THE VALUE OF PERFORMANCE.
NORTHROP GRUMMAN

Northrop Grumman Cybersecurity Research Consortium (NGCRC)

Fall 2014 Symposium

Crypsis: Efficient Confidentiality Preserving Big Data Analysis in Untrusted Clouds

04 Nov 2014

Julian Stephen, Savvas Savvides, Russell Seidel and Patrick Eugster

Purdue University

Problem Statement

NORTHROP GRUMMAN

"Today, running your business on private servers is on the same level of odd behavior as carrying scuba tanks to provide a private air supply"

RIP Server, Peter Coffee, Mar 29, 2014

- Data breach: "The Cloud Multiplier Effect" (Ponemon Institute)
  - Increased use of cloud can increase the probability of a $20 million breach by as much as 3x
  - 36 percent of business-critical applications are housed in the cloud
  - 30 percent of business information is stored in the cloud
- Challenges
  - How safe is it to trust a third party cloud provider?
  - How can banking, finance and insurance sectors leverage this potential?

Preserving Confidentiality

NORTHROP GRUMMAN

- *Fully* homomorphic encryption (FHE)
  - Prohibitive overhead, getting more practical
  - Limited expressiveness
- *Partially* homomorphic encryption (PHE)
  - Allows for certain operations to be performed in encrypted form
  - E.g.,
    - Paillier [Paillier;EuroCrypt'99] ► AHE: $D(E(x1) \oplus E(x2)) = x1 + x2$
    - ElGamal [ElGamal; ToIT'86] ► MHE: $D(E(x1) \oplus E(x2)) = x1 * x2$
    - DET (=), OPE (<)
- Conjecture
  - Partition programs according to attributes and use a different cryptosystem for each
  - We can use multiple PHE cryptosystems in parallel
  - Reencryption between PHE systems may be faster than FHE

