Kriptografi Atasi Zarah</i> - translated literally “cryptographic techniques overcoming particles”; particles here referring to the photons) is built upon the hard mathematical problem coined as the Hidden Number Problem (HNP). The idea revolves around the difficulty of reconstructing an unknown product from a given public parameter. The target of the KAZ-KA design is to be a quantum resistant key agreement candidate with short parameters, executing correctly 100% of the time, based on simple mathematics, having fast execution time and a potential candidate for seamless drop-in replacement in current cryptographic software and hardware ecosystems.</p>	<p>1. INTRODUCTION</p> <p>The proposed KAZ Key Agreement scheme, KAZ-KA (in Malay <i>Kriptografi Atasi Zarah</i> - translated literally “cryptographic techniques overcoming particles”; particles here referring to the photons) is built upon the hard mathematical problem coined as the Double Discrete Logarithm Problem (DDLp) and Hidden Number Problem (HNP). The target of the KAZ-KA design is to be a quantum resistant key agreement candidate with short parameters, executing correctly 100% of the time, based on simple mathematics, having fast execution time and a potential candidate for seamless drop-in replacement in current cryptographic software and hardware ecosystems.</p>	Perubahan pada Introduction (m/s:1)																																																												
4.	<p>One of our key strategy to obtain items (i) - (v) was by utilizing our defined Hidden Number Problem (HNP). It is defined in the following section.</p>	<p>One of our key strategy to obtain items (i) - (v) was by utilizing our defined Double Discrete Logarithm Problem (DDLp) and Hidden Number Problem (HNP). It is defined in the following section.</p>	Perubahan pada perenggan terakhir (m/s:1)																																																												
5.	<p>3. THE HIDDEN NUMBER PROBLEM (HNP) (Boneh and Venkatesan, 2001)</p> <p>Fix p and u. Let $O_{\alpha,g}(x)$ be an oracle that upon input x computes the most u significant bits of $\alpha g^x \pmod{p}$. The task is to compute the hidden number $\alpha \pmod{p}$ in expected polynomial time when one is given access to the oracle $O_{\alpha,g}(x)$. Clearly, one wishes to solve the problem with as small u as possible. Boneh and Venkatesan (2001) demonstrated that a bounded number of most significant bits of a shared secret are as hard to compute as the entire secret itself.</p> <p>The initial idea of introducing the HNP is to show that finding the u most significant bits of the shared key in the Diffie-Hellman key exchange using users public key is equivalent to computing the entire shared secret key itself.</p> <p>4. COMPLEXITY OF SOLVING THE HNP</p> <p>The complexity to obtain $\alpha \pmod{p}$ is $O(p)$. When deploying Grover’s algorithm on a quantum computer, the complexity to obtain $\alpha \pmod{p}$ is $O(p^{\frac{1}{2}})$.</p> <p>5. THE KAZ-KA KEY AGREEMENT ALGORITHM</p>	<p>3. THE DOUBLE DISCRETE LOGARITHM PROBLEM (DDLp)</p> <p>Let $N = \prod_{i=1}^l p_i$ be a public modulus. Choose random primes (g_1, g_2) where $\text{DLog}_{g_1}(g_2)$ and $\text{DLog}_{g_2}(g_1)$ modulo N does not exist. Let O_{g_1N} be the order of g_1 in \mathbb{Z}_N^*. Let O_{g_2N} be the order of g_2 in \mathbb{Z}_N^*. Compute $A \equiv g_1^{g_2} \pmod{N}$ for some $s \in \mathbb{Z}_{O_{g_1N}}$ and $t \in \mathbb{Z}_{O_{g_2N}}$. The DDLp is, upon given the parameters (A, g_1, g_2, N), one is tasked to identify the pair (s, t).