Algebraic cryptanalysis: AES and Boolean equations

Virendra Sule
Professor
Dept. of Electrical Engineering
IIT Bombay
vrs@ee.iitb.ac.in

(Presented at BARC)
August 5, 2011

Need for efforts in Cryptography

Does not need justification

- Secure information transaction: Confidentiality in communication, secure data storage, access control on information, authentication, secure channel for key exchange, non-repudiation.
- New applications: digital cash, E-voting ...
- Special applications: security of geographical data.
- Overcoming International restrictions on use of cryptography.

What is Cryptanalysis: breaching security

- Devising ways of gaining information about an encrypted message from publicly available messages.
- Estimating security weaknesses of primitives in practical situations by exploiting side channel information.
- Subverting functionality of cryptographic schemes: faking authentication, signature forgery, subverting protocols.
- Estimating difficulty of computing secret key bits from known plaintext ciphertext data: block and stream ciphers.
- Military cryptanalysis: ciphertext only attacks, modeling cipher machines, identification of cipher parameters.
- Side channel analysis of arithmetic on HW.

Importance of Cryptanalysis

- Capability in design and security evaluation of indigenous cryptographic infrastructure.
- Technological returns: world's first computer was developed in Bletchley park for cryptanalysis.
- Cannot reach space without investing in expertise in rocket engineering.
- Indigenous cryptological technology policy.

Algebraic cryptanalysis of block and stream ciphers

- Solving the algebraic system of equations $C = E(P, K)$ for key K given a pair of blocks (C, P).
- Only a single pair of block C, P is sufficient for solving the key. Unlike linear and differential cryptanalysis which require unrealistically large data to reflect statistical bias.
- TMTO or Rainbow table based attacks: infeasible due to size, requirements of same P block and large offline computation for each key.
- Solution problem is a satisfiability (SAT) problem commonly occurring in verification and digital system design. Open source tools are available and have effectively solved industrial size problems of verification.
- Can be formulated as a Boolean equation solving problem. (New proposal).
- SAT solution approach offers partial automation of the cryptanalysis process.

AES Round

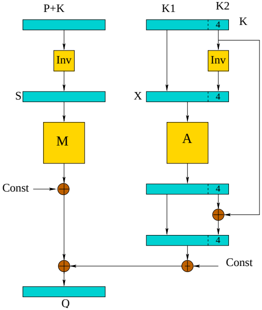

Figure: AES first round

AES equation systems

- Inversion map: $\text{Inv} : \mathbb{F}_{2^8} \to \mathbb{F}_{2^8}$

$$\begin{aligned} \text{Inv}(x) &= x^{-1} \quad \text{for } x \neq 0 \\ &= 0 \quad \text{for } x = 0 \end{aligned}$$

- Affine maps: State mixing $T_S(.)$, Key mixing $T_{Kr}(.,.)$
- `XOR` equation of round r

$$Q_r = T_S(S_r) \oplus T_{kr}(X_r, K_{r-1})$$

- Inversion equations of round r

$$S_r = \text{Inv}(Q_{r-1}), C_4(X_r) = \text{Inv}(C_4(K_{r-1}))$$

$C_i(X_r) = C_i(K_{r-1}),\ i = 1, 2, 3.$

- For $r = 10$ the mix column operation in $T_S(.)$ is absent. For $r = 1$, $Q_0 = P + K_0$. Constant in $T_{Kr}(.)$ equals θ^r where θ is the root of the polynomial $f(X) = X^8 + X^4 + X^3 + X + 1$ defining $\mathbb{F}_{2^8}$.

MQ system for AES

Follows from an MQ system representation for Inv(.) map. Given by

$$\begin{aligned} x^2y + x &= 0 \\ xy^2 + y &= 0 \end{aligned}$$

where $+$ is addition in $\mathbb{F}_{2^8}$. Denoting by $\bar{x}$ the 8-bit byte for x the equations in bit co-ordinates are,

$$\begin{aligned} L_{\bar{y}}\Sigma\bar{x} \oplus \bar{x} &= 0 \\ L_{\bar{x}}\Sigma\bar{y} \oplus \bar{y} &= 0 \end{aligned}$$

Using these equations to write quadratic relations in bits of S_r, Q_{r-1} and $X_r, Kr-1$ turns out to be an MQ system in smallest number of quadratic monomials.

Example of field multiplication and Frobenius map as Boolean operations

Field $\mathbb{F}_{2^4}$, generator polynomial $f(X) = X^4 + X + 1$. Operators represented in the polynomial basis.