Background

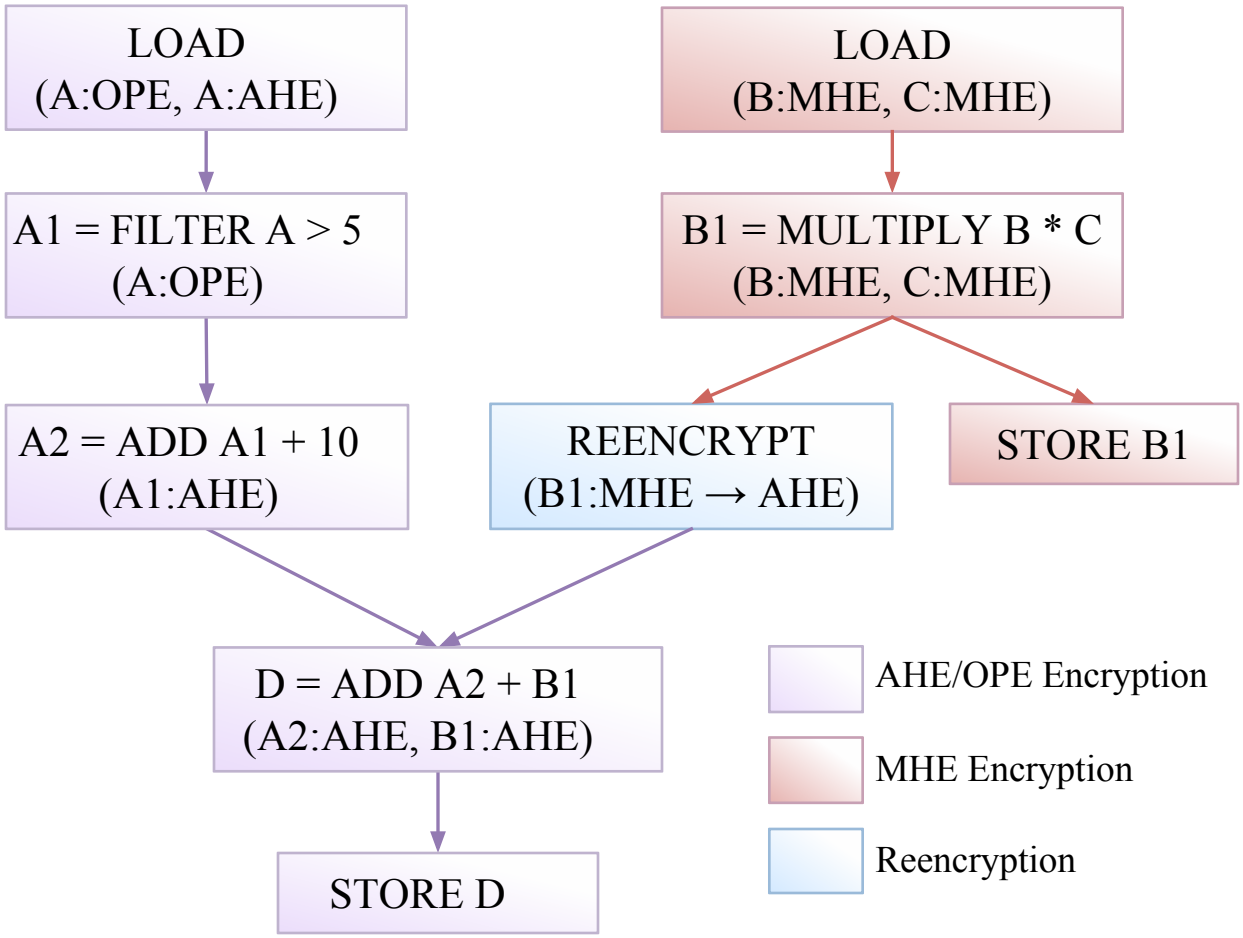
NORTHROP GRUMMAN

- Map Reduce [Dean&Ghemawat; OSDI'04]
  - Parallel execution (map and reduce functions)
  - Hadoop version 1.2.1
- Pig and Pig Latin [Gates et.al; VLDB'09]
  - Pig Latin - High level data flow language for expressing data analysis programs
  - Pig - runtime environment, generates Map Reduce programs
  - Pig version 0.11.1
- Example Pig Latin script

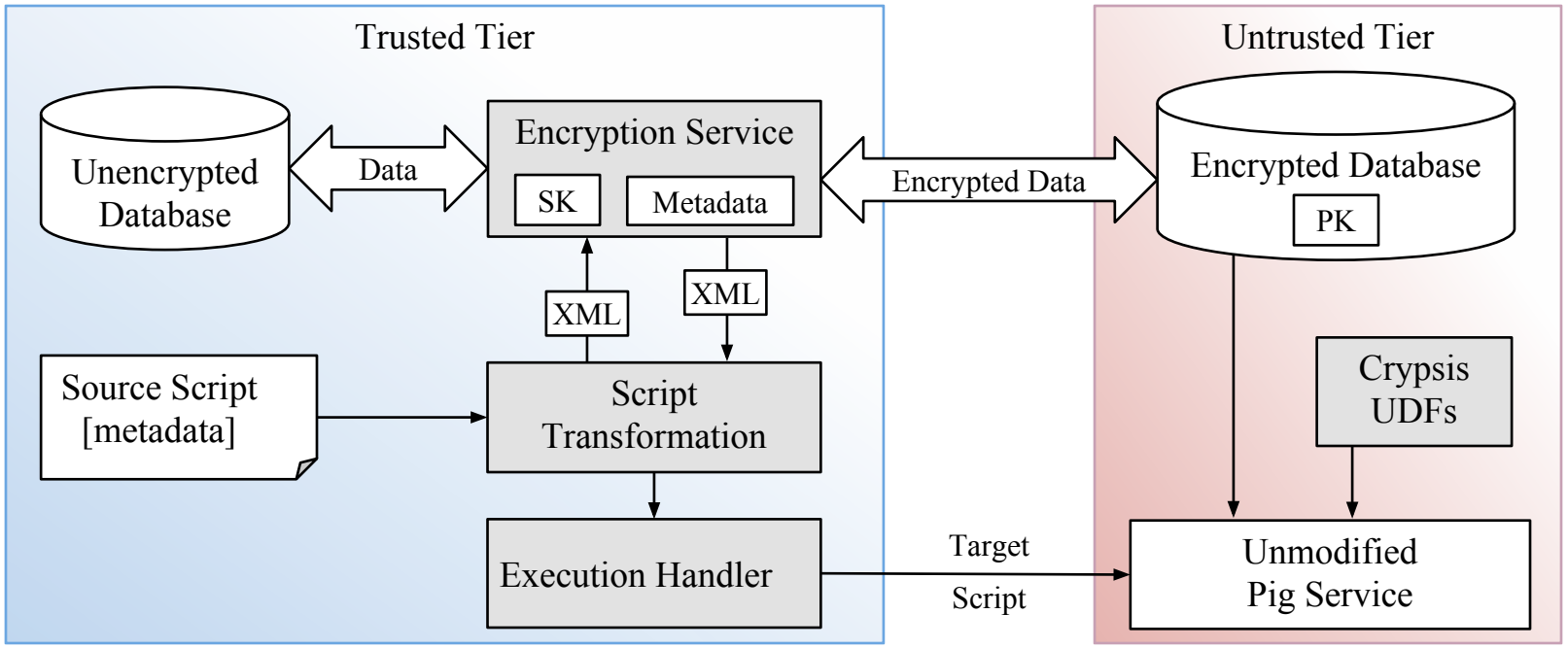
```
A = LOAD "file1" AS (a0, a1);
B = FILTER A BY a0 > 10;
C = GROUP B BY a1;
D = FOREACH C GENERATE group AS b0, SUM(C.a0) AS b1;
STORE D INTO "output";
```