</p> <p>4. UNIQUENESS</p> <p>Assume there exists (α, β) such that $g_1^\alpha g_2^\beta \equiv g_1^s g_2^t \pmod{N}$. We will have,</p> $g_1^{\alpha - s} \equiv g_2^{t - \beta} \pmod{N}$ <p>Which implies either,</p> $g_1 \equiv g_2^{\frac{t - \beta}{\alpha - s}} \pmod{N}$ <p>or</p> $g_2 \equiv g_1^{\frac{\alpha - s}{t - \beta}} \pmod{N}$ <p>for some $\frac{t - \beta}{\alpha - s}, \frac{\alpha - s}{t - \beta} \in \mathbb{Z}_{O_{g_1N}}$. This is false, since we have chosen random primes (g_1, g_2) where $\text{DLog}_{g_1}(g_2)$ and $\text{DLog}_{g_2}(g_1)$ modulo N does not exist.</p> <p>5. COMPLEXITY OF SOLVING THE DDLp</p> <p>The complexity to obtain s is $O(s)$. Due to the strategies during key generation, we have the complexity $O(s) > O(2^k)$ where k is the chosen security level, either 128, 192 or 256. When deploying Grover’s algorithm on a quantum computer, the complexity to obtain s is $O(2^{\frac{k}{2}})$.</p> <p>6. THE HIDDEN NUMBER PROBLEM (HNP) (Boneh and Venkatesan, 2001)</p>	Tambahan penerangan (m/s:2)																																																												
6.	<p>5.3 System Parameters</p> <p>From the given security parameter k (either 128, 192 or 256; depending on the security level needed), prepare a list of the first j-primes larger than 2, $P = \{p_i\}_{i=1}^j$ (as of printing it is suggested $j = 65, 96, 122$). Let $N = \prod_{i=1}^j p_i$ be a public modulus. Choose random primes (g_1, g_2) where $\text{DLog}_{g_1}(g_2)$ modulo N does not exist. Let O_{g_1N} be the order of g_1 in \mathbb{Z}_N^*. Let O_{g_2N} be the order of g_2 in \mathbb{Z}_N^*. The system parameters are $(g_1, g_2, O_{g_1N}, O_{g_2N}, N)$.</p>	<p>8.3 System Parameters</p> <p>From the given security parameter k (either 128, 192 or 256; depending on the security level needed), prepare a list of the first j-primes larger than 2, $P = \{p_i\}_{i=1}^j$ (as of printing it is suggested $j = 65, 96, 122$). Let $N = \prod_{i=1}^j p_i$ be a public modulus. Choose random primes (g_1, g_2) where $\text{DLog}_{g_1}(g_2)$ and $\text{DLog}_{g_2}(g_1)$ modulo N does not exist. Let O_{g_1N} be the order of g_1 in \mathbb{Z}_N^*. Let O_{g_2N} be the order of g_2 in \mathbb{Z}_N^*. The system parameters are $(g_1, g_2, O_{g_1N}, O_{g_2N}, N)$.</p>	Tambahan penerangan (m/s:3)																																																												

7.	<p>7. IMPLEMENTATION OF THE HIDDEN NUMBER PROBLEM (HNP)</p> <p>It is clear that the following parameters are implementing the HNP:</p> <ol style="list-style-type: none"> $e_1 \equiv g_1^{a_1} g_2^{2a_2} \pmod{N}$ $e_2 \equiv g_1^{a_2} g_2^{a_1} \pmod{N}$ $B_1 \equiv g_1^{b_1} g_2^{b_2} \pmod{N}$ $B_2 \equiv g_1^{b_2} g_2^{2b_1} \pmod{N}$ <p>Furthermore, c can be re-written as</p> $c \equiv (x)(e_1^{b_1} e_2^{b_2}) \pmod{N} \quad (1)$ <p>for unknown pair x. It is obvious that (1) is the HNP.</p>	<p>11. IMPLEMENTATION OF THE HIDDEN NUMBER PROBLEM (HNP)</p> <p>It is clear that the following parameters are implementing the HNP:</p> <ol style="list-style-type: none"> $A_1 \equiv g_1^{a_1} g_2^{2a_2} \equiv x_1 g_2^{2a_2} \pmod{N}$, where $x_1 \equiv g_1^{a_1} \pmod{N}$ $A_2 \equiv g_1^{a_2} g_2^{a_1} \equiv x_2 g_2^{a_1} \pmod{N}$, where $x_2 \equiv g_1^{a_2} \pmod{N}$ $B_1 \equiv g_1^{b_1} g_2^{b_2} \equiv x_3 g_2^{b_2} \pmod{N}$, where $x_3 \equiv g_1^{b_1} \pmod{N}$ $B_2 \equiv g_1^{b_2} g_2^{2b_1} \equiv x_4 g_2^{2b_1} \pmod{N}$, where $x_4 \equiv g_1^{b_2} \pmod{N}$ 	Penambahbaikan penerangan (m/s: 4)