- Multiplication operator

$$L_x y = \begin{bmatrix} x_0 & x_3 & x_2 & x_1 \\ x_1 & x_0 + x_3 & x_2 + x_3 & x_1 + x_2 \\ x_2 & x_1 & x_0 + x_3 & x_2 + x_3 \\ x_3 & x_2 & x_1 & x_0 + x_3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$

- Frobenius map

$$x^2 = \Sigma(x) = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Estimates on number of variables and equations

- Inv(.) map: 16 variables and 16 quadratic equations for each inversion. Number of inversions in each round: 16 state inversions, 4 key inversions: $20 \times 16 = 320$ quadratic equations.
- Affine maps: Number of variables:
  text mixing states S_r + key mixing states $X_r = 128 + 128 = 256$. Number of equations 256.
- Total numbers for each round: Number of equations: 576, Number of variables: $3 \times 128 = 384$.
- For ten rounds: Number of equations: 5760, Number of variables: 3840, Number of quadratic equations: 2560.
- The system of equations is banded (hence sparse). The states of each round appear in only one block, key and output variables common in adjacent blocks.

Variants of AES and properties

AES notation $S(n_r, p, q, n)$,
key length $= p \times q \times n =$ block length,

- n_r Number of rounds, p, q State matrix of bytes rows and columns, n degree of finite field extension (byte length).
- AES128 is $S(10, 4, 4, 8)$
- Irreducible polynomial of the field extension is specified. Different polynomial defines isomorphic AES.
- Round constants can be varied. Key mixing constant in every round is a power of the root of the polynomial.
- AES is closed under composition $C_1 = E_{K_1}(P)$, $C_2 = E_{K_2}(C_1)$ then there exists K such that $C_2 = E_K(P)$.
- Map inversion from ciphertext (or arrow reversion) possible for AES. Gives a fixed point problem.

Known records

- Only $S(r, 1, 1, n)$ solved for $(r, n) = (2..10, 4)$, $(r, n) = (2, 8)$ as of 2006 publication.
- SAT approach reported in 2010. Records unknown.
- Earlier approaches: Grobner basis and XL methods.

Boolean equations

- Polynomial equations in algebras over a field K

$$f(x, y, z) = 0$$

defined by algebraic operations of $K[x, y, z]$. Solutions in $\bar{K}$, algebraic extension of K.

- Boolean algebras: (analogous to polynomial rings)

$$\begin{aligned} B_0 &= (0, 1, \vee, \wedge, \neg) \\ B_1 &= (0, 1, x, \neg x, \vee, \wedge, \neg) \end{aligned}$$

$B_0 \subset B_1 \subset B_2 \ldots$

- In general B_n consists of all disjunctions of subsets of the set of all minterms in n variables and is the set of all Boolean functions $f : B_0^n \to B_0$. Hence $|B_n| = 2^{2^n}$.

Boolean Rings

For example

$$
\begin{aligned}
B_2 &= \{0, 1, x, x', y, y', x \wedge y, x \wedge y', x' \wedge y, x' \wedge y', \\
&\quad x \vee y, x \vee y', x' \vee y, x' \vee y', x' \wedge y \vee x \wedge y', x \wedge y \vee x' \wedge y'\}
\end{aligned}
$$

Boolean rings:

$$
\begin{aligned}
R_0 &= \mathbb{F}_2 = (0, 1, \oplus, .) \text{ the binary field} \\
R_1 &= \mathbb{F}_2/(x^2 - x) \\
R_2 &= \mathbb{F}_2[x, y]/(x^2 - x, y^2 - y)
\end{aligned}
$$

Boolean algebra to associated Boolean ring

To every Boolean algebra $(\{B, 0, 1\}, \vee, \wedge, \neg)$ there is associated Boolean ring $(\{R, 0, 1\}, \oplus, .)$ with Boolean ring-algebra correspondence:

$$\begin{array}{rcl} x \oplus y & = & (x \wedge \neg y) \vee (\neg x \vee y) \\ x \vee y & = & x \oplus y \oplus xy \\ \neg x & = & x \oplus 1 \end{array}$$

Boolean algebra to associated Boolean ring

To every Boolean algebra $(\{B, 0, 1\}, \vee, \wedge, \neg)$ there is associated Boolean ring $(\{R, 0, 1\}, \oplus, .)$ with Boolean ring-algebra correspondence:

$$\begin{aligned} x \oplus y &= (x \wedge \neg y) \vee (\neg x \vee y) \\ x \vee y &= x \oplus y \oplus xy \\ \neg x &= x \oplus 1 \end{aligned}$$

Change of notations

In a Boolean algebra B denote $x \wedge y$ by xy, $x \vee y$ as $x + y$, $\neg x$ as x'. xy to also denote product in the associated Boolean ring R.