Crypsis Intuition

NORTHROP GRUMMAN



Architecture Overview



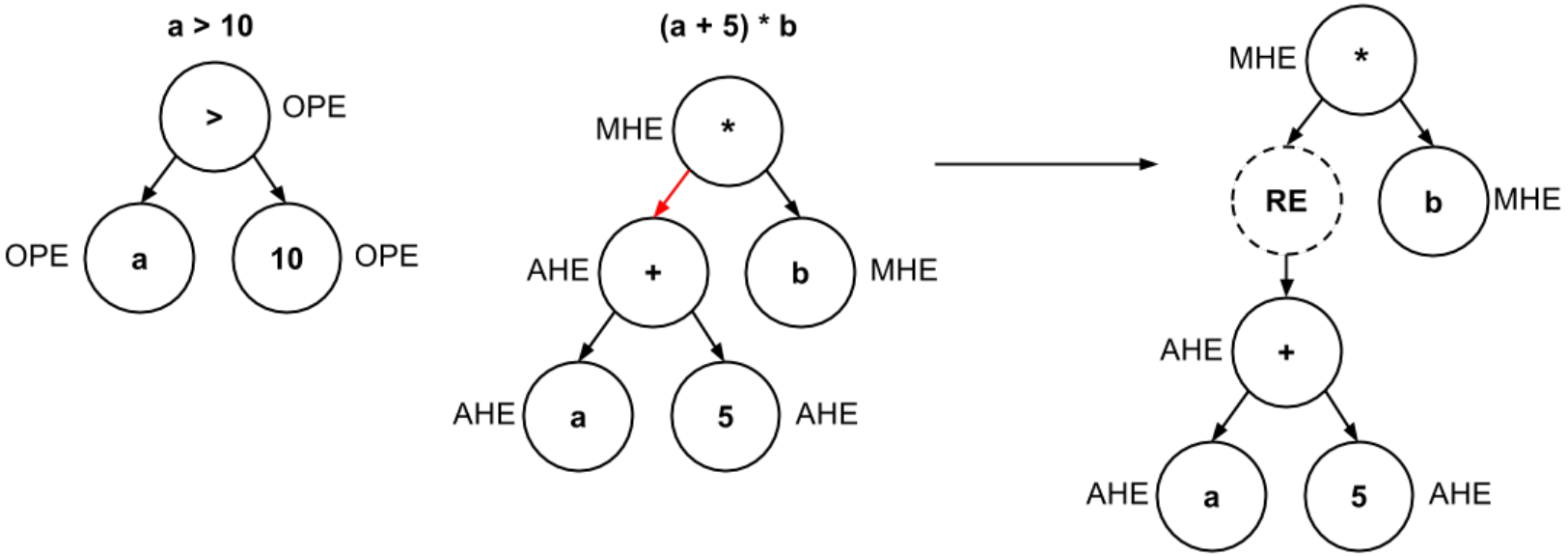
Practical Confidentiality Preserving Big Data Analysis
[J. Stephen et al; USENIX HotCloud14]

Script Transformation

NORTHROP GRUMMAN

- Script analysis
  - Generate Data Flow Graph (DFG)
  - Nodes are relations (LOAD, FOREACH, etc…)
  - Edges are data flow between operators
- Generate Map of Expression Trees (MET)
  - Contains all expressions of the script
  - Keys are used to assign expressions to DFG
- Generate Set of Annotated Fields (SAF)
  - One entry for each <relation, field> of the script
  - <relation, field>, parent, available encryptions, required encryptions
  - Get available encryptions from lineage of field (parent)
  - Get required encryptions using MET