8.	Tiada	<p>10. IMPLEMENTATION OF THE DOUBLE DISCRETE LOGARITHM PROBLEM (DDLp)</p> <p>It is clear that the following parameters are implementing the DDLp:</p> <ol style="list-style-type: none"> $A_1 \equiv g_1^{a_1} g_2^{2a_2} \pmod{N}$ $A_2 \equiv g_1^{a_2} g_2^{a_1} \pmod{N}$ $B_1 \equiv g_1^{b_1} g_2^{b_2} \pmod{N}$ $B_2 \equiv g_1^{b_2} g_2^{2b_1} \pmod{N}$ 	Tambahan penerangan (m/s:4)
9.	<p>8. DISCRETE LOGARITHM PROBLEM ANALYSIS</p> <p>Since $\text{DLog}_{g_1}(g_2)$ modulo N does not exist, an immediate DLP scenario does not occur with the equations (e_1, e_2, B_1, B_2). However, assume that h is a primitive root in \mathbb{Z}_N such that we have $h^{c_1} \equiv g_1 \pmod{N}$ and $h^{c_2} \equiv g_2 \pmod{N}$. One would now have the following equations:</p> $e_1 \equiv h^{c_1 a_1 + 2c_2 a_2} \pmod{N} \quad (2)$ $e_2 \equiv h^{c_1 a_2 + c_2 a_1} \pmod{N} \quad (3)$	<p>12. DISCRETE LOGARITHM PROBLEM ANALYSIS</p> <p>Since $\text{DLog}_{g_1}(g_2)$ and $\text{DLog}_{g_1}(g_1)$ modulo N does not exist, an immediate DLP scenario does not occur with the equations (A_1, A_2, B_1, B_2). However, assume that h is a primitive root in \mathbb{Z}_N where N is either $1, 2, 4, p^k$, or $2p^k$ where p is an odd prime number and k is a positive integer such that we have $h^{c_1} \equiv g_1 \pmod{N}$ and $h^{c_2} \equiv g_2 \pmod{N}$. One would now have the following equations:</p> $A_1 \equiv h^{c_1 a_1 + 2c_2 a_2} \pmod{N} \quad (1)$ $A_2 \equiv h^{c_1 a_2 + c_2 a_1} \pmod{N} \quad (2)$	Tambahan penerangan dan pembetulan typo (m/s: 5)
10.	Recall that N is a product of small primes. With high probability \mathbb{Z}_N does not have a primitive root. As such, an adversary would now search for an element $h_1 \in \mathbb{Z}_N$ such that the random walk of h_1 generates both g_1 and g_2 in \mathbb{Z}_N . The complexity is $O(N)$.	Assuming $\gcd(2c_1^2 - c_1^2, \phi(N)) = 1$ one would obtain a_2 . Thus, the existence of such primitive root $h \in \mathbb{Z}_N$ is the key towards the above cryptanalysis direction. However the parameter of N in KAZ-KA does not subscribe to either $1, 2, 4, p^k$, or $2p^k$ where p is an odd prime number and k is a positive integer.	Pebetulan penerangan (m/s: 5)
11.	<p>9. DERIVING THE SECURITY LEVEL OF KAZ-KA</p> <p>The challenge faced by the adversary is to retrieve (a_1, a_2) from (e_1, e_2). It is protected by the HNP.</p> <p>Due to the strategies during key generation, we have the complexity $O(a_1) > O(2^k)$ where k is the chosen security level, either 128, 192 or 256. When deploying Grover's algorithm on a quantum computer, the complexity to obtain N_1 is greater than $O(2^{\frac{k}{2}})$.</p>	<p>13. DERIVING THE SECURITY LEVEL OF KAZ-KA</p> <p>The challenge faced by the adversary is to retrieve (a_1, a_2) from (A_1, A_2). It is protected by the DDLp and HNP.</p> <div style="text-align: center;">5 <small>KAZ-KA 1.0</small></div> <hr/> <p>Due to the strategies during key generation, we have the complexity $O(a_1) > O(2^k)$ where k is the chosen security level, either 128, 192 or 256. When deploying Grover's algorithm on a quantum computer, the complexity to obtain a_1 is greater than $O(2^{\frac{k}{2}})$.</p>	Pembetulan penerangan dan typo (m/s: 5)