Boolean equations

B a Boolean algebra, $f_i, g_i : B^n \to B$, $i = 1, 2, \ldots, m$ functions defined by rules of operations in B. A Boolean system of equations is

$$f_i(x_1, x_2, \ldots, x_n) = g_i(x_1, x_2, \ldots, x_n)$$

Such an equation (or a system) is said to be consistent if there exist elements $a_1, \ldots, a_n$ in B such that

$$f_i(a_1, \ldots, a_n) = g_i(a_1, \ldots, a_n)$$

All such equations can be expressed in terms of an equation with a single Boolean function F,

$$F(x_1, \ldots, x_n) = \sum_i (f_i \oplus g_i) = 0$$

Boole-Shannon expansion

Theorem

Let be B be a Boolean algebra and $f : B^n \to B$ be a Boolean function. Then f can be expanded in two ways as

$$\begin{aligned} f(x_1, \ldots, x_n) &= x_1 f(1, x_2, \ldots, x_n) + x_1' f(0, x_2, \ldots, x_n) \\ f(x_1, \ldots, x_n) &= [x_1 + f(0, x_2, \ldots, x_n)][x_1' + f(1, x_2, \ldots, x_n)] \end{aligned}$$

Normal forms

Consequences of Boole-Shannon expansion

- If $F : B^n \to B$ then f has unique minterm canonical form

$$f = \sum_{a \in \{0,1\}^n} f(a) X^a$$

$a = (\alpha_1, \ldots, \alpha_n)$, $x^a = \prod x_i^{\alpha_i}$, $x_i^1 = x_i$, $x_i^0 = \neg x_i$.

- Conjunctive Normal Form (CNF),

$$\prod_j C_j = 1$$

where C_j are CNF expressions

- Disjunctive Normal Form (DNF),

$$\sum_j D_j = 0$$

where D_j are DNF expressions.

Solutions over Boolean algebras and SAT

Solutions of $F = 0$

The solutions of Boolean equations with co-efficients in B_0 that are in B_0 are SAT assignments. However solutions exist even over higher Boolean algebras if the equation is consistent. This fact has interesting and important implications.

Where do Boolean equations arise

- Design of switching circuits: synthesis, verification, reduction.
- AI and automated reasoning: Propositional logic, predicate logic, constraint programming.
- Software verification.
- Economics and marketing, discrete operations research.
- Search problems: molecular databases for drug discovery, chemistry, life sciences.
- Design of experiments: agriculture.
- Arithmetic of computations over groups, finite fields, number theory.
- Cryptography and cryptanalysis.
- SATisfiability: graph theory problems, complexity.

Applications to cryptology

- Algebraic cryptanalysis: Cipher algorithms expressed as Boolean equations.
- Hash function collision search. Condition as Boolean equations.
- Building models from data. Boolean Identification, modeling encryption of unknown algorithms.
- Boolean models of HW implementations: side channel analysis.
- Solving number theory problem: RSA factorization, Discrete log computation in finite fields.
- Boolean equation models of elliptic curves and high speed arithmetic.

Boolean Logic

Identities in switching logic (valid only in B_0)

- $x + y = 1$ iff $x = 1$ or $y = 1$.
- $xy = 0$ iff $x = 0$ or $y = 0$.

In Boolean logic

- $x + y = 1 \Leftrightarrow x = y' + p$ or $y = x' + p$. (But $x + y = 0 \Leftrightarrow x = 0, y = 0$ as in B_0).
- $xy = 0$ iff $x = y'q$ or $y = x'q$. (But $xy = 1 \Leftrightarrow x = 1, y = 1$ as in B_0).

Applying switching logic to Boolean equations may not give all solutions. Solutions obtained using Boolean algebra contain all solutions.