Identifying Reencryptions



- Reencryption required when:
  - Required encryption not available
  - Incompatible operations e.g. addition followed by a multiplication
  - Reencryption is conceptual (can continue computation on client)

- 17 PigMix2 benchmarks (PigMix1 + 5)
  - Only script 8 requires reencryption (averaging)
  - 1 additional script requires same attribute available in 2 encryptions

Transformation Example

NORTHROP GRUMMAN

Source Script

```
A = LOAD "file1" AS
    (a0, a1);
B = LOAD "file2" AS
    (x0);
C = FILTER A BY a0 > 10;
D = GROUP C BY a1;
E = FOREACH D GENERATE group AS
    b0, SUM(C.a0) AS b1;
F = JOIN E BY b0, B BY x0;
STORE F INTO "outfile";
```

Target Script

```
A = LOAD "enc_file1" AS
    (a0_ope, a0_ahe, a1_det);
B = LOAD "enc_file2" AS
    (x0_det);
C = FILTER A BY a0_ope > 918...234;
D = GROUP C BY a1_det;
E = FOREACH D GENERATE group AS
    b0, ENCSUM(C.a0_ahe) AS b1;
F = JOIN E BY b0, B BY x0_det;
STORE F INTO "enc_outfile";
```

Program Analysis for Secure Big Data Processing
[J. Stephen et al; IEEE/ACM ASE2014]

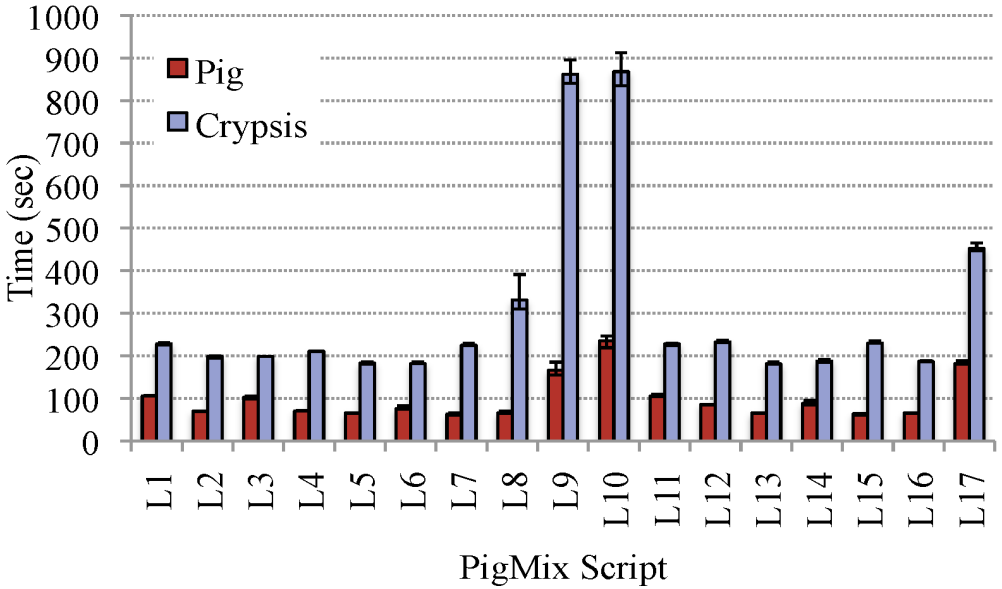
Crypsis Implementation

NORTHROP GRUMMAN

- Crypsis UDFs
  - Replace operations and aggregation functions with their encrypted version
  - Allows for an unmodified Pig service
- Secondary homomorphic property
  - AHE: $D(E(x1) \odot x2) = x1 * x2$
  - MHE: $D(E(x1) \odot x2) = x1 \wedge x2$
- Augmented expressiveness
  - $+, -, \sim, *, \wedge, ==, <, \leq, >, \geq$
  - Aggregation functions: SUM, MAX, MIN, DISTINCT, ORDERBY, AVG, MEDIAN, ABS
  - Negative numbers

Evaluation (PigMix)