12.	<p>11. ADVANTAGES AND LIMITATIONS</p> <p>As we have seen, KAZ-KA can be evaluated through:</p> <ol style="list-style-type: none"> Key length Speed No verification failure 	<p>15. ADVANTAGES AND LIMITATIONS</p> <p>As we have seen, KAZ-KA can be evaluated through:</p> <ol style="list-style-type: none"> Key length Speed No full key generation failure 	Pembetulan input (m/s: 7)
13.	<p>11.3 No Full Key Generation Failure</p> <p>From the proof of correctness, the probability of decapsulation failure is 0.</p>	<p>15.3 No Full Key Generation Failure</p> <p>From the proof of correctness, the probability of full key generation failure is 0.</p>	Pembetulan input (m/s: 7)
14.	<p>11.4 Limitation</p> <p>As we have seen, limitation of KAZ-KA can be evaluated through:</p> <ol style="list-style-type: none"> Based on not widely used problem, the Hidden Number Problem (HNP). 	<p>15.4 Limitation</p> <p>As we have seen, limitation of KAZ-KA can be evaluated through:</p> <ol style="list-style-type: none"> Based on not widely used problems, the Double Discrete Logarithm Problem (DDLp) and Hidden Number Problem (HNP). 	Pembetulan input (m/s: 7)
15.	<p>11.4.1 Based on not widely used problem, Hidden Number Problem (HNP)</p> <p>The HNP is not a known hard mathematical problem which is quantum resistant and is subject to future cryptanalysis success in solving the defined challenge either with a classical or quantum computer.</p>	<p>15.4.1 Based on not widely used problems, Double Discrete Logarithm Problem (DDLp) and Hidden Number Problem (HNP)</p> <p>The DDLp and HNP are not known hard mathematical problems which are quantum resistant and are subject to future cryptanalysis success in solving the defined challenge either with a classical or quantum computer.</p>	Pembetulan input (m/s: 8)

16.	<p>12. CLOSING REMARKS</p> <p>The KAZ-KA partial and full key generation mechanism exhibits properties that might result in it being a desirable post quantum key agreement mechanism scheme.</p> <p>To this end, the security is based on the HNP, which is not a widely used problem. We opine that, the acceptance of HNP as a potential quantum resistant hard mathematical problem will come hand in hand with a secure cryptosystem designed upon it. We welcome all comments on the KAZ-KA key agreement mechanism, either findings that nullify its suitability as a post quantum key agreement mechanism scheme or findings that could enhance its deployment and use case in the future.</p>	<p>16. CLOSING REMARKS</p> <p>The KAZ-KA partial and full key generation mechanism exhibits properties that might result in it being a desirable post quantum key agreement mechanism scheme.</p> <p>To this end, the security is based on the DDLP and HNP, which are not a widely used problems. We opine that, the acceptance of DDLP and HNP as potential quantum resistant hard mathematical problems will come hand in hand with a secure cryptosystem designed upon it. We welcome all comments on the KAZ-KA key agreement mechanism, either findings that nullify its suitability as a post quantum key agreement mechanism scheme or findings that could enhance its deployment and use case in the future.</p>	<p>Tambahan penerangan dan pembetulan typo (m/s: 8)</p>
-----	--	---	---