Inequality in Boolean algebra

If x, y are in B, we write $x \leq y$ if $x + y = y$. It follows:

$$\begin{aligned} x \leq y \quad y \leq z \quad &\Rightarrow x \leq z \\ x \leq y \quad &\Leftrightarrow xy' = 0 \end{aligned}$$

Example: Solving 2-CNF expressions

$$(x+y)(x'+z)(y+z)(y'+w)=1$$

From first two brackets $y=x'+p_1$, $z=x+p_2$. Third bracket is a tautology. Last bracket gives $w=y+p_3$. Hence the parametric solution is: $y=x'+p_1$,$z=x+p_2$,$w=y+p_3=x'+p_1+p_3$. All SAT assignments are

x	y	z	w
1	p_1	1	p_1+p_3
0	1	p_2	1

for p_1, p_2, p_3 arbitrary in $0, 1$.

Using DPLL algorithm to solve a system

$$\begin{array}{rcl} f(x,y,z) = xy \oplus yz \oplus zx & = & 0 \\ g(x,y,z) = x \oplus yz & = & 1 \end{array}$$

Shannon (DNF) expansion for f

$$\begin{array}{rcl} f & = & x(y \oplus z \oplus yz) + x'(yz) \\ & = & x(y + y'z) + x'(yz) \\ & = & xy + xy'z + x'yz \end{array}$$

Hence $f = 0$ is equivalent to CNF on conjugation

$$(x' + y')(x' + y + z')(x + y' + z')$$

Using Shannon (CNF) expansion for $g = x \oplus yz$ gives CNF for $g = 1$,

$$\begin{aligned} g &= [x' + (yz)'][x + yz] \\ &= (x' + y' + z')(x + y)(x + z) \end{aligned}$$

CNF database for equations $f = 0, g = 1$

$$(x' + y')(x + y)(x + z)(x' + y + z')(x + y' + z')(x' + y' + z')$$

DPLL to solve the system

Splitting rule at x

$$X_x = y'(y + z')(y' + z')$$

which has $(1, 0, 0)$ as a solution.

DPLL to solve the system

Splitting rule at x

$$X_x = y'(y + z')(y' + z')$$

which has $(1, 0, 0)$ as a solution.

$$X_{-x} = yz(y' + z')$$

which is unSAT. Hence $(1, 0, 0)$ is the only solution.

Solving over higher Boolean algebra

CNF expression

$$(x' + y')(x + y)(x + z)(x' + y + z')(x + y' + z')(x' + y' + z')$$

- $(x' + y') \to x' = y + p_1$.
- $(x + y) = (y' p_1' + y) \to p_1 = 0$.
- $(x + z) = (y' + z) \to z = y + p_2$.
- $(x' + y + z') = (y + z') = (y + y' p_2') \to p_2 = 0$.
- $(x + y' + z') = y' = 1$.
- $(x' + y' + z') \to T$.

Only solution is $(1, 0, 0)$.

Decision diagram based search

$$f(x_1, x_2, x_3) = 1 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_1x_2 \oplus x_2x_3 \oplus x_1x_2x_3 = 0$$

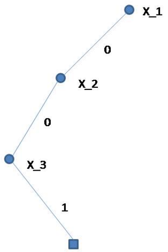

Figure: Decision Diagram

Decision diagram based search

$$f(x_1, x_2, x_3) = 1 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_1x_2 \oplus x_2x_3 \oplus x_1x_2x_3 = 0$$

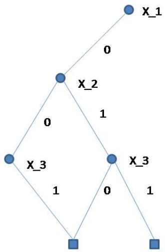

Figure: Decision Diagram

Decision diagram based search

$$f(x_1, x_2, x_3) = 1 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_1x_2 \oplus x_2x_3 \oplus x_1x_2x_3 = 0$$

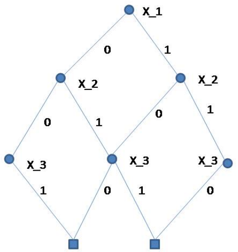

Figure: Decision Diagram

Boolean equation method

By Boole-Shannon expansion

$$f = x_1 f(x_1 = 1) + x_1' f(x_1 = 0)$$

Eliminant at x_1

$$f(x_1 = 1) = x_3$$

$$f(x_1 = 0) = 1 \oplus x_2 \oplus x_3 \oplus x_2 x_3$$

$$\text{Celim}\,(f, x_1) = x_3(1 \oplus x_2 \oplus x_3 \oplus x_2 x_3) = 0$$

for all x_2, x_3. Hence all solutions are

x_2	x_3	f	x_1
0	0	$1 \oplus x_1$	0
1	0	0	0, 1
0	1	x_1	0
1	1	x_1	0

Theorem on existence and structure of solutions

Theorem

Let $f : B \to B$ be a function on a Boolean algebra B. The equation $f = 0$ is consistent iff

$$f(0)f(1) = 0$$

Let S be the set of all solutions of a consistent equation $f = 0$. Define sets

$$\begin{aligned} I &= \{x \in B | f(0) \leq x \leq f'(1)\} \\ P &= \{f(0) + pf'(1), p \in B\} \end{aligned}$$

Then $I = P = S$.