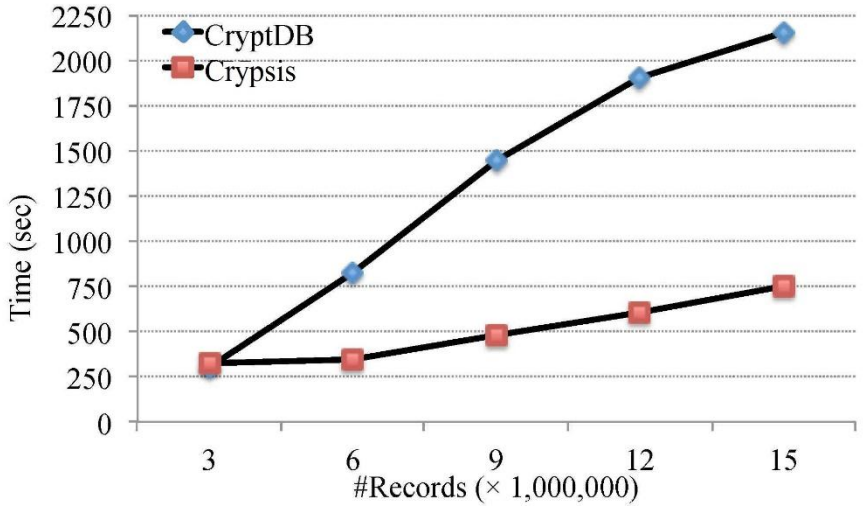
NORTHROP GRUMMAN

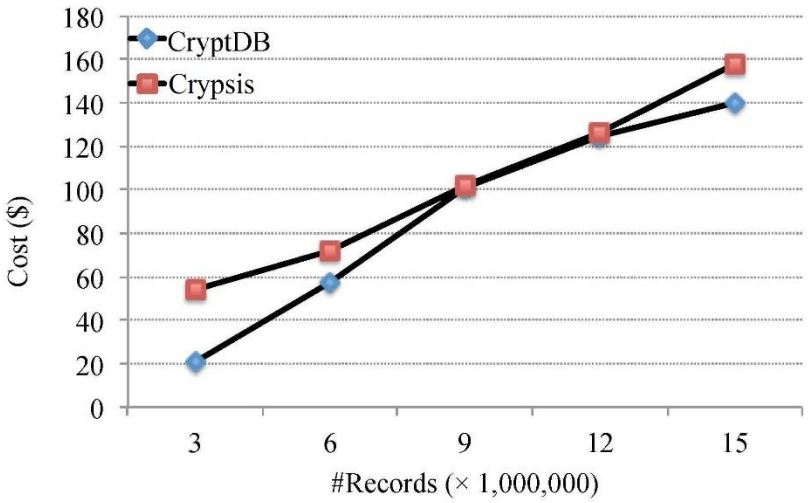


- 11 ec2 c3.large instances (2 vCPUs, 3.75GB ram)
- 5GB of data (over 3 million rows)
- An average of 3x overhead in terms of latency
- FHE can exhibit several 100 times overhead

Evaluation (CryptDB comparison)

NORTHROP GRUMMAN



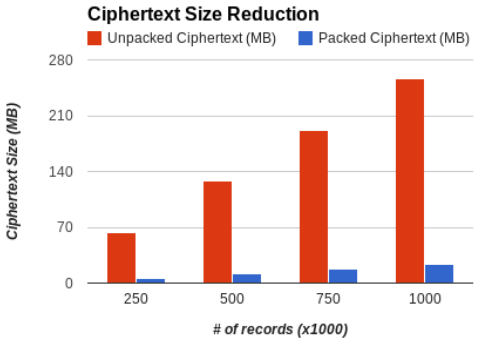


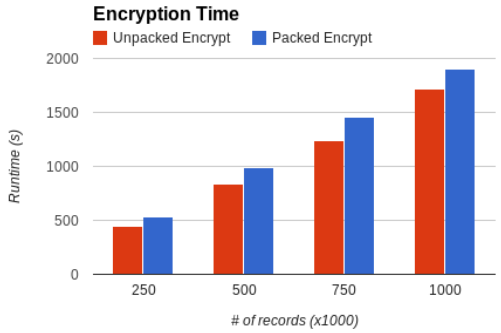
- 3 m3.medium instances (1 vCPUs, 3.75GB ram)
- ~ 3x faster for 15 million records
- Similar overall cost

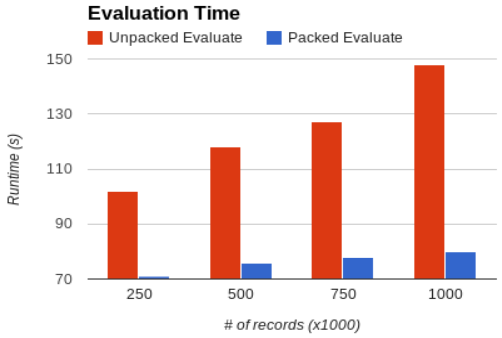
Packing Optimization

- Minimize ciphertext overhead
  - Pack multiple values in a single plaintext before encrypting
  - Must handle overflows