Boolean equation in one variable

- For a Boolean function $f : B \to B$ an equation $f = 0$ is of the form

$$ax + bx' = 0$$

  Consistency: If there is a solution z in B, then $az = bz' = 0$ which is equivalent to $b \leq z \leq a'$. Hence $b \leq a' \Leftrightarrow ab = 0$. Conversely if $ab = 0$ then $b \leq a'$ and all z such that $b \leq z \leq a'$ in particular $z = b, z = a'$ are solutions.
- To show that every solution z has a parametric form $z = b + pa'$, p in B, just verify that if z is a solution of a consistent equation, then for $p = zb'$, $z \oplus (b + pa') = 0$. This proves the above theorem.
- The set S is called the set of all particular solutions, I the set of subsumptive general solutions, P the ste of parametric general solutions.

Equation in several variables

Let $f : B^n \to B$ be a Boolean function. Define

$$\begin{array}{rcl} f_0 & = & f \\ f_1 & = & f_0(0, x_2, \ldots, x_n) f_0(1, x_2, \ldots, x_n) \\ \vdots & \vdots & \vdots \\ f_n & = & f_{n-1}(0) f_{n-1}(1) \end{array}$$

f_{i+1} is called conjunctive eliminant of f_i denoted $\mathrm{Celim}\,(f_i, x_i)$ w.r.t. x_i.

Corollary

- A Boolean equation $f(x_1, \ldots, x_n) = 0$ is consistent iff $f_n = 0$.
- The subsumptive general solution of $f = 0$ is given by

$$\begin{array}{ccc} f_{n-1}(0) & \leq x_n \leq & f'_{n-1}(1) \\ f_{n-2}(0) & \leq x_{n-1} \leq & f'_{n-2}(1) \\ \vdots & \vdots & \vdots \\ f_0 & \leq x_1 \leq & f'(1) \end{array}$$

- Parametric general solution is given by

$$x_{i+1} = f_i(0) + p_i f_i'(1)$$

for parameters p_i in $B_0[x_1, \ldots, x_i]$, $i = 1, \ldots, n-1$.

Orthonormal expansions

A set of Boolean functions $\{\phi_1, \ldots, \phi_n\}$ is called an orthonormal system if

$$\begin{aligned} \phi_i\phi_j &= 0, i \neq j \\ \sum_i \phi_i &= 1 \end{aligned}$$

A Boolean function $f(X)$ can be expanded w.r.t. an orthonormal system as

$$f(X) = \sum_i \alpha_i(X)\phi_i(X)$$

where α_i can be obtained by solving Boolean equations

$$\alpha_i f = \alpha_i \phi_i$$

If $\phi_i = 1$ are consistent then $\alpha_i = f(\phi_i = 1)$.

Boolean operations in terms of an expansion

If $f = \sum_i f_i \phi_i$, $g = \sum_i g_i \phi_i$ then

$$\begin{array}{rcl} f+g & = & \sum_i (f_i + g_i)\phi_i \\ fg & = & \sum_i f_i g_i \phi_i \\ f' & = & \sum_i f_i' \phi_i \end{array}$$

Examples of orthonormal systems

$$\begin{array}{ll} \text{1 var} & x, x' \\ \text{2 var} & xy, x'y, xy', x'y' \\ & x, x'y, x'y' \\ & x', \ldots \\ \text{3, var} & x, x'y, x'y'z, x'y'z' \end{array}$$

SAT formulation of DL in $\mathbb{F}_{2^m}$

DL problem In the finite field K. Compute $x < n = |K^*|$ given a, b in K such that

$$b = a^x$$

- Known algorithms use the group property of K^*. Index calculus uses field K in indirect way (smooth polynomials over the base field). Best algorithms are sub-exponential order

$$O(\exp[c(\log n)^{\epsilon}(\log \log n)^{1-\epsilon}]$$

  where $\epsilon < 1$.
- Boolean equation based algorithm.
  - Unit computational operations Boolean.
  - Explicit use of properties of K.
  - Present Formulation: valid only for $K = \mathbb{F}_{2^m}$. (Possible for $K = \mathbb{F}_{p^m}$ for small p, requires computing DL in $\mathbb{F}_p$).

MQ system

Proposition

DL is the unique solutions of the MQ system over $\mathbb{F}_{2^m}$

$$(x_i V_i \oplus 1) T^{i+1} \oplus T^i = 0$$

$i = 0, \ldots, m-1$ with boundary conditions $T^0 = b$, $T^m = 1$ where unknowns $x_i = 0, 1$ are DL bits, $V_0 = a$, $V_{i+1} = V_i^2$ and variables T^i in $\mathbb{F}_{2^m}$.

Boolean system of equations

Let t^i denote m-tuples of binary co-ordinates of T^i in a fixed basis chosen to represent $\mathbb{F}_{2^m}$ as a vector space $\mathbb{F}_2^m$.

Proposition

The MQ system above has following representation as an MQ system

$$x_i t^{i+1} \oplus F_i(t^{i+1}, t^i) = 0$$

where x_i and components of t^i are Boolean variables and maps F_i have linear functions in their components.

- Each of the above equations has only one quadratic term. Other terms are linear.
- Each bit x_i appears in a single block of m equations.
- The system gives a generic model of a hard SAT instance of an MQ system due to the perceived hardness of the DL problem.

Diffie Hellman conjecture

- Diffie Hellman Problem Given a, $b = a^x$, $c = a^y$ in a group (for some unknown x, y), compute $s = c^x = b^y$.
- If $x = \mathrm{Dlog}_a b$ is computed then s can be computed in polynomial time.
- Diffie Hellman conjecture Computing s is as much difficult as computing x from (a, b).

Affirmative answer to Diffie Hellman conjecture in SAT sense

Proposition

The DL MQ system together with MQ system representing the equation $s = c^x$ is reducible in polynomial time to an MQ system of the form

$$\Phi(T^i, R^i) = 0$$

with boundary conditions $T^0 = b$ and $R^0 = s$, $T^m = R^m = 1$

Proof: By Boolean elimination of x_i from combined MQ systems for $b = a^x$ and $s = c^x$.

Affirmative answer to Diffie Hellman conjecture in SAT sense

Proposition

The DL MQ system together with MQ system representing the equation $s = c^x$ is reducible in polynomial time to an MQ system of the form

$$\Phi(T^i, R^i) = 0$$

with boundary conditions $T^0 = b$ and $R^0 = s$, $T^m = R^m = 1$

Proof: By Boolean elimination of x_i from combined MQ systems for $b = a^x$ and $s = c^x$.

Corollary

The DH problem in $\mathbb{F}_{2^m}$ is polynomial time equivalent to DL computation as a SAT instance.

Decisional DHP

- If $b = a^x$, $c = a^y$ is a Diffie Hellman session, the Decisional Diffie Hellman Problem (DDHP) is to decide given s whether $s = a^{xy}$.
- SAT formulation: s is the shared key iff $s = b^y = c^x$, $b = a^x$, $c = a^y$ is SAT.
- DDHP is solved by SAT assignments in an MQ system of the same form as that in the DL problem.

Theorem

DDHP is polynomial time equivalent to DL computation as a SAT instance.

Conclusions

- Most ciphers designed to withstand statistical and TMTO attacks. Urgent need to evaluate security by algebraic cryptanalysis which can be carried out by Boolean equation solving.
- SAT problems can be solved by Boolean equation methods. Theory of Boolean equations has unexplored potential for parallel algorithms.
- Boolean equation formulation of AC of block and stream ciphers, number theory problems such as discrete log and factorization are not well explored.
- New high speed algorithms for arithmetic, group and finite field operations, elliptic curves can be explored from this viewpoint.
- HW realizations in FPGAs using Boolean equation theory are not well explored.

Conclusions

- Most ciphers designed to withstand statistical and TMTO attacks. Urgent need to evaluate security by algebraic cryptanalysis which can be carried out by Boolean equation solving.
- SAT problems can be solved by Boolean equation methods. Theory of Boolean equations has unexplored potential for parallel algorithms.
- Boolean equation formulation of AC of block and stream ciphers, number theory problems such as discrete log and factorization are not well explored.
- New high speed algorithms for arithmetic, group and finite field operations, elliptic curves can be explored from this viewpoint.
- HW realizations in FPGAs using Boolean equation theory are not well explored.

Thank You