$Padding\ n$	x_n	...	$Padding\ 1$	x_1
		$\otimes$		
$Padding\ n$	y_n	...	$Padding\ 1$	y_1
		$=$		
$x_n + y_n$		...	$x_1 + y_1$	







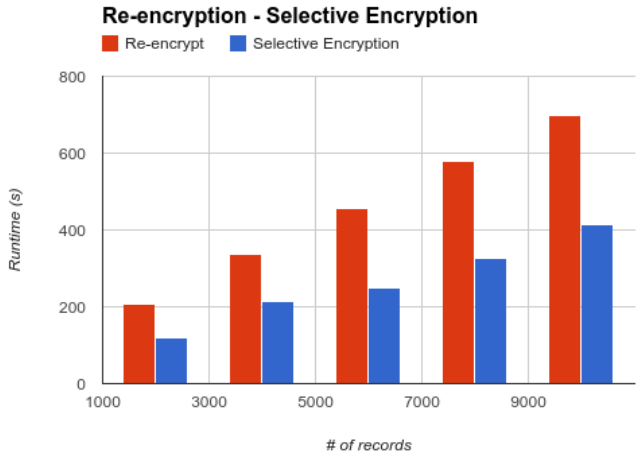
Selective Encryption

NORTHROP GRUMMAN

- Often only parts of input data need to be encrypted: selectively encrypt
  - Reduce overall size of data
  - Reduce required re-encryptions e.g. (a + b) * c

```
X = ENC_ADD(a_ahe, b_ahe)
Y = REENCRYPT(A, ahe->mhe)
Z = ENC_MULT(B, c_mhe)
```

```
X = ENC_ADD(a_ahe, b_ahe)
Y = ENC_PMULT(A, c_plain)
```



Limitations

NORTHROP GRUMMAN

- Iterations – recursion
  - Pig Latin is a query based language
  - No support for iterations or recursion
  - Use of UDFs for more complex operations
  - Iterations over encrypted data can be very expensive (reencryptions)
- User Defined Functions (UDFs)
  - Many big data analysis languages propose UDFs vs. built-in/pre-defined operators and functions
  - Black-boxes from perspective of program analysis (Cannot support transformation)
  - Can analyze byte-code on Java UDFs

Opportunities for Reducing Reencryption

NORTHROP GRUMMAN

- Statement re-ordering, e.g.,

```
a1 < 0   (OPE)
...          (CS1 != OPE)
a1 > 20 (OPE)
```

```
a1 < 0  (OPE)
a1 > 20 (OPE)
...          (CS1 != OPE)
```

- Computation re-writing, e.g.,

```
t1 = t2 * (t3 + t4)
...
t5 = t1 + t6
```

```
t1 = t2 * t3
t7 = t2 * t4
...
t5 = t1 + t7 + t6
```

- Condition re-writing, e.g.,

```
(x1 + x2 > 0)
```

```
(x1 > 0 AND x2 > 0)
```

Related Work

NORTHROP GRUMMAN

- CryptDB [Popa et al.;SOSP'11]
  - Encrypted database for MySQL (subset)
  - No Parallelism
  - No reencryption; client-side query completion
- MrCrypt [Lesani et al.;OOPSLA'13]
  - Program analysis for individual MapReduce tasks
  - No reencryption
- Monomi [Tu et al; VLDB'13]
  - Uses techniques to improve performance of complex queries on encrypted data
  - Built on top of Postgres, Centralized Design

Conclusions and Future Work

NORTHROP GRUMMAN

"Big data absolutely has the potential to change the way governments, organizations, and academic institutions conduct business and make discoveries, and its likely to change how everyone lives their day-to-day lives" - Susan Hauser, corporate vice president, Microsoft'

- Cloud computing
  - On demand computation infrastructure has great potential but inherent confidentiality concerns
- Crypsis
  - Addresses these confidentiality concerns. Efficient big data analysis over encrypted data
- Future work
  - More fine-grained encryption system
  - Identify more opportunities to reduce re-encryptions
- NSF Secure and Trustworthy Cyberspace grant
  - "Practical Assured Big Data Analysis in the Cloud"

THE VALUE OF PERFORMANCE.

NORTHROP GRUMMAN